Studies in Computational Intelligence 532

Roberto Cipolla
Sebastiano Battiato
Giovanni Maria Farinella *Editors*

Registration and Recognition in Images and Videos

Springer

Studies in Computational Intelligence

Volume 532

Series Editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

For further volumes:
http://www.springer.com/series/7092

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

Roberto Cipolla · Sebastiano Battiato
Giovanni Maria Farinella
Editors

Registration and Recognition in Images and Videos

Springer

Editors
Roberto Cipolla
Department of Engineering
University of Cambridge
Cambridge
United Kingdom

Sebastiano Battiato
Dipartimento di Matematica e Informatica
Università di Catania
Città Universitaria
Italy

Giovanni Maria Farinella
Dipartimento di Matematica e Informatica
Università di Catania
Catania
Italy

ISSN 1860-949X ISSN 1860-9503 (electronic)
ISBN 978-3-642-44906-2 ISBN 978-3-642-44907-9 (eBook)
DOI 10.1007/978-3-642-44907-9
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013953719

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Computer vision is the science and technology of making machines that see. It is concerned with the theory, design and implementation of algorithms that can automatically process visual data to recognize objects, track and recover their shape and spatial layout.

The International Computer Vision Summer School - ICVSS was established in 2007 to provide both an objective and clear overview and an in-depth analysis of the state-of-the-art research in Computer Vision. The courses are delivered by world renowned experts in the field, from both academia and industry, and cover both theoretical and practical aspects of real Computer Vision problems. The school is organized every year by University of Cambridge (Computer Vision and Robotics Group) and University of Catania (Image Processing Lab). Different topics are covered each year. A summary of the past Computer Vision Summer Schools can be found at: `http://www.dmi.unict.it/icvss`

This edited volume contains a selection of articles covering some of the talks and tutorials held during recent editions of the school and covering some of the key topics in computer vision. The chapters provide both an in-depth overview of challenging areas and key references to the existing literature. The main topics covered by the chapters include the visual field, advanced algorithms for visual feature extraction and description, feature matching and image registration, object detection and recognition, object tracking, image segmentation. Each chapter contains key references to the existing literature.

It is our hope that graduate students, young and senior researchers, and academic/industrial professionals will find the book useful for understanding and reviewing current approaches in Computer Vision, thereby continuing the mission of the International Computer Vision Summer School.

Sicily, Sept 2013

Roberto Cipolla
Sebastiano Battiato
Giovanni Maria Farinella

Acknowledgements

We would like to take this opportunity to thank all contributors of this book, and all people involved in the organization of ICVSS.

List of Contributors

Lamberto Ballan
University of Florence, Italy
e-mail: `lamberto.ballan@unifi.it`

Michel Barlaud
University of Nice-Sophia Antipolis, France
e-mail: `barlaud@i3s.unice.fr`

Stefano Berretti
University of Firenze, Italy
e-mail: `stefano.berretti@unifi.it`

Michael M. Bronstein
Università della Svizzera Italiana Lugano, Switzerland
e-mail: `bronstein@ieee.org`

Samuel Rota Bulò
DAIS, Università Ca' Foscari Venezia, Italy
e-mail: `srotabul@dais.unive.it`

Marco Cristani
Università degli Studi di Verona, Italy
e-mail: `marco.cristani@univr.it`

Alberto Del Bimbo
University of Firenze, Italy
e-mail: `alberto.delbimbo@unifi.it`

Jingming Dong
University of California, Los Angeles, USA

Graham D. Finlayson
University of East Anglia, UK
e-mail: graham@cmp.uea.ac.uk

Andrew Fitzgibbon
Microsoft Research, Cambridge, UK
e-mail: awf@microsoft.com

Jan Koenderink
Katholieke Universiteit Leuven, België
e-mail: jan.koenderink@ppw.kuleuven.be

Jonathan Masci
IDSIA, USI and SUPSI, Switzerland
e-mail: jonathan@idsia.ch

Jiří Matas
The Center for Machine Perception, Prague, Czech Republic
e-mail: matas@cmp.felk.cvut.cz

Davide Migliore
Evidence Srl, Italy

Vittorio Murino
Istituto Italiano di Tecnologia, Italy
e-mail: vittorio.murino@iit.it

Pietro Pala
University of Firenze, Italy
e-mail: pietro.pala@unifi.it

Marcello Pelillo
DAIS, Università Ca’ Foscari Venezia, Italy
e-mail: pelillo@dais.unive.it

Kari Pulli
NVIDIA Research, Santa Clara, USA
e-mail: karip@nvidia.com

Whitman Richards
Massachusetts Institute of Technology, USA
e-mail: wrichards@mit.edu

Daniel Rueckert
Imperial College London, UK
e-mail: d.rueckert@imperial.ac.uk

Juergen Schmidhuber
IDSIA, Switzerland
e-mail: ron@cs.technion.ac.il

Julia A. Schnabel
University of Oxford, UK
e-mail: julia.schnabel@eng.ox.ac.uk

Stefano Soatto
University of California, Los Angeles, USA
e-mail: soatto@ucla.edu

Martin Szummer
Microsoft Research, Cambridge, UK
e-mail: szummer@microsoft.com

Lorenzo Torresani
Dartmouth College, USA
e-mail: lorenzo@cs.dartmouth.edu

Alejandro Troccoli
NVIDIA Research, Santa Clara, USA
e-mail: atroccoli@nvidia.com

Andrea van Doorn
Delft University of Technology, Netherlands
e-mail: a.j.vandoorn@tudelft.nl

Tomáš Vojíř
The Center for Machine Perception, Prague, Czech Republic
e-mail: vojirtom@cmp.felk.cvut.cz

Contents

The Visual Field: Simultaneous Order in Immediate Visual Awareness

Jan Koenderink, Andrea van Doorn, and Whitman Richards

1 Introduction

When you open your eyes in bright daylight, you become visually aware of the scene in front of you. That is to say, generically, for you might be blind, hallucinate, and so forth. To many people visual awareness appears three-fold extended in space, and evolving over time. One makes out objects and processes in intricate interrelations. Here we are mainly interested in immediate visual awareness of the type that happens when you look at a painting. We assume you close one eye and hold your position with respect to the canvas. This restriction cuts down on the complexity, for instance, it plays down the importance of visuomotor factors, binocular multiperspective, and scene changes.

Assuming a "painter's attitude" [51, 23], you become aware of a two-fold extended manifold with various qualities residing at different locations. Such qualities are colors, shapes, and so forth. One of such qualities is "depth" [47, 48].

The structure of the two-fold extendedness is conventionally denoted "the visual field" [73]. It is an aspect of visual awareness that is at quite a remove from the immediate awareness you have when you open your eyes in a generic daily life setting. However, it is evidently a "presentation", that is to say *it happens to you*, it is pre-cognitive, not a product of reflective thought. The structure of the visual field has been the subject of numerous researches in psychophysics and experimental phenomenology. In this chapter we attempt a formal structural description.

Jan Koenderink
Katholieke Universiteit Leuven
e-mail: jan.koenderink@ppw.kuleuven.be

Andrea van Doorn
Delft University of Technology
e-mail: a.j.vandoorn@tudelft.nl

Whitman Richards
Massachusetts Institute of Technology
e-mail: wrichards@mit.edu

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_1, © Springer-Verlag Berlin Heidelberg 2014

We will take it for granted that the reader is familiar with the basic optics, physiology, and psychophysics that relate to the topic [20, 64]. We will summarily discuss some aspects of *local sign* though, since these are perhaps less familiar. Although we do not explicitly discuss changes of fixation, these are implicit in our account. We treat them as not essentially distinct (except for time course and so forth) from re-directions of attention.

2 Local Sign

The term "local sign" (G.: *Localzeichen*) is due to Hermann Lotze (1817–1881), possibly a development of Ernst Heinrich Weber's (1795–1878) perceptive fields (G.: *Empfindungskreisen*; [53, 82]). Lotze notices that the brain apparently associates directions in external space with fibers of the optic nerve. Awareness is of such directions, instead of somatic locations. He understands that these (at least in awareness, not necessarily in automatic visuomotor behavior) have somehow to be *learned*, since they cannot be understood on purely anatomical (such as somatotopic maps [61]) or physiological (such as biochemical gradients [71]) grounds. Lotze's hypothesis is that the local signs are learned through the optical consequences of voluntary eye movements.

A related, though different, notion was contributed by Platt [67]. Here the idea is that an eye moment has the effect (at least within a sufficiently short time interval) of shifting straight lines in the retinal image, that are oriented along the movement, along themselves. Thus, the movement reveals sets of mutually collinear retinal locations, at least in principle. This is a very general concept. Notice that it, unlike Lotze's proposal, does not yield a metrical structure. It merely reveals (at least in principle) the equivalence classes of mutually parallel lines in the visual field.

A very different proposal is due to Hermann von Helmholtz (1821–1894) (see [35]). Helmholtz was trained as an army general practitioner and had first hand experience with various diseases. He noticed that in cases of acute toothache the patient is often unable to indicate whether the source of the pain is in the upper or in the lower jaw. Pushing against the two places individually soon localizes the trouble. Why is this? According to Helmholtz obviously because the nerves serving opposite teeth invariably are stimulated in synchrony during the chewing of food. This signals to the brain that their *Empfingdungskreisen* overlap. Generalizing from this, correlation of neural activity is likely to give rise to awareness of spatial overlap of receptive fields on the sensitive body surface.

Starting from this idea one may derive the topology of the visual field from the correlation structure of the optic nerve activity [32, 33]. Unlike Lotze's mechanism, this yields a visual field structure that is unrelated to the visual directions in external space ("internal", as opposed to Lotze's "external" local sign).

A concept that has been around since the advent of precise eye movement recordings is that temporal signals might be used to obtain fine spatial detail resolution. We find it hard to associate a specific author with the idea. A modern account is framed in analogy with the function of the whiskers in animals like the rat (the vibrissal array, [1]).

In this paper we will refer frequently to the "Lotze, Platt, and Helmholtz mechanisms" of local sign. These principles have been largely framed in the context of the brain. However, each of them is evidently important to immediate visual awareness.

Theories of automatic visuomotor actions might do without local sign at all, as has been forcefully demonstrated by Braitenberg [6].

In the contemporary understanding of vision the brain is much like a special purpose computer. This computer implements "inverse optics" computations in a bottom-up fashion, eventually leading to a visual representation of the scene in front of the eye. The basic ideas have been formulated by David Marr (1945-1980), in his highly influential posthumous book "Vision" [55]. However, it may be seriously doubted whether inverse optics algorithms are up to such a task, given the inherent ambiguities of the optical structure. This has become increasingly clear thanks to the developments in computer vision [17]. At this point we stress the obvious fact that such a bottom up "representation" would be meaningless from a phenomenological perspective [38].

3 The Fuzzy Line

The visual field does not have infinite resolution. Unlike the Euclidean plane, it is "fuzzy"[1]. The "points" are actually perceptive fields (G.: *Empfindungskreisen*) that may be quite large[2]. Moreover, one may have perceptive fields of a range of sizes at any specific location [44, 4].

For the sake of conciseness we consider the simpler case of the fuzzy line here. It illustrates most of the properties we require in this chapter. There have been previous attempts at the construction of fuzzy geometries [24, 25, 8], but for our purpose we need somewhat different structures.

We use "gaussian points", that we interpret as perceptive fields in psychophysical, and/or receptive fields in neurophysiological interpretations. They have a gaussian profile, characterized by a location and a width[3] (see figure 1). In the physiological interpretation they yield a single parameter when confronted with the retinal image, proportional to the average irradiance for the given weight. This makes them true "points" in the Euclidean sense (DEFINITION 1 of the Elements: "*a point is that which has no parts*"[4] [14]).

Different from Euclidean points, gaussian points may *overlap* though. We define two measures of relation. The overlap is the correlation[5]. It takes values between zero and one, and may be found from experience with numerous retinal images.

Another useful notion is the support of the point, which is the segment contained between the points of inflection of the weight[6] . The range can be obtained from responses to punctate-like stimulation. Given this definition we may define another measure of overlap, the "intersection". The interaction is *false* or *true* according to whether the common segment is empty or not. This may be found from the simultaneous activity under punctate stimulation.

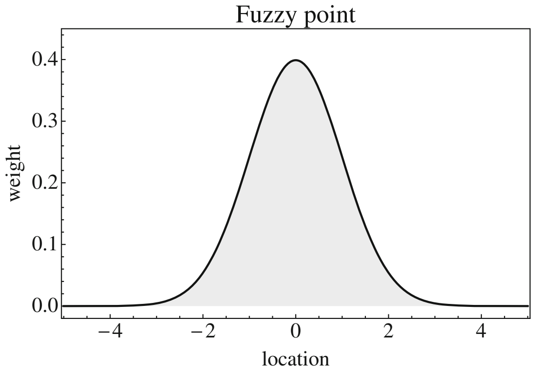


Fig. 1 This is the weight for a fuzzy point at location $x = 0$ and width $\sigma = 1$. The weight has been scaled such that the total integrated weight is unity. Thus, it is different from the conventional "fuzzy membership function".

Given the notions of support and intersection we may construct a relation of domination (figure 2). Given two points $\mathscr{P}$ and $\mathscr{Q}$ (say), then we say that $\mathscr{P}$ dominates $\mathscr{Q}$ if, and only if it is the case that for any $\mathscr{R}$ such that $\mathscr{R}$ intersects with $\mathscr{Q}$, it is the case that $\mathscr{R}$ also intersects with $\mathscr{P}$. The domination cannot be obtained from the overlap relation[7].

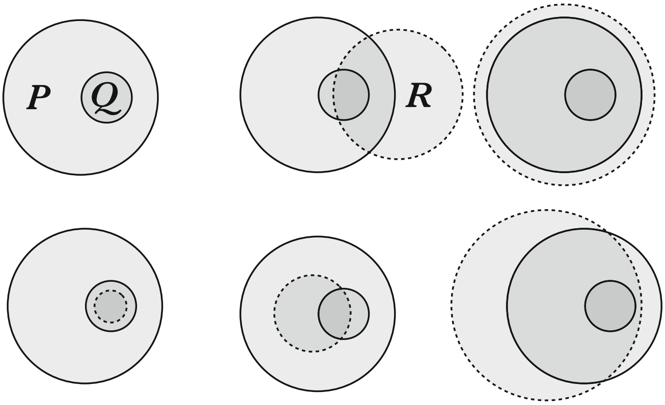


Fig. 2 The concept of "domination". At top left two regions, $\mathscr{P}$ and $\mathscr{Q}$, such that (visually) $\mathscr{Q} \subset \mathscr{P}$. How can this relation be given a purely functional meaning? The idea is shown in the figure at top center: the region $\mathscr{R}$, which overlaps with the region $\mathscr{Q}$ *automatically* overlaps with the region $\mathscr{P}$. The other subfigures show various geometries, apparently this relation continues to hold. We use it as a functional definition of the relation of domination of one region by another.

Notice that all these properties may be obtained by learning over huge sets of retinal images. The Helmholtz mechanism provides a topological structure, at least in restricted neighborhoods. The Lotze mechanism provides a coarse metric. The Platt mechanism allows one to pick out collinear points of similar widths, a projective property. When augmented with a temporal metric (the vibrissal mechanism), it provides a refined metric in local regions. In the case learning never stops, the geometry will keep up with various external changes, like growth in early childhood, change of spectacles later in life.

That the Helmholtz principle requires active development is clear from patients (*tarachopic amblyopes*) with intact, coarse Lotze local sign, but lack of fine grained, local Helmholtz local sign [22]. Since there is apparently nothing physiologically wrong with these patients, tarachopia is apparently a form of "Seelenblindheit" (*agnosia*).

It is important to notice that the fuzziness has important consequences that make the fuzzy geometry quite different from Euclidean geometry. Most importantly, while various interpolation-like manipulations work as expected, extrapolation-like manipulations do not work at all.

Here is an illustrative example. Given two points one has a "yardstick". By iteratively translating the yardstick over its own length we produce a discrete affine scale of mutually equidistant, collinear points. On the fuzzy line the fuzziness grows though, and after a finite number (typically: a few) of iterations the width of the newly added points will exceed the length of the yardstick[8]! On the other hand, interpolation has the opposite effect: the interpolated points have *less* width than the fiducial points (see figure 3).

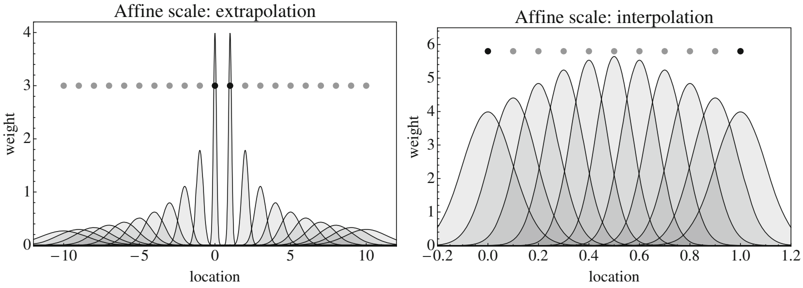


Fig. 3 Examples of affine scales. At left a case of extrapolation, at right a case of interpolation. As one extrapolates the width soon "explodes". Interpolation is fully stable though, the interpolated points are actually "sharper". The gray dot sequences indicate the Euclidean equivalent point series. The black dots indicate the fiducial point pair.

The overlap yields a kind of distance between points. One minus the overlap is zero for two identical points (same location, same width), and increases up to maximally one when the points are very different, either in location or width. Points with less than a certain small difference from a given point lie inside a circular region centered on the given point in the parameter (that is x–σ) plane[9].

The parameter plane turns out to have the metric of the hyperbolic plane[10].

4 The Fuzzy Plane

Most of the properties of the fuzzy plane can be immediately interpreted in analogy with the fuzzy line. The major changes have (obviously) to do with the additional dimension. Here we mention only a few topics, sufficient for our cause. Although it is not that hard to treat the fuzzy plane analytically, the relevant formulas are rather complicated. Since it is very easy to *simulate* the fuzzy plane, we opt for that here. It suffices to illustrate the major facts.

Consider geometrical constructions that start from a fiducial point configuration, and result in a target point. Perhaps the simplest example would be the bisection of a segment defined by two points (see figure 4). It is almost trivial to simulate this. One considers two points, each one drawn from the gaussian probability density function defined by the location and width of one of the endpoints. One finds the midpoint by Euclidean methods. Evidently this midpoint will turn out to be different any time you do this. After many trials you obtain a point cloud that approximates the probability density function that corresponds to the end result (a fuzzy point).

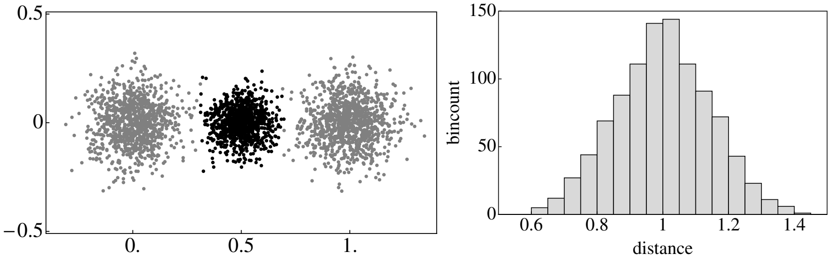


Fig. 4 The bisection operation. At left the simulation of bisection of two fuzzy points. At right a histogram of the distances between pairs of points from the simulation. The points are indicated in gray, the intersection in black. For the simulation we draw two points, one near$\{0,0\}$, the other near $\{1,0\}$ (the gray points) and find the bisection (black). The simulated bisections are in the general neighborhood of $\{0.5,0\}$. Apparently the distance between the fuzzy points is also fuzzy.

A similar method applies to a large variety of constructions (see figure 5). The results suggest that the geometry of the fuzzy plane is perhaps less trivial than one may expect. For instance, something like the "point of intersection" of two lines (each defined through a pair of points) is typically not a generic point. If the lines mutually subtend a narrow angle, the intersection is more similar to a line segment (see figure 6). In developing this geometry one needs to introduce geometrical

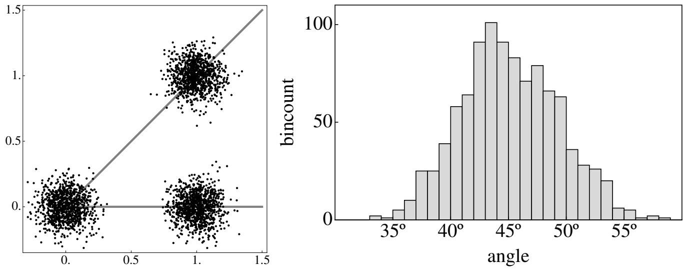


Fig. 5 The angle subtended by three fuzzy points $\{1,1\}$, $\{0,0\}$, and $\{1,0\}$. The lines indicate the corresponding Euclidean angle of 45°. At right the histogram of the angles subtended by triples from a simulation. Apparently the angle is fuzzy too (about $45° \pm 5°$).

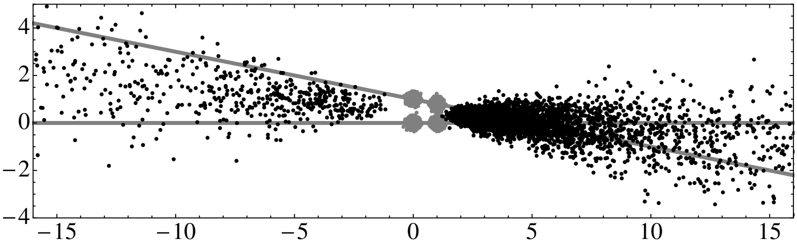


Fig. 6 The intersection of two lines. Each line is defined by a pair of fuzzy points (depicted in gray), the horizontal line through the points $\{0,0\}$, and $\{1,0\}$, the oblique line through $\{0,1\}$, and $\{1,0.8\}$. The Euclidean intersection is the point $\{5,0\}$. The corresponding Euclidean lines are drawn in gray. They mutually subtend a rather narrow angle. Notice that the simulated intersections can be *anywhere*, although they cluster about the Euclidean result. The cluster is very unlike a fuzzy point though. It is evidently elongated, and thus "line-like".

entities that have simultaneously point-like and line-like properties. Interesting as such observations are, we will not pursue them in this paper[11].

5 Atlas Structure: The Self-Similar Fuzzy Line and Plane

A notion of distance is necessarily complicated when points have different sizes. One way to grasp the essential idea is to consider a simple example, that we will refer to as the "atlas model".

Atlases contain maps of limited size, and of various scale. Thus Paris and Berlin are both on the map of continental Europe. The *Tour Eiffel* will be on a map of central Paris, the *Brandenburger Tor* on a map of central Berlin. These landmarks are not to be found on a single map though. The distance between the *Tour Eiffel* and the *Brandenburger Tor* is measured on the highest resolution map that (via many levels of indirection) contains them both. This map will have a rather low resolution. The distance is simply the distance Paris–Berlin on the map of continental Europe. The distance between the *Arc de Triomphe* and the *Brandenburger Tor* is the same as the distance between the *Tour Eiffel* and the *Brandenburger Tor*. Yet the *Tour Eiffel* and the *Arc de Triomphe* have a well defined distance as measured on the map of central Paris.

This induces a planar structure that is quite different from the familiar Euclidean plane.

The "location" in the spatial domain is defined hierarchically,[12] as familiar from daily life experience [69]. For instance, suppose you forgot a key, where would you look? Certainly in your home town, your neighborhood, your house, a certain room, a certain desk, a certain drawer, somewhere in the mess you probably expect there. If you had to specify the location over the phone it would depend on who you were speaking to (a stranger, your neighbor, or your spouse) where in the sequence you would start. Almost certainly you would describe some nested order though. The essential gain of such a description lies in the fact that you refer only to local structures. Doing this iteratively gets you by way of local methods (in the scale dimension) to "global" relations (in the space dimensions), though only indirectly so. This closely resembles the use of an atlas.

Such a formal structure appears fit to describe well known properties of the visual field. We discuss some of the major features here.

A map is defined by a (central) location, a scope, and a grain size. The location tells you that "this is a map of such-and-so", the scope tells you the "size" of the area that is covered by the map, and the grain size tells you the "resolution". Anything smaller than the resolution is either omitted, or represented with a conventional sign, that is a point–like entity without internal structure. In typical cases the ratio of the grain size to the scope is fixed throughout the atlas[13]. The inverse square of the ratio (for a planar map) is the "number of pixels", that is the number of independent entities represented in the map. In conventional geographical atlases it is large (a million say), in the visual field it is rather small, say ten to a hundred. This ratio is an important number that characterizes the atlas. We assume it will be fixed by neuroanatomical/physiological constraints[14]. For the sake of experimental phenomenology it is just a "constant of nature" descriptive of the human condition.

Modeling the atlas structure is straightforward. We represent maps just as we do points, with all the obvious consequences (thus maps overlap, have intersections, and may dominate each other).

A map contains points that intersect with it and have a width equal to the grain size of the map. A point is *located* on a map if it is close to a point of the map. Here "close" may be defined as having an overlap of at least some characteristic number, say one half. The precise magnitude of the number is not really important. When two points are both located on the same map, we may measure their distance on the map. In reporting it one mentions both the distance and the grain size, for instance 5 ± 2. Thus "distance" is indexed by resolution, one actually has a one-parameter family of distance functions. Two distances are different if they differ by more than the combined uncertainty (grain size)[15]. (See figures 7 and 8.)

Things start to be slightly interesting when one or perhaps both of the points are not on the map. Consider why a point might not be on a map. It may happen either because the point is *too large*, or because it is *too small*. These cases are categorically different. Europe is not on the city plan of Amsterdam: it is too *large*. Tietjerksteradeel (pronounce *Tytsjerksteradiel*) is not on the map of Europe because it is too *small*[16]. But Tietjerksteradeel *might* (through some conventional mark[17]) be indicated on the map of Europe, whereas it is evidently *impossible* to indicate Europe on the city plan of Amsterdam. This is not due to some lack of conventional signs.

A point may dominate a set of points belonging to a map, and this set may be a proper subset of the set of all points belonging to the map. In such a case the point is an "area" in terms of the map[18]. Then points of the atlas may be designated a distance from the area, for instance, the minimum distance of the given point in the atlas to any point of the area.

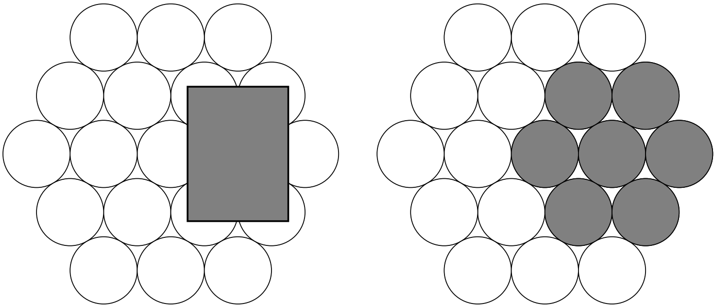


Fig. 7 The rectangular area at left is represented by the atlas area at right. Notice that the rectangular shape has become lost. It is retained in atlases of higher resolution, although these might only cover a corner.

If a point is too small, one needs to find a suitable "representative" in the atlas. Possible representatives of a point are points that dominate the point. If the points have representatives in the map, then these representatives define a small area in the map[19]. Any point of the area can be used to represent the point on the map.

A problem might be that the dominance relation may hardly be expected to be defined for points with extremely different width. The reason is that one cannot have a "punctate stimulation" that is effectively punctate for both, and simultaneously potent enough to excite both. In such cases the Helmholtz local sign mechanism has to break down. In practice there will be some vaguely defined limit on the ratio of widths. Thus points may be "really too small" or "really too large". In the case

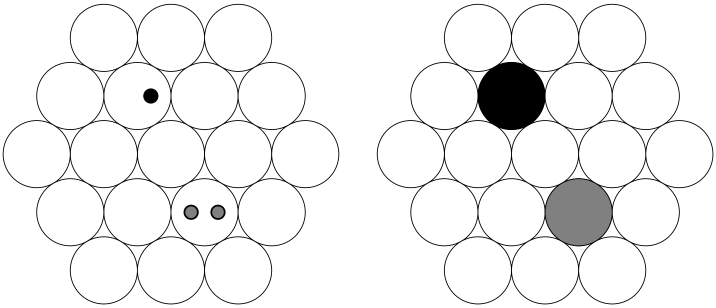


Fig. 8 The two gray dots (at left) have the same representative (right), whereas the black dot (left) has a different representative. Thus each of the gray dots has the same distance to the black dot in terms of this atlas. The mutual distance of the gray dots is zero, or rather 0 ± 1. Of coarse a finer grained atlas would “resolve” the gray dots, and assign them a finite distance.

of the visual field one cannot use the conventional methods of cartography. Any relation has to *exist*, that is to say, has to be relatable to prior experience, in order to be possibly meaningful[20].

From a formal point of view it is nice not to put arbitrary restrictions on atlas size. That is to say the point width and grain size could be infinitely small, and the scope of a map could be infinitely large. In real life one meets such restrictions of course. The visual field is about a hundred and eighty degrees in diameter, thus of finite extent, and the best resolution is about a minute of arc, thus not infinitesimally small.

Moreover, the resolution (smallest grain size) depends on the eccentricity, that is the distance to the center of the visual field. (See figure 9.) This leads to (important) complications that we have considered in some detail before [43]. It also forces the use of saccadic eye fixations, most of them involuntary, and evidently part of the micro genetic process. In this paper we have ignored this aspect, it is crucial to any understanding of the actual system.

When we ignore arbitrary constraints, the fuzzy plane, augmented with an atlas structure, is a self-similar entity ([54]). That is to say, if you scale all spatial dimensions by the same factor, you obtain an entity that is congruent to the original. That is why we refer to the “self-similar fuzzy plane”.

It is an ideal, but quite apt, formal description of the structure of the visual field. The self-similarity has a firm basis in psychophysical fact [79, 43, 40, 41, 42, 44, 76, 75, 4].

We propose that the formal structure may be fleshed out either in neurophysiology or in experimental phenomenology. These interpretations will be categorically different, of course.

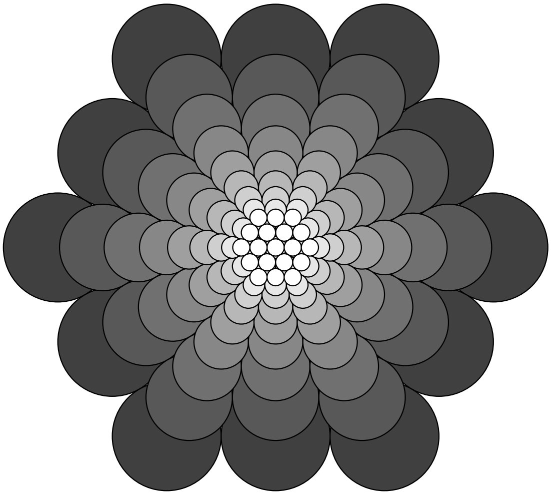


Fig. 9 A schematic model of the structure of the global visual field (illustration of a structure proposed by [43]). In reality there are many more cells, of course. One has a superposition of atlases of many sizes. Near the center the full range of sizes is present, near the periphery only coarser ones. Each atlas, also the coarser ones (represented in darker tint here) cover a convex part centered on the fovea. Thus, in this drawing, the finer atlases are drawn "on top" of the coarser ones.

5.1 The Self-similar Fuzzy Plane as a Sampling Structure in Physiology

In the physiological interpretation the fuzzy self-similar plane is essentially a *sampling structure*. (E.g., the structure shown in figure 9 has to be understood in that way.) That is to say, it is *embodied*, the elements are invariably present as neural structures. A "point" is to be thought of as an operator, an element that responds with a scalar variable (e.g., spike firing rate) when a retinal image is presented to the system [45]. The variable could be a local sample, with certain spatial uncertainty, of the retinal irradiance. In reality the simplest receptive fields, the embodiments of points, have a more intricate structure. The elementary sample is closer to a fuzzy, directional partial derivative of the retinal irradiance.

In this interpretation the self-similar fuzzy plane is part of an algorithmic structure, like a special purpose computer. It might implement bottom up inverse optics processes as mentioned above.

5.2 The Self-similar Fuzzy Plane as a Model of the Microgenesis of Visual Presentations

In the interpretation of experimental psychology the formal self-similar fuzzy plane is a virtual, potential entity. Its elements may actualize if there is awareness, *they are created with the awareness.*

Microgenesis generates presentations in legato fashion at a rate of roughly ten per second, it is a systolic process [7]. The awareness is spatially structured along the way of the formal structure. Not all elements need to be present. For instance, there will be no points in the awareness of an extent of blue sky.

The points are like the dots of paint of a painter[21], placed with a fine or a broad brush as the case may be. The points have no existence except as part of a map. The map has no existence except in terms of an atlas. Points as such rarely play a role in awareness, except perhaps in laboratory settings. The finest articulations of a presentation are still map-like. From a formal perspective the self-similar fuzzy plane might be described in terms of an infinite hierarchy of maps, with the "points" as formal limits that are never actually reached [83].

Most natural non-composite parts of awareness are like the touches or strokes applied by the painter. They will typically be areas in some atlas. Thus they have not only size and shape (in terms of the atlas), but also a resolution (characteristic of the atlas). Points will hardly ever occur as natural, non-composite parts.

A map may perhaps be likened to a small[22] thumbnail sketch. The atlas is then a collection of thumbnail sketches arranged in an order that is at least patch-wise hierarchical. Elements gain a *meaning*, or significance from their embedding in super-elements (the local context), whereas they themselves are *partial causes* of that super-element.

Thus the presentations (momentary visual awarenesses) have a tight structure of mutual dependencies. This structure is built from the bottom up, and grows through progressive diversification and pruning, like an evolutionary process. It is a living structure in the sense of having been constructed simultaneously with the presentation.

The microgenetic process generates "hallucinations" galore, and checks them against the optical structure at the sensitive body surface[23]. Hallucinations start off as very coarse "gist" [60]. Each systole of microgenesis generates numerous contenders, finally (when no further evolution is possible) one makes it into awareness. The losers in the evolution are like virtual alternative realities. Occasionally one believes to catch glimpses of them [29]. They may be likened to the *premières pensées* of the academic painter, thumbnail sketches that were eventually put aside. All these — even though put aside — were important in the genesis of the final work.

Diversification sets in as the current hallucination fails to account for the diversity encountered in the frond-end. The diversification is intentional. Thus, it is meaningful, but the meaning is more like a prior conviction.

Pruning occurs if the hallucination is more articulate, or differently articulated than the optical structure warrants. At such occasions the hallucination is "controlled", because contradicted by nature. At such an occasion the microgenetic process gains information in the sense of meaning (as opposed to mere structure). It is learning by mistake. In Erwin Schrödinger's (1887-1961) view this is where nature lights up to mind[24]. It is a spark of enlightenment [72]. This is how presentations can be meaningful and conducive to biological fitness. It is how one may imagine awareness to be generated, although of course not in the sense of causality conventionally adopted in the exact sciences.

Microgenesis freewheels in case the front-end is not filled with optical structure (eyes closed, darkness). In such cases one is aware of "visions", or dream images. Microgenesis often results in presentations that contain elements for which no immediate optical structures are present (e.g, Kanizsa's [31] amodal completions), or where certain optical structures are apparently ignored. Often large parts of the optical structure are summarily accounted for (the leaves of grass of the lawn say) as "texture". This is similar to the artist's use of hatching to fill areas[25].

In each systole microgenesis rebuilds the structure from scratch. Awareness is always *now*. The presentation is likely to be similar to the preceding one, although the evolutionary process may end up in a slightly different result, showing another page of the atlas so to speak, one that didn't develop before.

6 Metameric Images in the Fuzzy Self-Similar Plane

If locations in the visual field are only approximately determined, then local perturbations with amplitudes less than the uncertainty should not be detectable. If the field is indeed self-similar, then the permissable perturbation of the relative position of two points should depend upon their widths and mutual distance. Given an image, the set of all images that represent permissible perturbations should be an equivalence set with respect to visual awareness. We call it a "metamer", a term coined by Wilhelm Ostwald [62] (with a chemistry background, that is where the term "metamer" derives from), in the setting of colorimetry. In this section we explore such metameric images.

It is easy enough to prepare metameric images. First we implement the generation of perturbations at some fixed scale. Then we generate perturbations at many scales, and add them with the required weights. (See figure 10.)

The resulting images are interesting, since they look entirely "natural". (See figure 11.) Moreover, they are hard to distinguish in slightly eccentric fixation or at cursory glance. They are very similar to the metameric images described by Freeman [18], and used by Balas [3] to account for such perceptual effects as the "crowding" phenomenon [63].

The image at top left in figure 11 is the original one. It is also a member of the metamer. That it would ever come up in awareness has zero probability because the metamer is of infinite cardinality.

The crowding phenomenon (figure 12) may be understood as deriving from spatial uncertainty that is large with respect to the details to be resolved. It can be modeled in various ways [3, 18]. The present model works just as well.

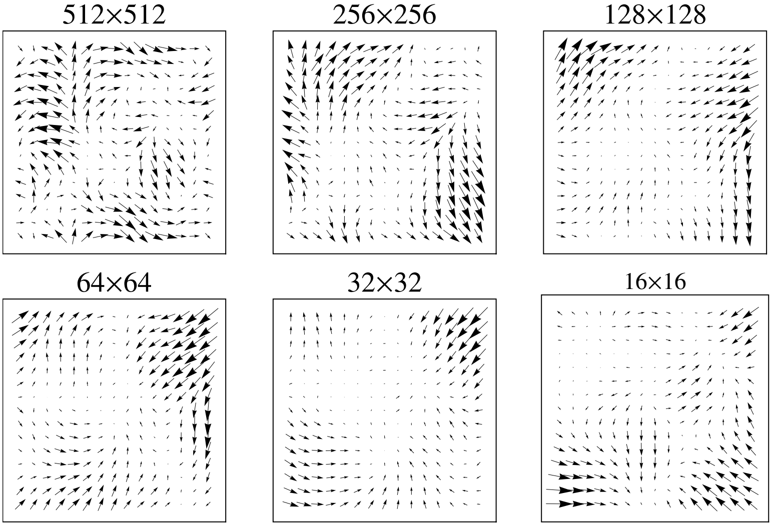


Fig. 10 A statistically self-similar displacement field. The figures are nested areas of different size. In each case the mean has been subtracted. Notice that there is statistically similar (random) structure on any scale. In the case of some arbitrary *smooth* field such a process of zooming in would eventually converge to a uniform, unidirectional field. You may see this watching a weather channel on TV. The wind pattern in your local area is always uniform, no matter how complicated on the continental scale.

7 The Pointless Order

A notion of "pointless order" is based upon the local sign concept proposed by Helmholtz. The essential idea it that correlation between activation in nerve fibers might be a cue that the receptive fields on the sensitive body surface at which the activation arose are spatially overlapping. Thus the correlation structure of the optic nerve induces overlap relations that may again be interpreted as a Čech homology [9, 32, 35]. The topology reveals that the visual field is two-fold extended, and so forth. It is a very powerful idea, much advanced over ideas based on anatomical somatotopy as are still commonly held today.

Helmholtz's notion may be generalized in various ways. Here we apply it to the Gestalt [56] mereotopology [83, 80]. A structure obtains a location within a superstructure on the basis of its parthood–relations. Here "part" is understood in a serial way, e.g., although a leaf is perhaps most naturally seen as part of a twig, we extend parthood to the tree (of which the twig is a part), the forest (of which the tree is a part), and so forth. Such a Gestalt mereotopology may be constructed over the geometry provided by the self-similar fuzzy plane.

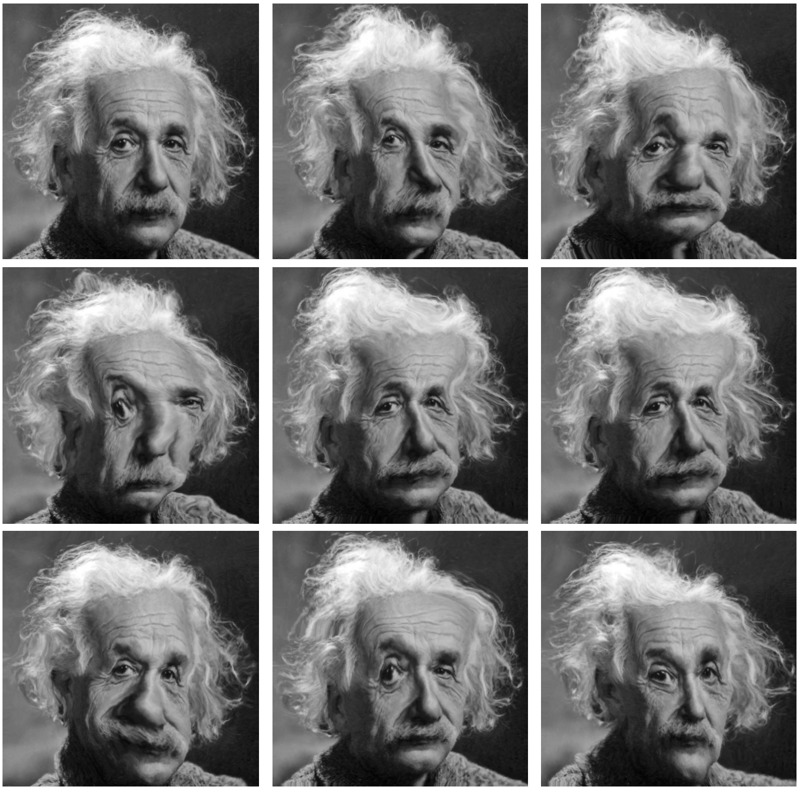


Fig. 11 A fiducial image is shown on top left. All other images are metameric copies obtained via self-similar displacement fields. Try to fixate to the side, you will notice that the metameric images become hard to distinguish from each other. The images may be described as "metamers of the cursory glance".

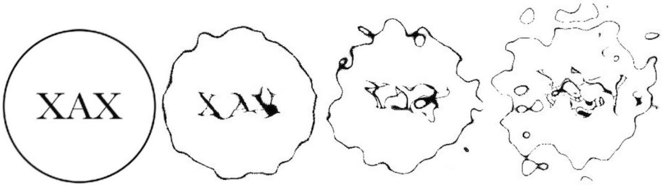


Fig. 12 The "crowding" phenomenon. Various amounts of disorder have been applied. At some point the letters start to intermingle, and apparently "crowd" each other.

Parthood induces a partial order of structures. The simplest local structures are illustrated in figure 13. Although the pointless order can be very intricate, and typically *is* very intricate in immediate visual awareness, only simple substructures are relevant at any time. One has to think of simple "puzzles", grouping of (much) less than a dozen [57] pieces.

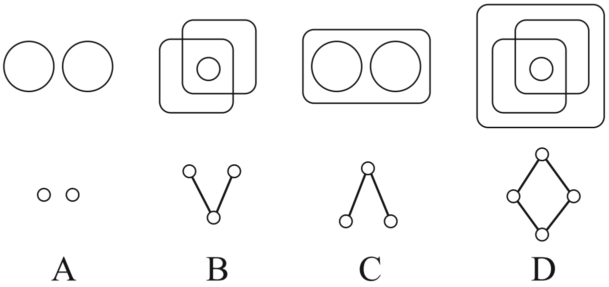


Fig. 13 Simple examples of the implicate order of two structures. At top we show the parthood relations as Venn diagrams [81], at bottom we draw the corresponding Hasse diagrams [5]. In A the two structures are unrelated. The Hasse diagram is not even connected. In B the two structures are *connected* through a third structure that is part of both, like a bag of marbles. In C the two structures are *related* through a third structure of which both are a part, like the fingers of a hand. In D the two structures are both connected and related. The Hasse diagram is a lattice.

In terms of the atlas simile, figure 13 A shows the relation between *Milwaukee* and *Paris* (none). Figure 13 B shows the connection between the *Via Appia* and the *Via Aurelia* (both contain Rome). Figure 13 C shows the relation between the *Arc de Triomphe* and the *Tour Eiffel* (both in Paris). Figure 13 D again shows the relation between the *Via Appia* and the *Via Aurelia*, now more detailed (both contain Rome and are contained in Italy). Notice that the structure depends on the view, e.g., whether you take Italy into account when you focus on Rome. This is entirely typical. The grouping process fluctuates both with respect to location and scope. A single map leads to numerous groupings, according to the current "set" [39].

In figure 14 we show how these notions apply to Gestalt formation by grouping. In figure 14 upper left one has two oblique lines, each formed through the grouping of seven collinear points. The two lines are mutually unrelated, one point doing double duty: different "copies" apparently belong to different lines. In figure 14 upper center this point has individuality. It is part of both lines, the lines are mutually *connected*. In figure 14 upper right this connectedness has been destroyed, this case is much like the one at upper left. In figure 14 center left the two lines are part of a cross, the cross being part of an "**OXO**" Gestalt. Here the lines are related through the superordinate **OXO** Gestalt, although they have lost much of their individuality. In figure 14 mid-center the lines are both connected and related. In figure 14 center right the cross has regained individuality because the **OXO** Gestalt is much weakened (the cross tends to be part of a weak "**OX**" Gestalt). In figure 14 bottom left the cross appears as a separate Gestalt, like in that on top left. This also happens at bottom center, where the two **O**'s group through "common fate", although the **X** might be part of a triangular Gestalt with **OO** as base, and the cross as vertex. In figure 14 bottom right the cross groups with the leftside **O** through proximity. It becomes part of an **OX** configuration, whereas at bottom center it retains its individuality. In microgenesis all such interpretations (and many more) are simultaneously entertained. Anything one does to the pattern (think of relative movements, for instance) will change their chances to pop up in immediate visual awareness.

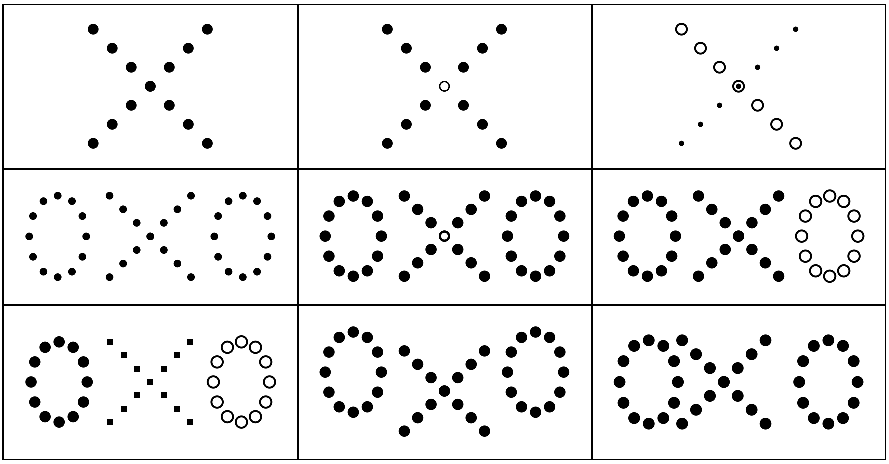


Fig. 14 Various cases of pointless order in grouping

In figure 15 we illustrate the extended meaning of "parthood". At top left one sees two circles and two disks. Each disk is seen to "belong" to one of the circles, due to spatial inclusion. There is a tendency to group all elements as "two eyes". This tendency is much reduced by adding another "eye" (top right), although there is a weak tendency to fluctuate between a pair of eyes at left or one at right. In figure 15 center right the eyes become part of a face. The outer circle relates the pair of eyes to the line segment. In the figure at center left the same is achieved through the addition of a piecewise straight stroke. The eyes become again part of a face, although not spatially included in it. In the figures at bottom the weak tendency to see eye pairs in a row of three circles with internal disks (top right) has been much strengthened through the addition of some elements. One sees two "faces", one eye doing double duty. The center eye tends to be part of either the left or the right face as one shifts attention, though it is also possible to see it as part of both faces by attending to the figure as a whole.

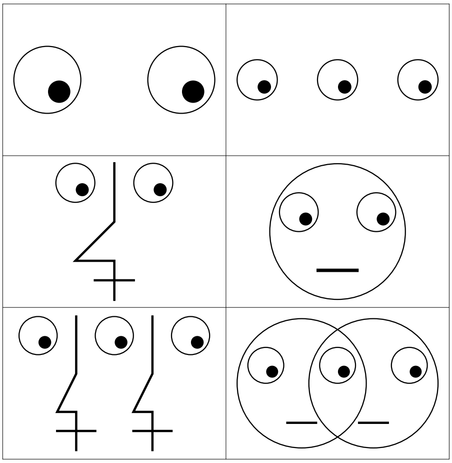


Fig. 15 Various cases of pointless order in grouping illustrating cases where "parthood" is not necessarily the same as spatial inclusion

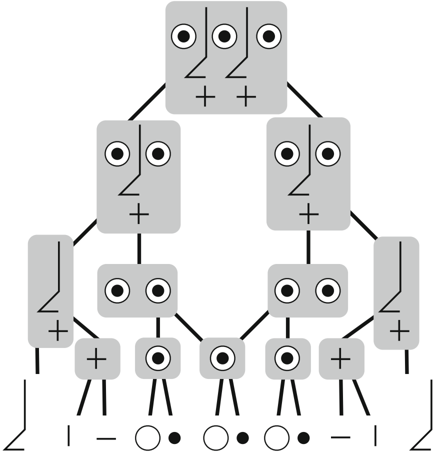


Fig. 16 A Hasse diagram of Gestalt relations. The "Janus face" is spontaneously seen as the merge of two single faces. Each face is seen as the grouping of two sub-Gestalts, these again can be broken down... Although one obtains glimpses of yet other sub-Gestalts (such as a row of "three eyes") these tend to be less salient. Notice that with 12 parts there exist 4094 proper subgroups, few of which are salient. Visual awareness is *very* selective.

The changes in Gestalt formation through grouping in those figures can be formally described as the formation or breaking of links in the Hasse diagrams that describe the partial order of parthood. An example is shown in figure 16. The super-Gestalt, a "Janus face", is built from twelve atomic glyphs. The set of all feasible groupings is the powerset, of cardinality 4095 (not counting the empty set). The full lattice of inclusion is huge (over half a million relations). The most salient Gestalts that feature in visual awareness are plotted in the Hasse diagram. It includes only a tiny subset of the formal possibilities. Most random groupings fail to be salient at all. The various "solutions" may be rated by a measure of Gestalt *Prägnanz*. A number of formal metrics are available [15, 12]. There will be numerous local maxima, and models of the microgenetic process should take all of these into account as potential contenders to the role of next presentation. Some artists play intentionally on this ever fluctuating ambiguity of visual microgenesis [65].

7.1 Awareness and Hallucination

In cases where the evolutionary process may equally well take two different routes, one often notices sudden flips of awareness. One presentation alternates with another. Well known instances are the necker-cube [58], and the duck-rabbit [30]. The alternation rates are slower than the systolic rate of microgenesis, suggesting that one important determinant of any presentation is the immediately preceding one.

In a "good look" which involves a few dozen presentations one thus "pages through the atlas". In this view the atlas exists over time, and is continually "reprinted" as a perhaps slightly different edition.

This account is a variety of *vision as controlled hallucination*. A common objection is that hallucination will have zero probability to be "veridical". From an evolutionary perspective "veridical perception"[26] should be replaced with "perception subserving fitness" though [74, 52, 69, 26]. In this setting fitness implies efficacious optically based behavior. This again implies that the hallucinations should sufficiently account for the optical structure impinging on the retinas, this is the "control" part of "controlled hallucination". Thus, the objections from veridicality against a "top-down" account of presentations are misplaced.

When the optical structure is lacking in relation to possible scenes, hallucination is hardly, and if so arbitrarily constrained. This happens when you view some random pattern for instance. You are aware of fleeting representations that are mainly of your own making. Such cases have been discussed in the art of painting (the polemic between Sandro Botticelli and Leonardo da Vinci[27]). Thus Sandro Botticelli would become aware of Olympic scenes or fantasy landscapes, Leonardo da Vinci of battles, a modern teenager perhaps of witches with machine guns. It is the basis of the Rorschach test [70]. Such presentations follow up in arbitrary order, sometimes one believes to just catch a glimpse of something, sometimes the presentations last for a glance. Similar phenomena have been described throughout the ages, in psychology starting with William James [29].

Another objection is that hallucinations have no natural way to get started. This is true if you think of perceptions as necessarily being *imposed* like Aristotle's notion of the wax receiving the form of the seal, otherwise a natural starting point would be the person's emotional core. It is not an issue in practice, because one important influence on the present systole of microgenesis is the previous one. Thus, in the large majority of cases, the microgenetic process is partly constrained by the previous one from the earliest stage. In (very singular) cases of a "scene cut" (say you are blindfolded, magically transported to the Sahara and the blindfold is suddenly removed[28]), you need to start from scratch.

No doubt bottom-up processes must play a role then. They are fast and automatic, even partly protopathic. Because of that they cannot be involved in awareness[29], which is all meaning and quality. However such processes may provide a "gist" that influences the initial microgenesis, much like the patterns that started off Sandro Botticelli as he threw a paint-soaked sponge against the wall and contemplated the ensuing patterns. Such processes may well go on at all levels, generating fleeting structures that — although in themselves meaningless — are like proto-objects, and when picked up by microgenesis become seeds that decisively influence the course of diversification.

Awareness is likely to converge very soon on something conducive to increased fitness. This is because microgenesis is likely to win any reasonable game of "twenty questions" against nature [38].

8 Conclusions

The visual field is a two-fold extended entity. Its structure is very different from that of the familiar Euclidean plane though. Differences exist on many levels, from local to global, and in projective and metrical properties.

It is perhaps surprising that one may discuss the formal structure of the visual field at all. After all the visual field is not like a pre-existing canvas upon which visual awareness draws colored patches. The visual field is a living entity that is generated along with the awareness. Microgenesis creates presentations in roughly a tenth of a second. As is well known [2, 49, 50] the temporal and spatial order of the presentations may be different from that of the optical structure delivered to the eye. The awareness tends to be more coherent than the optical structure when the latter is artificially put in local disarray [49, 50]. Thus, it is not that the visual field may be regarded as a "container" with well defined structure, in the Newtonian sense [59]. It is more like a Leibnitzian system of relations that only exists relative to the *relata* [10].

From a neurophysiological viewpoint the awareness is somehow related to (though one has not the faintest notion how the ontological gap might be bridged) the combination of motor commands, present state, and neural nerve activity. From a phenomenological perspective visual awareness simply happens. In experimental phenomenology one notices relations with situational awareness, emotional state, current activity, and the optical structure present at the eyes. The formal account offered here may find application in both neurophysiology and experimental phenomenology, though obviously in (very) different ways.

When applying these ideas to neurophysiology one has to deal with a huge variety of boundary conditions that are quite alien to the ideal, formal account. Nothing can be continuous, thus one has to discretize, nothing can be arbitrarily small or large, thus one has to introduce numerous arbitrary (from the ideal perspective) parameters, and so forth. Moreover, in animal physiology one easily extends the studies of the system to situations that are not in the normal range of the free living animal. Sometimes the constraints are so pressing that they force structures that are perhaps in doubtful taste from an ideal point of view. The upside is that one may obtain models that are really interesting because of their capability to describe actual brain structures.

Our own interest is mainly experimental phenomenology. Here one meets with completely different constraints. In describing the phenomena one prefers ideal, formal systems that happen to fit the phenomena. In practice the level of detailing in experimental phenomenology is restricted, moreover the range of situations is necessarily restricted to the normal human environment. Thus, it is perhaps less surprising that one often finds simple formal descriptions to apply really well, even in a quantitative sense [48].

The self-similar fuzzy plane is such a simple formal system. In its simplest form, as described in this paper, it already accounts for a large variety of phenomenological facts[30]. One extension that is soon required is the non-uniform nature of the visual field, with its minimum grain size increasing gradually from center to periphery [40, 41, 42]. Such an extension requires one to take account of involuntary fixational saccades. To construct such a description should be straightforward. From a formal perspective the eye-movements are just another mode of addressing the atlas, a mere implementation issue[31].

Another complication is the nature of the points. In the present account we modeled points in terms of gaussian weights. Such a point should be understood as an *operator*. It operates on the retinal illuminance pattern, and it yields a sample of the irradiance at a given location, with a given resolution. The sample is a scalar value, representative for the magnitude of the local illuminance. This is a simplification in various ways. One is the lack of spectral resolution. This is easily enough taken care of through the introduction of points with different spectral sensitivities. (Red, green and blue points, say.)

A more important issue is the structure of a point. The gaussian point is indeed a “point” in the sense of Euclid’s “that which has no parts”. It makes the fuzzy line formally into a scale space. It is well known that the neurophysiology of the peripheral visual system implies that points have considerable structure. A simple model constructs the point as a jet space [13], up to about the fourth order [45, 36, 37, 46]. A jet contains all fuzzy spatial derivatives, thus the point actually represents the local illuminance structure in terms of a truncated Taylor series. This has important consequences. For instance, the Helmholtz local sign mechanism now has much more “to work on”. One regards the correlation matrix for the full jet instead of just a single number. Such studies have already yielded important insights [21, 18], with immediate applications to experimental phenomenology [3] and theoretical neurophysiology [66].

Working out the structure of such an "augmented fuzzy self-similar plane" will be a major, though most worthwile undertaking.

As a final issue we mention the use we repeatedly made of a very general principle, that might easily be overlooked as essentially trivial. It is that *nothing may enter awareness that cannot be ultimately traced to prior awareness*[32]. In a sense awareness is a cross-section of recapitulated experience. The notion has been formulated explicitly by von Uexküll (1864–1944) [77]. It is a principle implicitly used by Lotze, Helmholtz and Platt in their understanding of local sign. This principle is often ignored, or violated in main stream research. For instance, this mistake is at the basis of such common notions as the importance of somatotopy to the awareness of a coherent two-dimensional visual field, the contemporary neglect of the concept of local sign, or the reliance on feature detectors. Although the principle is very general, we have used it to put strong constraints on the formal structure of the visual field. In our framing it does not apply to sensory motor reflexes and automatic behavior. For instance, freshly hatched chickens show remarkable optically guided behavior[33] [78]. To hold that this would be accompanied by a keen visual awareness would violate the principle as stated. Far reaching implications of the principle remain to be developed.

Notes

[1] "Fuzzy" is a convenient term although our discussion does not fully conform to classical "fuzzy set theory" [84].

[2] The construction of the Euclidean plane with circular disks replacing the points (Huntington's [27]'s original idea of points as "one inch spheres") superficially looks like it might describe the *Empfingdungskreisen*. However, the disks are located with infinite precision, thus this is not a "fuzzy plane" at all.

[3] The weight is defined as $g(u;x,\sigma) = \exp(-(u-x)^2/2\sigma^2)/\sqrt{2\pi}\sigma$; x the location, σ the width parameter. Thus the fuzzy line is closely related to "scale space" [34], [16]

[4] Strangely, Euclid defines points twice (the second definition being DEFINITION II: "*The extremities of lines are points*") and makes no further use of the definition(s), never referring to them in the remainder of the text. Perhaps the definitions were added later as an afterthought.

[5] First define the symmetric binary relation (g_1, g_2) as $\int_{-\infty}^{+\infty} g(u;x_1,\sigma_1)g(u;x_2,\sigma_2)\,\mathrm{d}u$, and let $|g|^2 = (g,g)$.
Then the correlation can be defined as $C(x_1,\sigma_1;x_2,\sigma_2) = (g_1,g_2)/|g_1||g_2|$.

[6] The support of $g(u;x,\sigma)$ is $\{x-\sigma, x+\sigma\}$.

[7] A certain overlap specifies either spatial separation, or a different width. Moreover, it remains undecided which of the points is the larger. We have not been able to find a way to decide on this on the basis of sampled retinal inputs.

[8] Very roughly the length of the yardstick divided by the width of the points is the maximum number of "reasonable" iterations.

[9] The squared distance of a point with parameters $\{x+\mathrm{d}x, \sigma+\mathrm{d}\sigma\}$ from a fiducial point $\{x,\sigma\}$ (where $\mathrm{d}x$ and $\mathrm{d}\sigma$ are supposed "infinitesimal" is $\mathrm{d}s^2 \propto (\mathrm{d}x^2 + \mathrm{d}\sigma^2)/\sigma^2$).

[10] This is evident from the metric mentioned in the previous note. Interesting as this is from a formal viewpoint, we will not use this fact further in the paper. It may well have application in scale space theory [34], [16], and is most attractive from a formal perspective.

[11] Hjelmslev [24, 25] was one of the first mathematicians to consider such problems.

[12] Notice that such a structure greatly reduces computational complexity. The recursive subdivision of a domain of N elements reduces search times from being proportional to N to being proportional to $\log N$ [11].

[13] In conventional atlases the page size and either the printing technique or your visual acuity at normal reading distance determine this ratio.

[14] There may actually be physical bounds on the grain size due to the structure of electromagnetic radiation. We ignore such (and several other) complications. They are not immediately relevant to the discussion, and fairly obvious when one has to deal with them. Such possible complications will not give rise to surprises.

[15] The grain size to use when the two grain sizes are s_1 and s_2 (say) is $\sqrt{s_1^2+s_2^2}$, this is just Gauss's law of error propagation [19].

[16] Tietjerksteradeel counted 32.172 inhabitants on april first 2011.

[17] In cartography this is known as "generalization" [28].

[18] For instance "Central Park" is an area on maps of central New York.

[19] Of diameter 0 ± 1.

[20] It is basic to require that anything that enters awareness has somehow to be related to prior experience. Notice that this general principle has very important consequences for the possible structure of the visual field. The concept is due to von Uexküll [77].

[21] This does not imply that we commit ourselves to a naïve "picture in the brain" theory.

[22] "Small" in the sense of not highly structured. The actual "size" being irrelevant in a self-similar structure. Judging from psychological data [57] it is unlikely that the maps will be highly structured.

[23] With "sensitive body surface" one may mean a variety of things. For vision the retina would be the outermost surface. However, the neural process that accompanies microgenesis most likely address one or more cortical layers. Here the optical structure — still meaningless and uninterpreted — is stored in a volatile buffer that is continually overwritten. It has been cleared of a priori meaningless structure, and is made available in a convenient format. From our perspective we need not differentiate between the scene, the retinal irradiance, the primary visual cortex, ... They are just some of the multiple levels of indirection involved.

[24] Schrödinger's suggestion is the only "bridging hypothesis" known to us that connects the mental to the physical in a way that makes at least intuitive sense and does not insult our rational understanding. It is not a causal connection of course [38]. That would be nonsensical, and was not Schrödinger's intention.

[25] The hatching has no relation to the optical structure for the area. It is either arbitrary (e.g., mutually parallel oblique strokes), or symbolic, perhaps suggesting "foliage", and so forth.

[26] The notion of "veridicality" is meaningless on many counts [38]. However, we can hardly discuss this in the present paper.

[27] Perhaps starting with the account by Pliny [68] on painting practices by Apelles

[28] Such cases are rare and singular, though popular examples in philosophical discussions tend to focus on them.

[29] This is another application of Schrödinger's principle.

[30] Notice that we say "phenomenological", rather than "psychophysical". Reason is that awareness implies first person accounts, whereas psychophysics relies (by design) on third person accounts, or, perhaps better, on "intersubjectivity". Much of psychophysics is really

"non-invasive physiology". Saying that, it is nevertheless the case that the present formalism may be applied in many cases to describe psychophysical findings.

[31] This is similar to having a variety of memories in a computer, say a disk, RAM banks, and caches on the processor chip. From a formal viewpoint there is no need to differentiate between those.

[32] The principle is, of course, connected to Schrödinger's hypothesis. A spark of awareness is due to a contradicted expectation. But expectations must be due to prior experience.

[33] The newly hatched chicken can have no expectations. Its actions are automatic. Applying Schrödinger's principle this means that the chicken is not aware. It is a true zombie. For instance, a newly hatched chicken will take the first thing in its ken for "the mother hen". It even does this if it happens to be the cat. To a zombie that makes no difference, of course.

References

1. Ahissar, E., Arieli, A.: Figuring space by time. Neuron 32, 185–201 (2001)
2. Albertazzi, L.: The Time of Presentness. A Chapter in Positivistic and Descriptive Psychology. Axiomathes 10, 49–74 (1999)
3. Balas, B., Nakano, L., Rosenholtz, R.: A summary-statistic representation in peripheral vision explains visual crowding. Journal of Vision 9, 1–18 (2009)
4. Bijl, P., Koenderink, J.J., Toet, A.: Visibility of blobs with a gaussian luminance profile. Vision Research 29, 447–456 (1989)
5. Birkhoff, G.: Lattice Theory. American Mathematical Society, Washington, DC (1948)
6. Braitenberg, V.: Vehicles: Experiments in synthetic psychology. MIT Press, Cambridge (1984)
7. Brown, J.W.: Self-embodying mind: Process, brain dynamics and the conscious present. Barrytown/Station Hill Press, Barrytown, NY (2002)
8. Buckley, J.J., Eslami, E.: Fuzzy plane geometry I: Points and lines. Fuzzy Sets and Systems 86, 179–187 (1997)
9. Čech, E.: Théorie générale de lhomologie dans un espace quelconque. Fundamenta Mathematicæ 19, 149–183 (1932)
10. Clarke, S.: A Collection of Papers, which passed between the late Learned Mr Leibnitz, and Dr Clarke, In the Years 1715 and 1716, by Samuel Clarke D D. James Knapton, London (1717)
11. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to algorithms, 3rd edn. MIT Press, Cambridge (2009)
12. Daskalakis, C., Karp, R.M., Mossel, E., Riesenfeld, S., Verbin, E.: Sorting and Selection in Posets. In: Mathieu, C. (ed.) Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, New York, pp. 392–401 (January 2009)
13. Ehresmann, C.: Introduction à la théorie des structures infinitésimales et des pseudo-groupes de Lie, pp. 97–127. Geometrie Differentielle, Colloq Inter du Centre Nat de la Recherche Scientifique, Strasbourg (1953)
14. Heath, T.L.: (ca 300 BCE) The Thirteen Books of Euclid's Elements Translation and commentaries, vol. 3. Dover Publications, New York (1956)
15. Fagin, R., Kumar, R., Mahdian, M., Sivakumar, D., Vee, E.: Comparing partial rankings. SIAM J. Discrete Math. 20, 628–648 (2006)
16. Florack, L.M.J.: Image Structure. Kluwer Academic Publishers, Dordrecht (1997)
17. Forsyth, D.A., Ponce, J.: Computer Vision: A Modern Approach. Prentice Hall, Upper Saddle River (2003)

18. Freeman, J., Simoncelli, E.P.: Metamers of the ventral stream. Nature Neuroscience 9, 1195–1201 (2011)
19. Gauss, C.F.: Theoria motvs corporvm coelestivm in sectionibvs conicis Solem ambientivm (Theory of the motion of the heavenly bodies moving about the Sun in conic sections) Davis, C.H. (transl). Little, Brown and Company, Boston (1809)
20. Graham, C.H. (ed.): Vision and Visual Perception. John Wiley and Sons, New York (1965)
21. Griffin, L.D.: The 2nd order local-image-structure solid. IEEE Trans. Pattern Anal. Mach. Intell. 29, 1355–1366 (2007)
22. Hess, R.F.: Developmental sensory impairment, amblyopia or tarachopia. Human Neurobiol. 1, 17–29 (1982)
23. von Hildebrand, A.: Das Problem der Form in der bildenden Kunst. Heitz, Strassburg (1893)
24. Hjelmslev, J.T.: Die Geometrie der Wirklichkeit. Acta Math. 40, 35–66 (1916)
25. Hjelmslev, J.T.: Die natürliche Geometrie Abh. Math. Sem. Univ. Hamburg 2 (1923)
26. Hoffman, D.D.: Sensory experiences as cryptic symbols of a multi-modal user interface. In: Bauer, M., Liptay, F., Marschall, S. (eds.) Kunst und Kognition, pp. 261–279. Wilhelm Fink, Munich (2008)
27. Huntington, E.V.: A set of postulates for abstract geometry, expressed in terms of the simple relation of inclusion. Mathematische Annalen 73, 522–559 (1913)
28. Imhof, E.: Kartographische Geländedarstellung. Walter de Gruyter, Berlin (1965)
29. James, W.: The Principles of Psychology, vol. 2. Dover Publications, New York (1890)
30. Jastrow, J.: The mind's eye. Popular Science Monthly 54, 299–312 (1899)
31. Kanizsa, G.: Grammatica del vedere. Saggi su percezione e Gestalt. Il Mulino, Bologna (1997)
32. Koenderink, J.J.: Simultaneous order in nervous nets from a functional standpoint. Biol. Cybern. 50, 35–41 (1984a)
33. Koenderink, J.J.: Geometrical structures determined by the functional order in nervous nets. Biol. Cybern. 50, 43–50 (1984b)
34. Koenderink, J.J.: The structure of images. Biol. Cybern. 50, 363–370 (1984c)
35. Koenderink, J.J.: The concept of local sign. In: van Doorn, A.J., van de Grind, W.A., Koenderink, J.J. (eds.) Limits in Perception. VNU Science Press, Utrecht (1984d)
36. Koenderink, J.J.: Operational significance of receptive field assemblies. Biol. Cybern. 58, 163–171 (1988)
37. Koenderink, J.J.: The brain a geometry engine. Psychological Res. 52, 122–127 (1990)
38. Koenderink, J.J.: Vision and information. In: Albertazzi, L., Tonder, G.J., van, V.D. (eds.) Perception Beyond Inference: The Information Content of Visual Processes. MIT Press, Cambridge (2011)
39. Koenderink, J.J.: Gestalts and Pictorial Worlds. Gestalt Theory 33, 289–324 (2011)
40. Koenderink, J.J., Bouman, M.A., Bueno de Mesquita, A.E., Slappendel, S.: Perimetry of contrast detection thresholds of moving spatial sine wave patterns. I. The near peripheral visual field (eccentricity 0-8 degrees). J. Opt. Soc. Am. 68, 845–849 (1978)
41. Koenderink, J.J., Bouman, M.A., Bueno de Mesquita, A.E., Slappendel, S.: Perimetry of contrast detection thresholds of moving spatial sine wave patterns. II. The far peripheral visual field (eccentricity 0-50 degrees). J. Opt. Soc. Am. 68, 850–854 (1978b)
42. Koenderink, J.J., Bouman, M.A., Bueno de Mesquita, A.E., Slappendel, S.: Perimetry of contrast detection thresholds of moving spatial sine wave patterns. III. The target extent as a sensitivity controlling parameter. J. Opt. Soc. Am. 68, 854–860 (1978c)
43. Koenderink, J.J., van Doorn, A.J.: Visual detection of spatial contrast: influence of location in the visual field, target extent, and illuminance level. Biological Cybernetics 30, 157–167 (1978)

44. Koenderink, J.J., van Doorn, A.J.: Invariant features of contrast detection: an explanation in terms of self-similar detector arrays. J. Opt. Soc. Am. 72, 83–87 (1982)
45. Koenderink, J.J., van Doorn, A.J.: Representation of local geometry in the visual system. Biol. Cybern. 55, 367–375 (1987)
46. Koenderink, J.J., van Doorn, A.J.: Generic neighborhood operators. IEEE PAMI 14, 597–605 (1992)
47. Koenderink, J.J., van Doorn, A.J.: Pictorial space. In: Hecht, H., Schwartz, R., Atherton, M. (eds.) Looking Into Pictures; an Interdisciplinary Approach to Pictorial Space, Cambridge, MA, pp. 239–299 (2003)
48. Koenderink, J.J., van Doorn, A.J., Wagemans, J.: Depth. i–Perception 2, 541–564 (2011)
49. Koenderink, J.J., Richards, W.A., van Doorn, A.J.: Blow-up: a free lunch? i–Perception 3, 141–145 (2012)
50. Koenderink, J.J., Richards, W.A., van Doorn, A.J.: Spacetime disarray & visual awareness. i–Perception 3, 159–165 (2012)
51. da Vinci, L.: Codex Urbinas. Gathered together by Francesco Melzi before 1542. First printed in French and Italian as Trattato della pittura by Raffaelo du Fresne in 1651 (1452-1519)
52. Lorenz, K.: Die Rückseite des Spiegels. Piper Verlag, München (1973)
53. Lotze, H.: Medicinische Psychologie oder Physiologie der Seele. Weidmannsche Buchhandlung, Leipzig (1852)
54. Mandelbrot, B.: How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension. Science, New Series 156(3775), 636–638 (1967)
55. Marr, D.: Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. Freeman, New York (1982)
56. Metzger, W.: Gesetze des Sehens. Verlag Waldemar Kramer, Frankfurt (1975)
57. Miller, G.A.: The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological Review 63, 343–355 (1956)
58. Necker, L.A.: Observations on some remarkable optical phænomena seen in Switzerland; and on an optical phænomenon which occurs on viewing a figure of a crystal or geometrical solid. London and Edinburgh Philosophical Magazine and Journal of Science 1, 329–337 (1832)
59. Newton, I.: Philosopiæ Naturalis Principia Mathematica. Josephi Streater, London (1687)
60. Oliva, A.: Gist of the scene. In: Itti, L., Rees, G., Tsotsos, J.K. (eds.) Encyclopedia of Neurobiology of Attention, pp. 251–256. Elsevier, San Diego (2005)
61. Orban, G.A., Van Essen, D., Vanduffel, W.: Comparative mapping of higher visual areas in monkeys and humans. Trends in Cognitive Sciences 8, 315–324 (2004)
62. Ostwald, W.: Einführung in der Farbenlehre. Verlag von Philipp Reclam jun, Leipzig (1919)
63. Pelli, D.G., Farell, B., Moore, D.C.: The remarkable inefficiency of word recognition. Nature 423, 752–756 (2003)
64. Palmer, S.E.: Vision Science: Photons to Phenomenology. MIT Press, Cambridge (1999)
65. Pepperell, R.: Seeing without objects: Visual indeterminacy and art. Leonardo 30, 394–400 (2006)
66. Petitot, J.: Neurogéométrie de la vision - Modles mathèmatiques et physiques des architectures fonctionnelles, Ecole Polytechniques editions, Paris (2009)
67. Platt, J.R.: How we see straight lines. Scientific American 202, 121–129 (1960)
68. Pliny the Elder (77–79CE) Historia Naturalis
69. Riedl, R.: Biologie der Erkenntnis: Die stammesgeschichtlichen Grundlagen der Vernunft. Parey, Berlin (1980)

70. Rorschach, H.: Rorschach Test – Psychodiagnostic Plates. Hogrefe Publishing Corp., Cambridge (1927)
71. Sansom, S.N., Livesey, F.J.: Gradients in the Brain: The Control of the Development of Form and Function in the Cerebral Cortex. Cold Spring Harb. Perspect. Biol. 1, a002519 (2009)
72. Schrödinger, E.: Mind and Matter. Cambridge University Press, Cambridge (1958)
73. Smythies, J.: A note on the concept of the visual field in neurology, psychology, and visual neuroscience. Perception 25, 369–371 (1996)
74. Tinbergen, N.: The study of instinct. Oxford Clarendon Press, London (1951)
75. Toet, A., Koenderink, J.J.: Differential spatial displacement discrimination thresholds for gabor patches. Vison Research 28, 133–143 (1988)
76. Toet, A., van Eekhout, P., Simons, H.L.J.J., Koenderink, J.J.: Scale invariant features of differential spatial displacement discrimination. Vison Research 27, 441–451 (1987)
77. von Uexküll, J.: Streifzüge durch die Umwelten von Tieren und Menschen: Ein Bilderbuch unsichtbarer Welten. J Springer (mit Kriszat G), Berlin (1934)
78. Vallortigara, G., Regolin, L., Chiandetti, C., Rugani, R.: Rudiments of mind: Insights through the chick model on number and space cognition in animals. omparative Cognition & Behavior Reviews 5, 78–99 (2010)
79. van Doorn, A.J., Koenderink, J.J., Bouman, M.A.: The influence of retinal inhomogeneity on the perception of spatial patterns. Kybernetik 10, 223–230 (1971)
80. Varzi, A.C.: Parts, wholes, and part-whole relations: the prospects of mereotopology. Data and Knowledge Engineering 20, 259–286 (1996)
81. Venn, J.: On the Diagrammatic and Mechanical Representation of Propositions and Reasonings. Philosophical Magazine and Journal of Science Series 5 10, 59 (1880)
82. Weber, E.H.: Die Lehre vom Tastsinne und Gemeingefühle. Verlag von Friedrich Vieweg und Sohn, Braunschweig (1851)
83. Whitehead, A.N.: Process and Reality: An Essay in Cosmology. In: Griffin, D.R., Sherburne, D.W. (eds.) Free Press, New York (1929) (1979 corrected edition)
84. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)

A Primer for Colour Computer Vision

Graham D. Finlayson

Abstract. Still, much of computer vision is predicated on greyscale imagery. There are good reasons for this. For much of the development of computer vision greyscale images were all that was available and so techniques were developed for that medium. Equally, if a problem can be solved in greyscale - and many can be - then the added complexity of starting with 3 image planes as oppose to 1 is not needed. But, truthfully, colour is not used ubiquitously as there are some important concepts that need to be understood if colour is to be used correctly. In this chapter I summarise the basic model of colour image formation which teaches that the colours recorded by a camera depend equally on the colour of the prevailing light and the colour of objects in the scene. Building on this, some of the fundamental ideas of colorimetry are discussed in the context of colour correction: the process whereby acquired camera RGBs are mapped to the actual RGBs used to drive a display. Then, we discuss how we can remove colour bias due to illumination. Two methods are presented: we can solve for the colour of the light (colour constancy) or remove it through algebraic manipulation (illuminant invariance). Either approach is necessary if colour is to be used as a descriptor for problems such as recognition and tracking. The chapter also touches on aspects of human perception.

1 Colour Image Formation

The visible spectrum occupies a very small part of the electromagnetic spectrum. For humans and cameras the visible spectrum lies approximately between 400 and 700 Nanometres[27] (see Figure 1).

School of Computing Sciences,
university of East Anglia,
Norwich
NR4 7TJ
United Kindom
graham@cmp.uea.ac.uk

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_2, © Springer-Verlag Berlin Heidelberg 2014

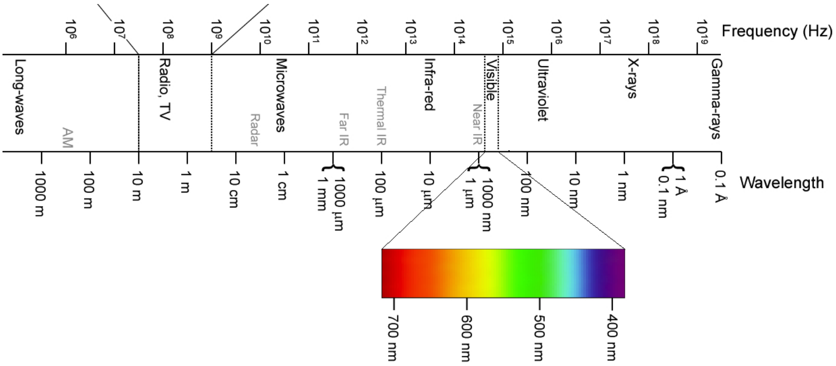


Fig. 1 The Visible Spectrum (Image taken from http://en.wikipedia.org/wiki/Electromagnetic_spectrum)

The spectral power distribution illuminating a scene is denoted $E(\lambda)$. The light strikes an object with surface spectral reflectance $S(\lambda)$ and the light reflected is proportional to the multiplication of the two functions (this product is sometimes called the colour signal). The light is then sampled by a sensor with a spectral sensitivity $R(\lambda)$. The various spectral quantities are shown in Figure 2. The integrated response of a sensor to light and surface is calculated in (1).

$$\rho_k^{E,S} = \int_\omega R_k(\lambda)E(\lambda)S(\lambda)d\lambda \quad k \in \{R,G,B\} \tag{1}$$

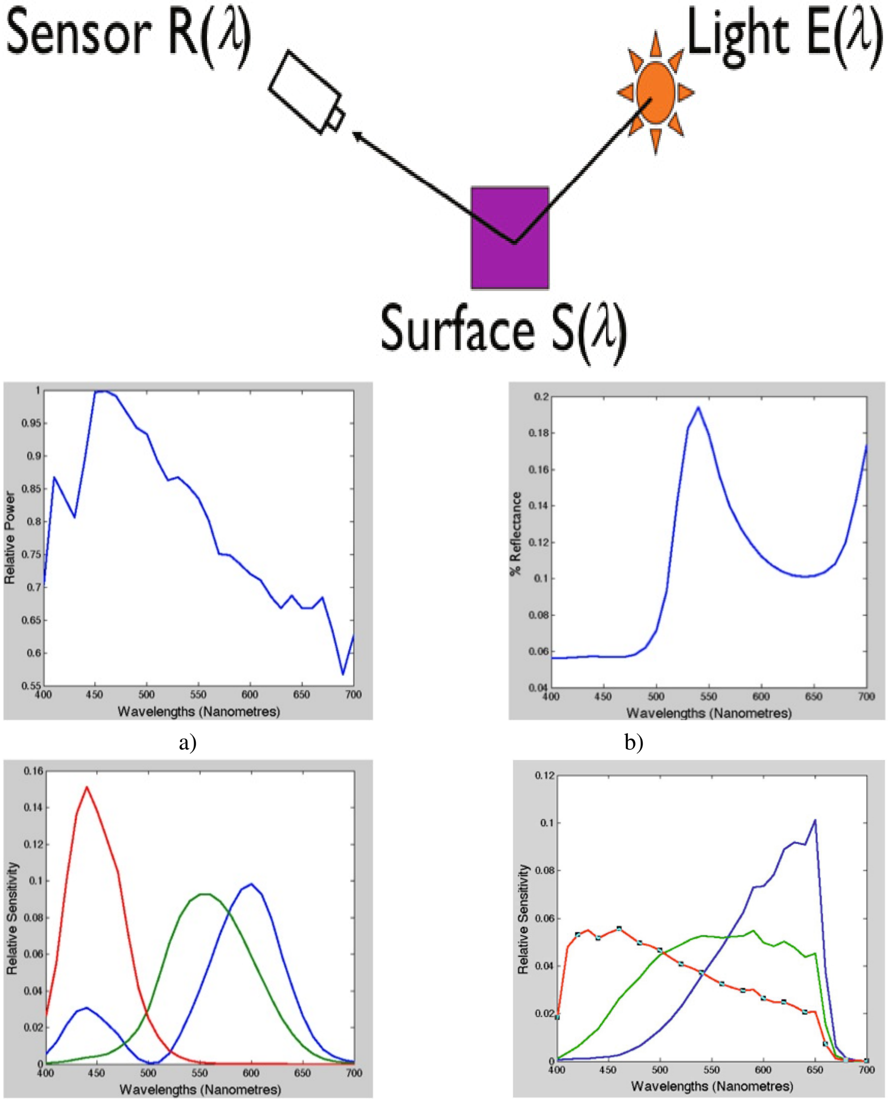
Where ω denotes the visible spectrum. Immediately, we see that light and surface play, mathematically, the same symmetric role. Each is as important as the other in driving image formation.

Notice that (1) includes no information about either the location of the light sources or the location of the viewer. This is because (1) is an accurate model only for the matte - or Lambertian - aspect of reflectances. Lambertian surfaces scatter the incoming light in all directions equally and they appear to have the same colour viewed from any position[15].

1.1 Colour Correction

Suppose we take a picture with a camera and then we wish to display it on a colour monitor. The raw acquired image cannot be used to directly drive a display. Rather, the image is transformed to a format suitable for display through a process called colour correction.

To understand colour correction, let us assume that the camera samples light like we do (for the purposes of this example, let us assume the camera curves equal those shown in 2c). How then do we transform the RGBs a camera measures to those that



a)

b)

c)

d)

Fig. 2 Middle left shows the spectrum of a bluish light (power concentrated in the shorter wavelengths). The reflectance spectrum of a dark green surface is shown in 2b). Bottom left (2c) we show the XYZ colour matching functions. These are not the sensitivities of an actual camera rather they are reference curves useful for the standard communication of colour[27, 17]. Lastly, in 2d) we show the curves for a commercial camera (Sigms SD9). Notice how differently they sample light compared with the XYZ functions.

drive a display to arrive at the reproduction we seek (i.e. a displayed image that looks like the scene we took a picture of).

To answer this question let us assume that a monitor has 3 colour outputs with spectral power distributions in the short (or blue), medium (green) and long (red) parts of the visible spectrum. The camera response to each display colour is written as:

$$\rho_q^p = \int_\omega R_k(\lambda) P_q(\lambda) d\lambda \quad k \in \{R, G, B\} \quad q \in \{l, m, s\} \tag{2}$$

in (2) $P_q(\lambda)$ denotes the spectral output of the three channels of a colour display. Notice that both equations (1) and (2) are linear systems (double the light double the response). The import of this here is that the response of the camera red sensor to the long and medium display outputs turned on together - e.g. at 50% and 75% intensities - is simply the sum of the individual responses:

$$\begin{array}{l}\rho_k^{0.5l+0.75m} = \int_\omega R_k(\lambda)[0.5P_l(\lambda) + 0.75P_m(\lambda)]d\lambda \\ = 0.5\int_\omega R_k(\lambda)P_l(\lambda)d\lambda + 0.75\int_\omega R_k(\lambda)P_m(\lambda)d\lambda\end{array} \tag{3}$$

An implication of (3) is that the camera response to an arbitrary intensity weighting of the display outputs can be written as a matrix equation:

$$\begin{bmatrix} \rho_r \\ \rho_g \\ \rho_b \end{bmatrix} = \begin{bmatrix} \rho_r^l & \rho_r^m & \rho_r^s \\ \rho_g^l & \rho_g^m & \rho_g^s \\ \rho_b^l & \rho_b^m & \rho_b^s \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \Rightarrow \underline{\rho} = M\underline{\alpha} \tag{4}$$

here α, β and γ vary the intensity of the colour channels (from 0 to 100% or minimum to maximum power). We are now in a position to solve for the display weights i.e. solve for the correct RGBs to drive the display. Denoting the 3-vector of responses in (1) as $\underline{\rho}$ then the correct image display weights $\underline{\alpha}$ (the values recorded in an image pixel) is calculated as:

$$\underline{\alpha} = M^{-1}\underline{\rho} \tag{5}$$

Equation (5) is called colour correction[26]. Note the 3x3 matrix M^{-1} is fixed for a given camera and display. Equation (5) is also the exact solution for colour matching (i.e. how we mix three primary lights to match an arbitrary test light).

However, in reality, it is never the case that a camera samples light like colour matching functions. Thus, the mapping which takes acquired RGBs to display outputs is approximate (and is solved for through regression[26]). We will return to this problem again in section 4 - see equation (15).

For historical reasons, displays typically have a non-linear transfer function. That is, the brightness output is (roughly) the square of the rgb driving the display. Thus the values stored in an image are the square-root of the weights calculated in (5). This process is called gamma correction.

Colour correction is a first order effect. The raw images recorded by a camera are not suitable for display, The effect of colour correction is illustrated in Figure 3.

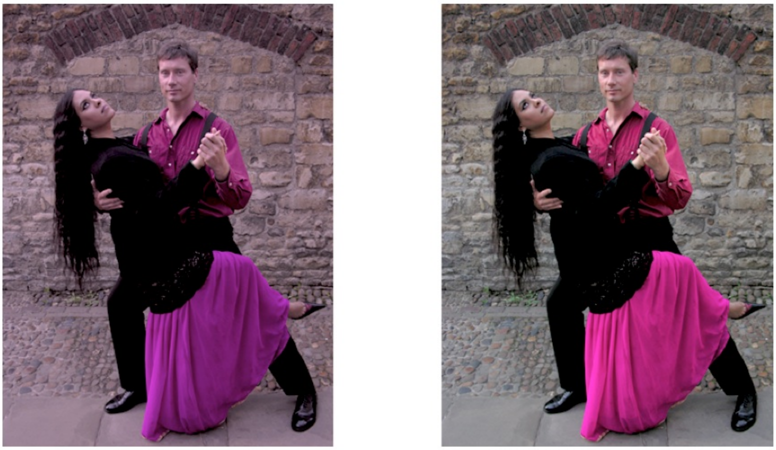


Fig. 3 Left: image before colour correction. Right shows the corrected colours

2 Colour Constancy and Illuminant Estimation

The colours we see in the world do not depend on the colour of the illuminant. A white T-shirt looks white whether it is viewed in direct sunlight (yellowish colour of light) or in deep shadow (bluish light). Indeed, from an evolutionary point of view such colour constancy is clearly very desirable. As an example, in primate vision it has been proposed that colour is an important cue for judging the ripeness of fruit[24].

More generally, we do not expect the colour of the world to change as we move from one environment to another. Indeed, colour is often the primary designator we use in describing objects e.g. the red car or the green door. Yet, physically, the colour signal reflected from a surface may not, is typically not, the same as the object reflectance.

The idea that the colour we see was not a property of the spectrum of light entering the eye (the Newtonian view) is a relatively modern notion. Indeed, Edwin Land (the progenitor of Polaroid corporation) sparked a huge debate in the colour community when, in the 1950s, 60s and 70s, he proposed his Retinex Theory[19] of colour vision (to account for the phenomenon of colour constancy).

Simply, and perhaps somewhat obviously in hindsight, the Retinex theory proposes that the colours we see depend on the context in which we see them. Figure 4 (an example from Beau Lottos lab) illustrates this point. The same physical sample, viewed in two different illumination contexts, looks like it has a different colour.

In Figure 5, we show a colour constancy example from the Computer Vision literature[1]. Here the same object is captured under 4 different lights. It is remarkable how much the colour varies. It is evident then that raw colour does not correlate with object colour. Only if an object's colour *is* independent of illumination can it be used for recognition, indexing or tracking.

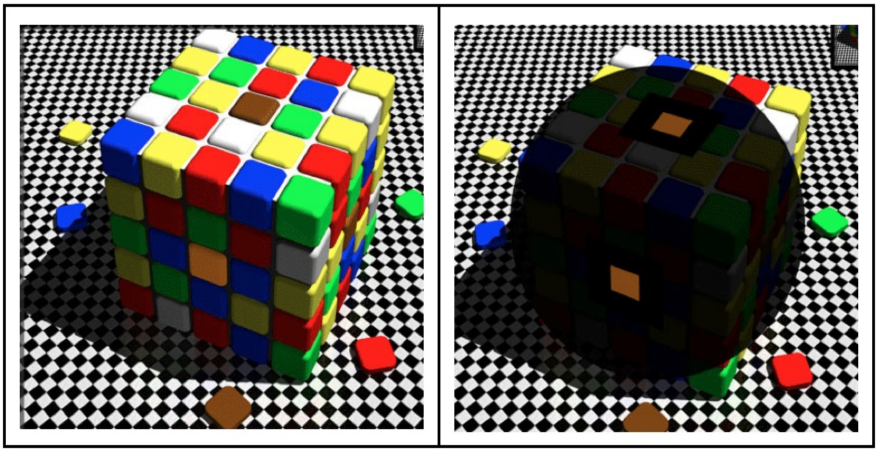
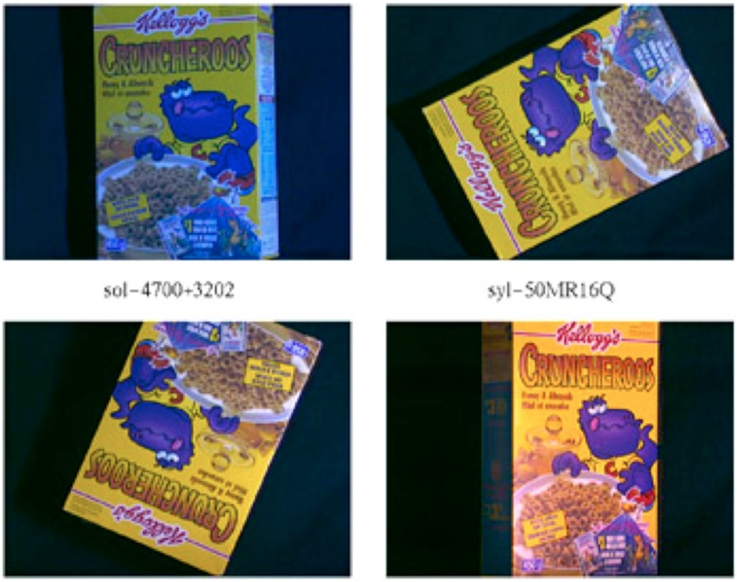


Fig. 4 The brown and orange surface chips on the top and front sides emit physically the same light. We see them as different colours as we perceive them both as a function of other colours in the scene and our physical interpretation of the scene. Clearly, we interpret the front cube face as being in shadow. The right hand panel of the figure (by using a black mask to remove the local context) demonstrates the chips reflect the same identical physical signal (from http://www.lottolab.org/).



sol-4700+3202

syl-50MR16Q

Fig. 5 The same object viewed under 4 common lights. It is remarkable how much the colours of the object depends on the colour of the prevailing light[1]

However, getting the colour right is in itself of great interest as a problem in digital photography. We are very attuned (and highly critical) judges of the colours that look right or which look wrong when we look at photographs. Figure 6 shows an example of colour constancy from digital photography. To recover the image on the right we must estimate the illuminant colour and then remove its bias from the image. In photography, this process is often called 'White point adjustment'.

2.1 Estimating the Illuminant

Simple as Equation (1) is, it is in fact quite complex. Even assuming we know the spectral sensitivities of our camera, it is not immediately apparent that we can decouple and recover light $E(\lambda)$ from reflectance $S(\lambda)$. Indeed, each RGB supplies only 3 measurements which is not a propitious starting point for determining how we can solve the colour constancy problem.

To understand how we can, *practically*, solve the colour constancy problem, let us begin by making simplifying assumptions. First, let us assume that rather than recovering the spectrum of the light (or the spectrum of the surface reflectance) we instead wish only to recover the RGB of the light and the RGB of the surface. Second, let us assume that the camera measured RGB is the multiplication of the

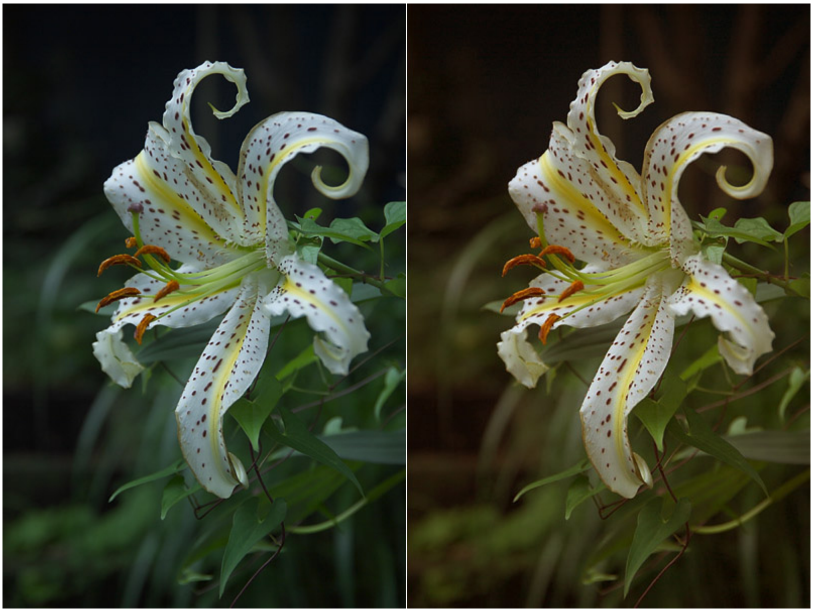


Fig. 6 Left shows raw camera image, right after colour constancy (called white balance adjustment in photography). From http://en.wikipedia.org/wiki/Color_balance

RGB of the light and the RGB of the surface (this assumption is commonly made in computer graphics[3]). The **RGB model of colour image formation** is written as:

$$\rho_k^S = \int_\omega S(\lambda)R_k(\lambda)d\lambda \quad \rho_k^E = \int_\omega E(\lambda)R_k(\lambda)d\lambda$$

$$\rho_k^{E,S} = \rho_k^E \rho_k^S \qquad \text{(RGB model of image formation)}$$

Remarkably, these assumptions, with certain caveats, generally hold[6]. An important interpretation of ρ_k^S is that it is the colour of the surface viewed under a white uniform light $E(\lambda) = 1$. Subject to this observation, colour constancy can be thought of as mapping the rgbs measured in an image back to a reference lighting condition. That is, the colour constancy problem involves solving for ρ_k^S . Clearly, if we can estimate the illuminant (solve for the rgb of the light) then by dividing out we can estimate the surface colour.

2.2 *The Maloney Wandell Algorithm*

In 1986 Maloney and Wandell[22] presented perhaps the first formal treatments of the colour constancy problem. Their idea was that if light and surface were modelled by 3- and 2-dimensional linear models it would be possible to solve for colour constancy at a colour edge (i.e. given the rgb response of just two coloured surfaces).

Linear models of light and surface are written as

$$E(\lambda) = \Sigma_{i=1}^3 \varepsilon_i E_i(\lambda) \quad S(\lambda) = \Sigma_{j=1}^3 \sigma_i S_i(\lambda) \tag{6}$$

The intuition bbehind Maloney and Wandells approach is simple equation counting. Given two RGBs we have 6 measurements. Assuming the same light and two reflectances in a scene there are 2*2+3=7 unknowns. However, given the image formation equation (1) it is clear that we cannot distinguish between a bright light illuminating a dim scene or the converse. Thus, there are 6 equations and 6 unknowns to solve for. So, under the linear model assumptions (6), it is plausible we can estimate the RGB of the light given a pair of rgbs (for two different surfaces viewed under the same illuminant). Further, and crucially, 2- and 3-dimensional models for surface and light capture most of the variation found in typical reflectances and illuminations[22, 21].

So, how does plausibility translate into an actual algorithm? Well, here, we do not present their exact solution method (which is very general) but rather the equivalent algorithm that is simpler to implement[5] (which follows from the RGB model of image formation). We begin by observing that if reflectance has two degrees of freedom then this means that the RGB response of any surface under a single light must lie on a 2-dimensional plane. This idea is illustrated by the plane on the left of Figure 7.

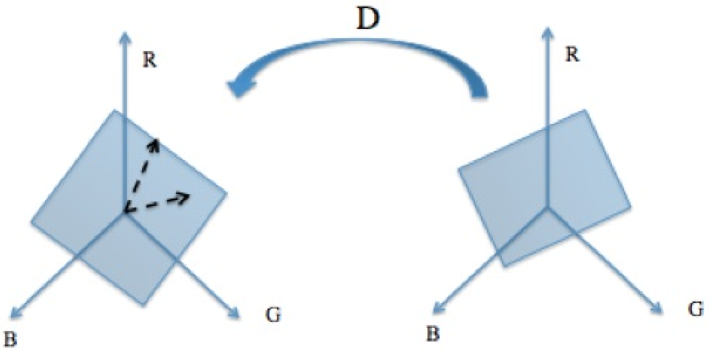


Fig. 7 Left shows the plane of RGBs measured by camera (spanned by the two actual measurements shown as dotted lines) under an unknown light. Right, the set of all camera measurements - also a plane - for a white reference light. The mapping taking right to left defines the colour of the unknown light.

The plane on the right shows the set of all possible camera responses for known white light reference conditions. Let us now rewrite the RGB model of image formation as the matrix equation:

$$\begin{bmatrix} \rho_R^{E,S} \\ \rho_G^{E,S} \\ \rho_B^{E,S} \end{bmatrix} = \begin{bmatrix} \rho_R^E & 0 & 0 \\ 0 & \rho_G^E & 0 \\ 0 & 0 & \rho_B^E \end{bmatrix} \begin{bmatrix} \rho_R^S \\ \rho_G^S \\ \rho_B^S \end{bmatrix} \Rightarrow \underline{\rho} = M\underline{\alpha} \qquad (7)$$

Because there is a unique diagonal matrix mapping one plane to any other plane[5] then if we can find the diagonal matrix mapping the plane in the right of Figure 7 (the plane where rgbs lie under a white light) to the plane we observe for our RGB camera (the one on the left) then we have solved for the colour of the light. The diagonal matrix D and the colour of the light are one and the same thing).

Finally, by dividing out, we can solve for the colour of the surfaces.

$$\frac{\rho_k^{E,S}}{\rho_k^E} = \rho_k^S \qquad (8)$$

Unfortunately, as elegant as this algorithm is, it actually delivers terrible colour constancy performance (the linear model assumptions do not hold sufficiently well). However, the idea that the colours we observe in an image provide prima facie evidence about the colour of the light is a good one: the reddest red RGB cannot be measured under the bluest light[12]. This idea is a the heart of many modern illuminant estimation algorithms.

Moreover, and more importantly, the tool of linear models has proven to be invaluable both to understanding complex problems and arriving at tractable algorithmic solutions.

2.3 Statistical Illuminant Estimation

Curiously, most illuminant estimation algorithms are based on a much simpler heuristic idea. Specifically, that the colour bias due to the illumination will manifest itself in summary statistics calculated over an image. If the colour of the prevailing light is yellowish then the mean of the image will be more yellow than it ought to be. So, it is reasoned, mapping the mean of the image so it is neutral (the mean of the red, green and blue channels are all equal) should deliver colour constancy. This approach is called grey-world colour constancy. It is easy to show that dividing by the mean is mathematically correct if the expected colour of every scene is gray[13].

In Lands Retinex theory[19] it was (effectively) argued that the maximum red, maximum green and maximum blue channels response is a good estimate of the colour of the light. Should every scene contain a white reflectance then this simple maxRGB approach will work. It would, for example work for the example shown in Figure 6. However, it is easy to find examples of images where neither max RGB nor grey world work very well.

It was observed[11] that the grey-world and maxRGB algorithms are simply the $p = 1$ and $p = \infty$ Minkoswki norms. Minkowski illuminant estimation is, assuming N pixels in an image, written as:

$$\rho_k^E = [\Sigma_{i=1}N[\rho_{k,i}]^p/N]^{(1/p)} \tag{9}$$

Remarkably, across a number of image datasets[10] a p-norm of 4 or 5 returns more accurate estimates of the illuminant than either max RGB or grey-world.

2.4 Evaluating Illuminant Estimation Algorithms

What do we mean if we say that one illuminant estimation algorithm works better than another? (i.e. that a p=4 Minkowski norm approach works best). In answering this question it is common to assume that the measured physical white point (the rgb of a white tile placed in the scene) is the correct answer. The angle between the estimated rgb of the illuminant and the actual true white point (the rgb for the white tile) is taken to be a measure of how accurate an estimate of the illuminant actually is.

The reference[14] provides a broad survey of a large number of illuminant estimation algorithms evaluated on a large number of data sets. The reported experiments convey two important messages. First illuminant estimation is a hard problem and even the best algorithms can fail (sometimes spectacularly). Second, progress on improving illuminant estimation is slow: its taken 30 years to provide a modest increase in performance.

Camera manufacturers remain interested in improving their white balance algorithms. Not only do they seek methods which work better, they are interested in identifying images that have multiple lights (sun and shadow) [16]. Modern algorithms have implemented Face detectors to aid estimation[23, 2].

3 Illuminant Invariants

The colours in two pictures of the same object observed from different viewpoints can be quite appear to be quite differen from each other and so, it is not always easy to find corresponding parts from two images that are the same (a necessary step to solve the stereopsis or shape from motion problems). Thus, in carrying out geometric matching it is common to seek geometric invariants i.e. features which do not change with a change in viewpoint. In David Lowes famous SIFT detector[20] features are sought that are invariant to scale and rotation. However, photometric invariance is also a useful property.

In the colour world, Swain and Ballard found that the distribution of colours in an image provides a useful cue for object recognition and object localisation[25]. Unfortunately, a precondition for that method to work is that image colour correlates with object colour. Yet, as we have seen in section 2, the same object will have a different image colour when viewed under differently coloured lights. Indeed, the same physical colour might have a range of intensities (i.e. shading) if the object has shape or the illumination intensity varies across a scene. Equally, if the colour of the light changes then the physical recorded colour will change as well. In either case, matching colour (e.g. to a database of images) without considering this problem can result in very poor recognition performance

Colour change due to changing lighting intensity (due to Lamberts law) and lighting colour is illustrated in Figure 8. The simple test image shown at the top of the figure is imaged under 3 coloured lights and from 3 different light positions. Directly below the image capture diagram we show the corresponding 3x3 image patches for the 9 imaging conditions. It is apparent that there is a remarkable variety of different coloured images resulting from the same physical scene.

If we think of this simple colour edge as a region of interest in the image, then Figure 8 informs us the edge RGBs change when the viewing condition changes. Photometric normalisatiion methods seek simple algebraic formulae or algorithms for canceling out this image variation.

3.1 Intensity Invariance

We can normalise for the lighting geometry of the scene - the intensity variation of the object RGBs due to shading and the position of the light source - simply, by dividing each RGB by its magnitude:

$$\begin{bmatrix} r \\ g \\ b \end{bmatrix} = \begin{bmatrix} R/(R+G+B) \\ G/R+G+B) \\ B/R+G+B \end{bmatrix} \tag{10}$$

Clearly [R G B] and [kR kG kB] have the same normalised output. Note also post-normalisation that b=1-r-g. That is, by removing intensity the colour at each pixel is parameterised by just two numbers. The tuple (r,g) is sometimes called the chromaticity of the RGB.

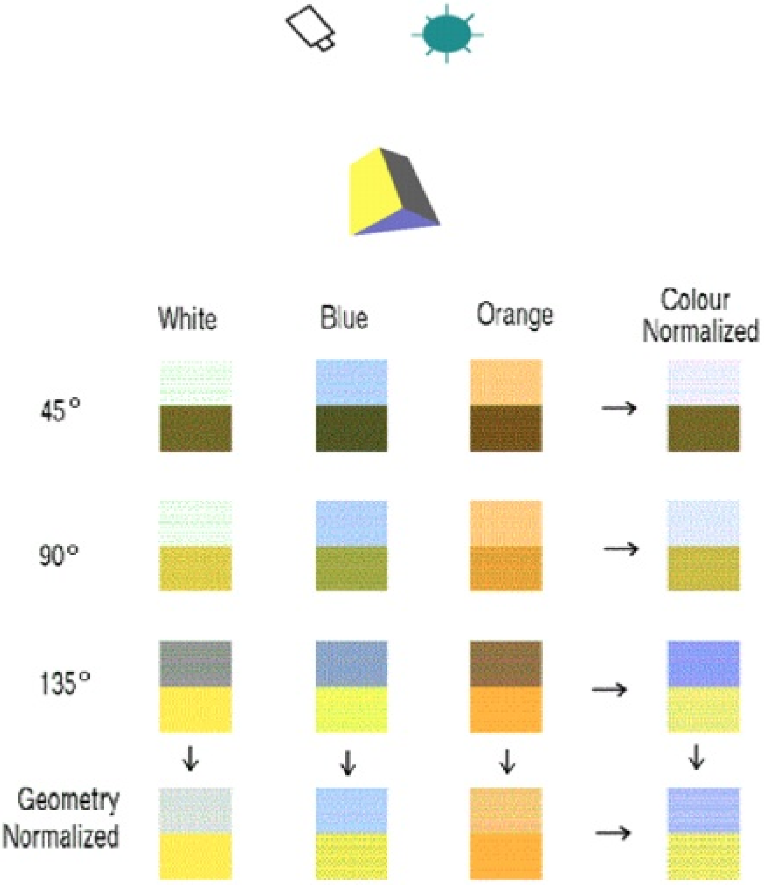


Fig. 8 Top a simple wedge is viewed under a light source. The light source can be one of 3 different colours and be place at 3 different positions. The upper 3x3 image outputs show the range of recorded colours for the wedge for the 3 lighting positions and 3 light colours. The last row shows the output of intensity normalisation and the last column the result of colour normalisation. The patch bottom right is the output of colour and intensity nomalisations carried out iteratively.

The effect of intensity normalisation is shown in the bottom row of Figure 8. It is clear when ony the intensity varies that intensity normalisation suffices to make the colour images the same.

3.2 Colour Invariance

We achieve invariance to the colour of the light in a similar way though now we work not with the RGB at each pixel but rather with all the pixel values in a single colour channel. The RGB of the ith pixel is made invariant to the colour of the light by calculating:

$$\begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix} = \begin{bmatrix} R_i/\mu(R) \\ G_i/\mu(G) \\ B_i/\mu(B) \end{bmatrix} \tag{11}$$

In (11) we divide each pixel value by the average of all the pixels (in the same colour channel). Because we are adopting the RGB model of image formation (from section 2) the illuminant colour must appear in both the numerator and denominator of the right hand side of (11) and, so, must cancel.

The effect of this illuminant normalisation is shown in the right hand column of Figure 8. If only the illuminant colour changes then (7) suffices to normalise the colours (all the images in the same row have the same output colours).

However, by dividing by the mean is similar to the grey-world colour constancy algorithm discussed in section 2 (we divide by the p=1 norm of Eq. (9)). The only difference is that in colour constancy research we wish the normalised colours to look correct. The bar is set lower for colour invariance: it suffices that same object viewed under different lights is normalised to the same (albeit often false) image colours.

3.3 Comprehensive Normalisation

Remarkably, in[8] it was shown that if we iteratively calculate (10) (intensity invariance) and then (11) (colour invariance) then this process converges to an output that is independent of lighting geometry and light colour. The 9 input images in Figure 8 all converge to the same single output shown bottom right. Importantly, colour normalisation (intensity, colour and comprehensive) has been shown to be useful for object recognition and image indexing[8].

3.4 Colour Constancy at a Pixel

Let us suppose that we could calculate intensity and colour invariance at a pixel i.e. at an image containing a single RGB pixel. We cannot do this using comprehensive normalisation. Indeed, any input pixel will, by iteratively applying (10) and (11), result in the triple (1,1,1) i.e. we get invariance in a trivial sense (all input RGBs map to the same output colour).

In fact under typical illuminant conditions it is, remarkably, possible to find a single scalar value that is independent of the intensity and is independent of the colour of the light. To see this, we begin by adopting an alternative chromaticity definition:

$$\begin{bmatrix} r \\ b \end{bmatrix} = \begin{bmatrix} R/G \\ B/G \end{bmatrix} \tag{12}$$

Note for all RGBs G/G=1 and so we ignore this term ($[r \;\; b]^t$] encodes RGB up to an unknown intensity) The formula in (12) is useful because it implies that[4]:

$$\begin{bmatrix} r^{E,S} \\ b^{E,S} \end{bmatrix} = \begin{bmatrix} r^E & 0 \\ 0 & b^E \end{bmatrix} \begin{bmatrix} r^S \\ b^S \end{bmatrix} \tag{13}$$

i.e. the chromaticity response is a simple multiplication of the chromaticity of the light and the chromaticity of the reflectance. The diagonal model of (3) for RGBs holds for spectral band ratios too.

In[7] the following experiment is carried out. A picture of a Macbeth colour checker, shown in the top of Figure 9, is captured. There we mark 7 basic colours: Red, Orange, Yellow, Green, Blue, Purple and White patches. We now take pictures of these patches under 10 different typical lights ranging from indoor yellow tungsten to white cloudy day light to blue sky i.e. a range of typical lights For each of the 7 patches we plot the spectral band ratios on a log-log scale. We plot these results in the graph shown at the bottom of Figure 9.

Note, that as the illumination changes the spectral band ratios sweep out a line on the log-log plot. More importantly, the slope of each line is the same for all surface colours but the intercept varies. Clearly then, the intercept can be used as an scalar measure of reflectance which, empirically at least, does not vary with illumination.

Suppose we take a picture of the world where there are cast shadows. The light colour in and out of the shadow are different (the light for the shadow region is much bluer). Assuming that the data shown in Figure 9 holds in general (implies that there is an intrinsic reflectance invariant calculable at a pixel) then we should be able to simply - trivially - remove shadows.

First, we remember we know the slope of the lines in Figure 9. Second, for a given RGB we calculate its log spectral band ratio coordinates. Then we can calculate the intercept (with either the x- or y- axis). We then take the scalar image of intercepts and recode as a greyscale image. In Figure 10 we show the outputs of applying this methodology. The shadows magically disappear.

In general, the grey-scale invariant image, that is independent of the colour of the light and intensity, conveys salient information and has been shown to be useful in applications ranging from scene understanding[9], to object recognition[7] to tracking[18].

The reader will, no doubt, be curious as to why spectral band ratios on a log log plot look as they do in Figure 9. Well, it turns out that most lights (at least in

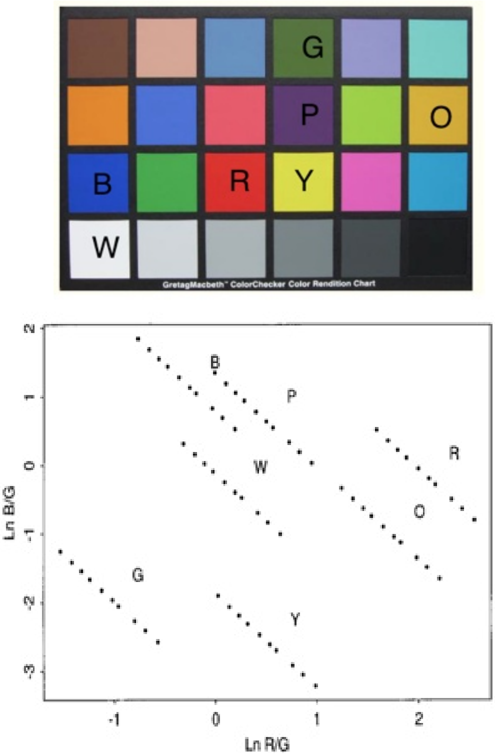


Fig. 9 Top, Macbeth colour checker. Bottom, log spectral band ratios for marked surfaces for 10 lights (the dots shown).

terms as how they project to form RGBs) can be thought of as Planckian black-body radiators. The colour of black-body radiators is parameterised by one number: temperature. Because of the form of the mathematical equation that models black-body radiation, it turns out that the log-log plot must look like Figure 8. For a description of why this is the case, the reader is referred to [7].

4 Computer Vision and Colour Perception

There are many applications where we would like a machine vision to see like we do. Unfortunately, we do not have good operational models of our own vision system so we, instead, seek to equip machine vision systems with simpler -though, still useful - competences. In the applied colour industries there are specialised measurement devices that attempt to numerically gauge the similarity of colour pairs. Figure 10 illustrates the industrial colour difference problem. Perceptual relevance in machine vision is sometimes taken to mean that the vision system might be used for colour difference assessment.

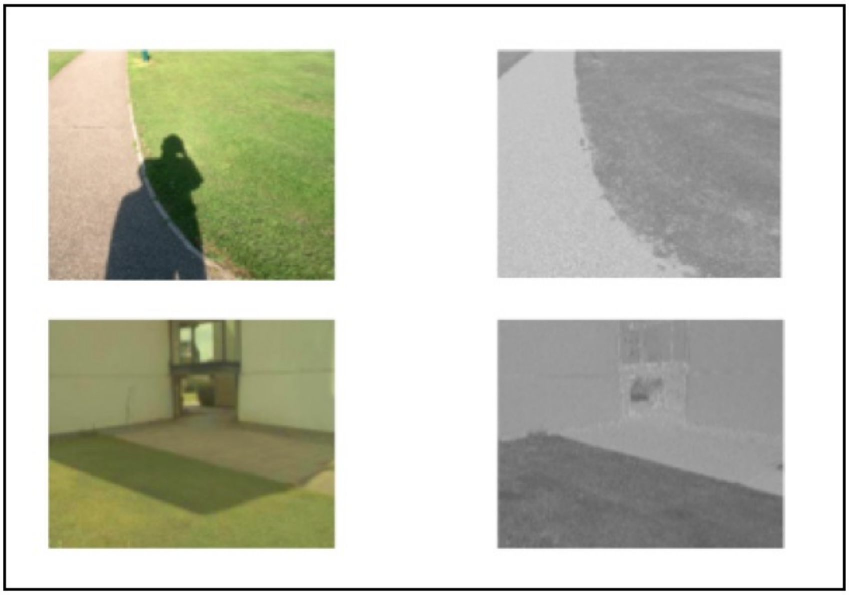


Fig. 10 Left, Raw camera image. (As described in the text), illuminant invariant calculated in the right: the shadows magically disappear.).

Assuming we had camera sensors with sensitivities the same as those shown in 2c (not generally the case) and if we also knew the colour of the light then there are standardised formula[27] for mapping camera measurements to, so called, Lab values. Euclidean distance in Lab space approximately account for the perceived difference between stimuli. Specifically, a distance of 1 correlates with a just noticeable difference. We recapitulate the CIE Lab[27] equations below:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x/x_n \\ y/y_n \\ z/z_n \end{bmatrix}$$

$$\begin{bmatrix} L \\ a \\ b \end{bmatrix} = \begin{bmatrix} 0 & 1160 & \\ 500 & -500 & 0 \\ 0 & 200 & -200 \end{bmatrix} \begin{bmatrix} x^{1/3} \\ y^{1/3} \\ z^{1/3} \end{bmatrix} + \begin{bmatrix} -16 \\ 0 \\ 0 \end{bmatrix} \tag{14}$$

Here x, y and z are the responses of a camera with sensitivities 2c. The triple (x_n, y_n, z_n) is the camera response to the illuminant. The Lab formula was derived [27] by fitting psychophysical data (real colour difference judgements made by people). In Figure 11, the 'Delta E' colour difference is about 9 indicating a visually significant colour difference.

To use a vision system for colour grading when the sensitivities are not like those in 2c (the actual sensitivities of a commercial camera are shown in 2d) then this

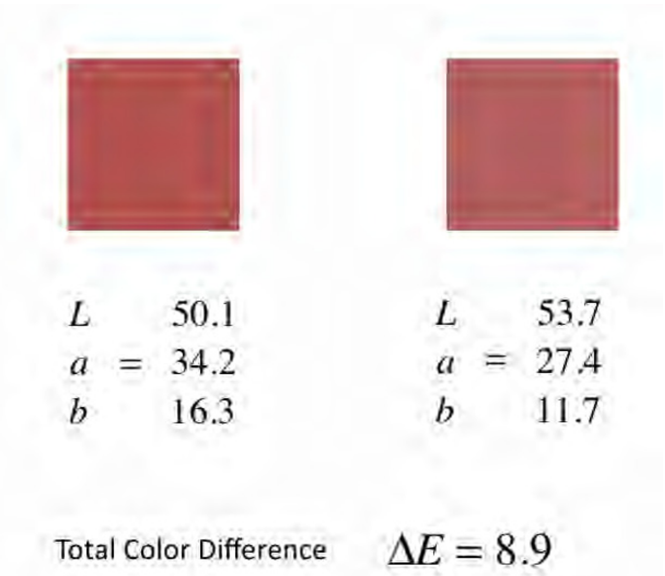


Fig. 11 Two similar colour patches are measured and their colours summarised according to three (L, a and b) coordinates. The Euclidean distance between the triples correlates with perceived colour difference.

means that the actual camera colours must be mapped to approximate corresponding xyzs. This mapping is often solved for as a simple linear transform:

$$\min_{T} ||R_{N\times 3}T_{3\times 3} - X_{N\times 3}|| \tag{15}$$

where, $R_{N\times 3}$ above is a set of measured RGBs for a calibration target (e.g. of the kind shown on the left of Figure 9). $X_{N\times 3}$ are the corresponding measured XYZs. Once we have solved for the best regression matrix T we can use a camera to measure arbitrary scenes and calculate Labs according to the above formulae.

4.1 Colour Difference Formulae and Computer Vision (a Cautionary Remark)

That we might carry out a simple calibration and recover approximate Lab values is all well and good if we wish to carry out colour measurement. But, the reader should be aware that Euclidean distance on CIE Lab values only models small colour differences. If a pair of colours are compared and found to be (say) 20 units apart, this means almost nothing at al. i.e. we cannot measure the perceived closeness of red and green using Lab colour differences.

Unfortunately, in computer vision researchers sometimes assume that once we transform to Lab then we have somehow carried out a ’perception transform’. It is naively proposed that, simply, by transforming to Lab space we can claim perceptual relevancy. One cannot. It is often quite inappropriate to claim that a given tracking, recognition, object finding algorithm in Lab space says anything much about our own perception or how we ourselves solve these problems.

5 Conclusion

Colour is a huge field and is studied in physics, computer science, psychology and neuroscience (among other fields). While great progress has been made in the last 100 years, colour is still far from a solved problem. That this is so, accounts, in part, for colour sometimes being used wrongly in computer vision.

In this short primer we have tried to introduce the reader to colour in computer vision. We have explained how camera RGBs are mapped to image colours that drive the display (colour correction). Removing colour bias due to illumination (colour constancy) is perhaps the most studied aspect of colour in computer vision. Solving for colour constancy is essential if colour is to be used as an absolute correlate to reflectance. However, relative measures of colour - functions of proximate pixels - can be used to cancel illumination effects (colour invariance). Remarkably, we can calculate a grey-scale invariant at a pixel which cancels the colour and intensity of the light (with respect to which shadows, magically, disappear).

The assumption that a camera system might easily play a surrogate role for our own vision system is a seductive idea. The good news is that, yes, colour cameras can be used for colour measurement. The bad news is that colour measurement does not really say anything very profound about how we see

Acknowledgements. The author gratefully acknowledges the support of EPSRC grant H022236,

References

1. Barnard, K., Martin, L., Funt, B., Coath, A.: A data set for color research. Color Research and Application 27(3), 147–151 (2002), `http://dx.doi.org/10.1002/col.10049`,doi:10.1002/col.10049
2. Bianco, S., Schettini, R.: Color constancy using faces. In: CVPR, pp. 65–72 (2012)
3. Borges, C.: Trichromatic approximation for computer graphic illumination models. Computer Graphics 25, 101–104 (1991)
4. Finlayson, G.: Color in perspective. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1034–1038 (1996)
5. Finlayson, G., Drew, M., Funt, B.: Color constancy: Generalized diagonal transforms suffice. J. Opt. Soc. Am. A 11, 3011–3020 (1994)
6. Finlayson, G., Drew, M., Funt, B.: Spectral sharpening: Sensor transformations for improved color constancy. J. Opt. Soc. Am. A 11(5), 1553–1563 (1994)
7. Finlayson, G., Hordley, S.: Color constancy at a pixel. JOSA-A 18(2), 253–264 (2001)
8. Finlayson, G.D., Schiele, B., Crowley, J.L.: Comprehensive colour image normalization. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 475–490. Springer, Heidelberg (1998)

9. Finlayson, G.D., Hordley, S.D., Lu, C., Drew, M.S.: On the removal of shadows from images. IEEE Trans. Pattern Anal. Mach. Intell. 28(1), 59–68 (2006)
10. Finlayson, G.D., Rey, P.A.T., Trezzi, E.: General p constrained approach for colour constancy. In: ICCV Workshops, pp. 790–797 (2011)
11. Finlayson, G.D., Trezzi, E.: Shades of gray and colour constancy. In: Color Imaging Conference, pp. 37–41 (2004)
12. Forsyth, D.: A novel algorithm for color constancy. Int. J. Comput. Vision 5, 5–36 (1990)
13. Gershon, R., Jepson, A., Tsotsos, J.: Ambient illumination and the determination of material changes. J. Opt. Soc. Am. A 3, 1700–1707 (1986)
14. Gijsenij, A., Gevers, T., van de Weijer, J.: Computational color constancy: Survey and experiments. IEEE Transactions on Image Processing 20(9), 2475–2489 (2011)
15. Horn, B.: Robot Vision. MIT Electrical Engineering and Computer Science Series. MIT Press (1986)
16. Hubel, P.M.: The perception of color at dawn and dusk. In: Color Imaging Conference, pp. 48–51 (1999)
17. Hunt, R.: Measuring Colour, 3rd edn. Fountain Press (2001)
18. Jiang, H., Drew, M.S.: Shadow resistant tracking using inertia constraints. Pattern Recognition 40(7), 1929–1945 (2007)
19. Land, E.: The retinex theory of color vision. Scientific American, 108–129 (1977)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
21. Maloney, L.: Evaluation of linear models of surface spectral reflectance with small numbers of parameters. J. Opt. Soc. Am. A 3, 1673–1683 (1986)
22. Maloney, L., Wandell, B.: Color constancy: a method for recovering surface spectral reflectance. J. Opt. Soc. Am. A 3, 29–33 (1986)
23. Montojo, J.: Face-based chromatic adaptationfor tagged photo collections (2009)
24. Regan, B., Julliot, C., Simmen, B., Vinot, F., Charles-Dominique, P., Mollon, J.: Frugivory and colour vision in alouatta seniculus, a trichromatic platyrrhine monkey. Vision Research 38(21), 3321–3327 (1998), `http://www.sciencedirect.com/science/article/pii/S0042698997004628`, doi:10.1016/S0042-6989(97)00462-8
25. Swain, M., Ballard, D.: Color indexing. International Journal of Computer Vision 7(11), 11–32 (1991)
26. Vrhel, M.J., Trussell, H.J.: The mathematics of color calibration. In: ICIP (1), pp. 181–185 (1998)
27. Wyszecki, G., Stiles, W.: Color Science: Concepts and Methods, Quantitative Data and Formulas, 2nd edn. Wiley, New York (1982)

Descriptor Learning for Omnidirectional Image Matching

Jonathan Masci, Davide Migliore, Michael M. Bronstein, and Jürgen Schmidhuber

Abstract. Feature matching in omnidirectional vision systems is a challenging problem, mainly because complicated optical systems make the theoretical modelling of invariance and construction of invariant feature descriptors hard or even impossible. In this paper, we propose learning invariant descriptors using a training set of similar and dissimilar descriptor pairs. We use the similarity-preserving hashing framework, in which we are trying to map the descriptor data to the Hamming space preserving the descriptor similarity on the training set. A neural network is used to solve the underlying optimization problem. Our approach outperforms not only straightforward descriptor matching, but also state-of-the-art similarity-preserving hashing methods.

1 Introduction

Feature-based matching between images has become a standard approach in computer vision literature in the last decade, in many respects due to the introduction of stable and invariant feature detection and description algorithms such as SIFT [22] and similar methods [26, 2, 37]. The usual assumption guiding the design of feature descriptors is invariance across viewpoints, which should guarantee that the same feature appearing in two different views has the same descriptor. Since perspective

Jonathan Masci · Jürgen Schmidhuber
IDSIA, USI and SUPSI – Galleria 2, 6928 Manno-Lugano, Switzerland
e-mail: {jonathan,juergen}@idsia.ch

Davide Migliore
Evidence S.r.l.
e-mail: d.migliore@evidence.eu.com

Michael M. Bronstein
Dept. of Informatics Università della Svizzera Italiana Lugano, Switzerland
e-mail: michael.bronstein@usi.ch

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_3, © Springer-Verlag Berlin Heidelberg 2014

transformations are approximately locally affine, it is common to construct affine-invariant descriptors [20].

While being a good model in many cases, affine invariance is not sufficiently accurate in cases of wide baseline (very different view points) or even more complicated setting of optical imperfections such as lens distortions, blur, etc. In particular, in omnidirectional vision systems the distortion is introduced intentionally (e.g., using a parabolic mirror [24]) to allow a 360° view. Designing invariant descriptors for such cases is challenging, as the invariance is complicated and cannot be easily modeled.

An alternative to 'invariance-by-construction' approaches which rely on a simplified invariance model is to *learn* the descriptor invariance from examples. Recent work of Strecha *et al.* [34] showed very convincingly that such approaches can significantly improve the performance of existing descriptors.

In this paper, we consider the learning of invariant descriptors for omnidirectional image matching. We construct a training set of similar and dissimilar descriptor pairs including strong optical distortions, and use a neural network to learn a mapping from the descriptor space to the Hamming space preserving similarity on the training set. Experimental results show that our approach outperforms not only straightforward descriptors, but also other similarity-preserving hashing methods. The latter observation is explained by the suboptimality of existing approaches which solve a simplified optimization problem.

The main contribution of this paper is two-fold. First, we formulate a new similarity-sensitive hashing algorithm. Second, we use this approach to learn smaller invariant descriptors suitable for feature matching in omnidirectional images. The rest of the paper is organized as follows. In Section 2, we overview the related works. Section 3 is dedicated to metric learning and similarity-preserving hashing methods. In Section 4, we describe our NNhash approach. Section 5 contains experimental results. Finally, Section 6 discusses potential future work and concludes the paper.

2 Background

Although feature-based correspondence problems have been investigated in depth for standard perspective cameras, omnidirectional image matching still remains an open problem, largely because of the complicated geometry introduced by lenses and curved mirrors. Broadly speaking, the existing approaches either try to reduce the problem to the simpler perspective setting, or design special descriptors suitable for omnidirectional images.

Svoboda *et al.* [35] proposed to use adaptive windows around interest points to generate normalized patches with the assumption that the displacement of the omnidirectional system is smaller than the depth of the surrounding scene. Nayar [27] showed that, given the mirror parameters, it is possible to generate a perspective version of the omnidirectional image and Mauthner *et al.* [23] used this approach to generate perspective representation of each interest point region. This unwarping procedure removes the non-linear distortions and enables the use of algorithms designed for perspective cameras. Micusik and Pajdla [25] checked the candidate correspondences between two views using the RANSAC algorithm and the epipolar constraint [12]. Construction of scale-space by means of diffusion on manifolds was used in [3, 15, 10] for the construction of local descriptors. Puig *et al.* [28] integrated the sphere camera model with the partial differential equations on manifolds framework.

Another possible solution is to consider different kind of features to exploit particular invariance in omnidirectional systems, for example, extracting one-dimensional features [5] or vertical lines [31] and defining descriptors suitable for omnidirectional images.

More recently, it was shown in [34] that one can approach the design of invariant descriptors from the perspective of *metric learning*, constructing a distance between the descriptor vectors from a training set of similar and dissimilar pairs [1, 41]. In particular, *similarity-preserving hashing* methods [13, 33, 42, 21, 29] were found especially attractive for descriptor learning, as they significantly reduce descriptor storage and comparison complexity. These methods have also been applied to image search [16, 38, 18, 17, 19, 40], video copy detection [7], and shape retrieval [6].

In [30], binary codes were produced using a restricted Boltzmann machine and in [42] using spectral hashing in an unsupervised setting. The authors showed that the learnt binary vectors capture the similarities of the data. With such an approach it is however impossible to explicitly provide information about data similarities. Since in our problem it is easy to produce labeled data, supervised metric learning is advantageous.

3 Similarity Preserving Hashing

Given a set of keypoint descriptors, represented as n-dimensional vectors in $\mathbb{R}^n$, the problem of *metric learning* is to find their representation in some metric space $(\mathbb{Z}, d_{\mathbb{Z}})$ by means of a map of the form $\mathrm{y} : \mathbb{R}^n \to (\mathbb{Z}, d_{\mathbb{Z}})$. The metric $d_{\mathbb{Z}} \circ (\mathrm{y} \times \mathrm{y})$ parametrizes the similarity between the feature descriptors, which may be difficult to compute in the original representation. Typically, $(\mathbb{Z}, d_{\mathbb{Z}})$ is fixed and y is the map we are trying to find in such a way that, given a set $\mathscr{P}$ of pairs of descriptors from corresponding points in different images (*positives*) and a set $\mathscr{N}$ of pairs of descriptors from different points (*negatives*), we have $d_{\mathbb{Z}}(\mathrm{y}(\mathrm{x}), \mathrm{y}(\mathrm{x}^+)) \approx 0$ for all $(\mathrm{x}, \mathrm{x}^+) \in \mathscr{P}$ and $d_{\mathbb{Z}}(\mathrm{y}(\mathrm{x}), \mathrm{y}(\mathrm{x}^-)) \gg 0$ for all $(\mathrm{x}, \mathrm{x}^-) \in \mathscr{N}$ with high probability.

A particular setting of this problem, where $\mathbb{Z} = \{\pm 1\}^m$ is the m-dimensional space of binary strings and $d_{\mathbb{H}^m}(\mathrm{y}, \mathrm{y}') = \frac{m}{2} - \frac{1}{2}\sum_{i=1}^{m} \mathrm{sign}(y_i y_i')$ is the Hamming metric, the problem is referred to as *similarity-preserving hashing*. Here, we limit our attention to affine embeddings of the form

$$\mathrm{y} = \mathrm{sign}(\mathbf{P}\mathrm{x} + \mathbf{t}) \;, \tag{1}$$

where $\mathbf{P}$ is an $m \times n$ matrix and $\mathbf{t}$ is an $m \times 1$ vector. Our goal is to find such $\mathbf{P}$ and $\mathbf{t}$ that minimize one of the following cost functions,

$$\begin{aligned} L_{\mathrm{c}}(\mathbf{P},\mathbf{t}) &= \mathbb{E}\{\mathrm{y}(\mathrm{x})^{\mathrm{T}}\mathrm{y}(\mathrm{x}^{-}) - \alpha \mathrm{y}(\mathrm{x})^{\mathrm{T}}\mathrm{y}(\mathrm{x}^{+})\}, \text{ or} \\ L_{\mathrm{d}}(\mathbf{P},\mathbf{t}) &= \mathbb{E}\{\alpha\|\mathrm{y}(\mathrm{x}) - \mathrm{y}(\mathrm{x}^{+})\|^2 - \|\mathrm{y}(\mathrm{x}) - \mathrm{y}(\mathrm{x}^{-})\|^2\} \end{aligned}$$

for $(\mathrm{x},\mathrm{x}^{+}) \in \mathscr{P}$ and $(\mathrm{x},\mathrm{x}^{-}) \in \mathscr{N}$. Both cost functions try to map positives as close as possible to each other (expressed as large correlations or small distance), and negatives as far as possible from each other (small correlation or large distance), in order to ensure low false positive (FPR) and false negative (FNR) rates. $\alpha > 0$ is a parameter determining the tradeoff between the FPR and FNR. In practice, the expectations are approximated as means on some sufficiently large training set.

The problem $\min_{\mathbf{P},\mathbf{t}} L(\mathbf{P},\mathbf{t})$ is a non-linear non-convex optimization problem without an obvious simple solution. It is commonly approached by the following two-stage relaxation: first, approximate the map $\mathrm{y} \approx \mathbf{P}\mathrm{x}$ by removing the sign and the offset vectors, minimizing

$$\begin{aligned} \hat{L}_{\mathrm{c}}(\mathbf{P}) &= \mathbb{E}\{(\mathbf{P}\mathrm{x})^{\mathrm{T}}(\mathbf{P}\mathrm{x}^{-}) - \alpha(\mathbf{P}\mathrm{x})^{\mathrm{T}}(\mathbf{P}\mathrm{x}^{+})\}, \text{ or} \\ \hat{L}_{\mathrm{d}}(\mathbf{P}) &= \mathbb{E}\{\alpha\|\mathbf{P}(\mathrm{x} - \mathrm{x}^{+})\|^2 - \|\mathbf{P}(\mathrm{x} - \mathrm{x}^{-})\|^2\} \end{aligned}$$

w.r.t. to $\mathbf{P}$ (introducing some regularization, e.g., $\mathrm{P}^{\mathrm{T}}\mathrm{P} = \mathrm{I}$, in order to avoid a trivial solution $\mathrm{P} = 0$). Second, fix $\mathbf{P}^{*} = \arg\min_{\mathbf{P}} \hat{L}(\mathbf{P})$ and solve $\mathbf{t}^{*} = \arg\min_{\mathbf{t}} L(\mathbf{P}^{*},\mathbf{t})$ w.r.t. $\mathbf{t}$. To further simplify the problem, it is also common to assume *separability*, thus solving independently for each dimension of the hash.

3.1 Similarity-Sensitive Hashing (SSH)

In [33], the above strategy was used for the approximate minimization of the cost L_{c}. The computation of optimal parameters $\mathbf{P}$ and $\mathbf{t}$ was posed as a boosted binary classification problem, where $d_{\mathbb{H}}(\mathrm{y},\mathrm{y}')$ acts as a strong binary classifier, and each dimension of the linear projection $\mathrm{sign}(\mathbf{p}_i\mathrm{x} + t_i)$ is considered a weak classifier (here, $\mathbf{p}_i$ denotes the ith row of $\mathbf{P}$). This way, AdaBoost can be used to find a greedy approximation of the minimizer of L_{c} by progressively constructing $\mathbf{P}$ and $\mathbf{t}$. At the i-th iteration, the i-th row of the matrix $\mathbf{P}$ and the i-th element of the vector $\mathbf{t}$ are found minimizing a weighted version of L_{c}. Since the problem is non-linear, such an optimization is a challenging problem. In [33], random projection directions were used. A better method for projection selection similar to linear discriminative analysis (LDA) was proposed [7, 8]. Weights of false positive and false negative pairs are increased, and weights of true positive and true negative pairs are decreased, using the standard AdaBoost reweighting scheme [11].

3.2 Covariance Difference Hashing (Diff-Hash)

Strecha *et al.* [34] used the following relaxation of the problem. First, a simplified problem without the sign non-linearity and threshold is solved for projection matrix P,

$$\begin{aligned}\min_{\mathrm{P}^{\mathrm{T}}\mathrm{P}=\mathrm{I}} \; & \mathbb{E}\{(\mathrm{Px})^{\mathrm{T}}(\mathrm{Px}^-)|(\mathrm{x},\mathrm{x}^-)\in\mathscr{N}\} \\ & -\alpha\mathbb{E}\{(\mathrm{Px})^{\mathrm{T}}(\mathrm{Px}^+)|(\mathrm{x},\mathrm{x}^+)\in\mathscr{P}\}.\end{aligned} \tag{2}$$

The constraint $\mathrm{P}^{\mathrm{T}}\mathrm{P}=\mathrm{I}$ is rather arbitrary and required to avoid the trivial solution $\mathrm{P}=0$. This problem can be rewritten as

$$\min_{\mathrm{P}^{\mathrm{T}}\mathrm{P}=\mathrm{I}} \mathrm{tr}\,(\mathrm{P}^{\mathrm{T}}(\Sigma_- - \alpha\Sigma_+)\mathrm{P}), \tag{3}$$

where $\Sigma_\pm$ denote the $n\times n$ covariance matrices of the positive and negative data, respectively. The solution of (2) is given explicitly as

$$\mathrm{P}=[\lambda_1^{1/2}\mathrm{v}_1,\ldots,\lambda_m^{1/2}\mathrm{v}_m]^{\mathrm{T}}=\Lambda_m^{1/2}\mathrm{V}_m,$$

the m smallest eigenvectors of the matrix $\Sigma_- - \alpha\Sigma_+ = \mathrm{V}\Lambda\mathrm{V}^{\mathrm{T}}$ of weighted covariance differences (here we assume eigenvalues and corresponding eigenvectors are sorted in increasing order). The relaxed problem is thus separable and can be solved separately in each dimension (in particular, adding dimension $m+1$ amounts to taking the next eigenvector and does not require recomputing the previous dimensions).

Second, fixing the projection $\mathrm{P}=\Lambda_m^{1/2}\mathrm{V}_m$, find the threshold vector by solving

$$\begin{aligned}\min_{\{a_i\}} \; & \textstyle\sum_{i=1}^m \mathbb{E}\{\mathrm{sign}(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i)\mathrm{sign}(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i)|\mathscr{N}\} \\ & -\alpha\textstyle\sum_{i=1}^m \mathbb{E}\{\mathrm{sign}(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i)\mathrm{sign}(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i)|\mathscr{P}\}.\end{aligned}$$

The problem is separable and can be solved independently in each dimension i. The terms in the problem can be identified with the false positive and negative rates as function of the threshold a_i,

$$\begin{aligned}\mathrm{FNR}(a_i) &= \Pr(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i<0 \text{ and } \mathrm{p}_i^{\mathrm{T}}\mathrm{x}'+a_i>0|\mathscr{P}) \\ &+ \Pr(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i>0 \text{ and } \mathrm{p}_i^{\mathrm{T}}\mathrm{x}'+a_i<0|\mathscr{P})\end{aligned}$$

and

$$\begin{aligned}\mathrm{FPR}(a_i) &= \Pr(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i<0 \text{ and } \mathrm{p}_i^{\mathrm{T}}\mathrm{x}'+a_i<0|\mathscr{N}) \\ &+ \Pr(\mathrm{p}_i^{\mathrm{T}}\mathrm{x}+a_i>0 \text{ and } \mathrm{p}_i^{\mathrm{T}}\mathrm{x}'+a_i>0|\mathscr{N}).\end{aligned}$$

The above probabilities can be estimated from histograms (cumulative distributions) of $\mathrm{p}_i^{\mathrm{T}}\mathrm{x}$ and $\mathrm{q}_i^{\mathrm{T}}\mathrm{y}$ on the positive and negative sets. The optimal threshold is obtained by means of one-dimensional search,

$$\min_{a} \alpha \mathrm{FNR}(a) + \mathrm{FPR}(a). \tag{4}$$

3.3 LDAHash

A similar method was derived in [34] by transforming the coordinates as $\mathrm{C}_{-}^{-1/2}\mathrm{x}$, which allows to write $\min_{\mathbf{P}} \hat{L}_{\mathrm{d}}(\mathbf{P})$ as

$$\min_{\mathrm{P}} \mathrm{tr}\{\mathrm{P}(\mathrm{C}_{+}\mathrm{C}_{-}^{-1})\mathrm{P}^{\mathrm{T}}\} \text{ s.t. } \mathrm{P}^{\mathrm{T}}\mathrm{P} = \mathrm{I}. \tag{5}$$

This approach resembles linear discriminant analysis (LDA), hence the name LDA-hash. Requiring an orthonormal projection matrix P, the problem has a separable closed-form solution consisting of the m smallest eigenvectors of $(\mathrm{C}_{+}\mathrm{C}_{-}^{-1})$.

4 Neural Network Hashing (NNhash)

The problem of existing and most successful similarity-preserving hashing approaches such as LDA- or diff-hash is that they do not solve the optimization problem $\min_{\mathbf{P},\mathbf{t}} L(\mathbf{P},\mathbf{t})$ but rather its relaxation. As a result, the parameters $\mathbf{P}^{*},\mathbf{t}^{*}$ found by these methods in the aforementioned two-stage separable scheme is suboptimal, i.e., $L(\mathbf{P}^{*},\mathbf{t}^{*}) > \min L$. Our experience shows that in some cases, the suboptimality is dramatic (at least an order of magnitude).

A way of solving the 'true' optimization problem is by formulating it in the neural network (NN) framework and exploiting numerous optimization techniques and heuristics developed in this field. Since we have a way of cheaply producing labeled data, we decide to adopt the *siamese network* architecture [32, 14] which, contrary to conventional models, receives two input patterns and minimizes a loss function similar to equation (2),

$$L_{\mathrm{nn}}(\mathbf{P},\mathbf{t}) = \frac{1}{2}\|\mathrm{y}(\mathrm{x}) - \mathrm{y}(\mathrm{x}^{+})\|^2 + \frac{1}{2}(\max\{0, m - \|\mathrm{y}(\mathrm{x}) - \mathrm{y}(\mathrm{x}^{-})\|\})^2, \tag{6}$$

where the constant m represents the margin between dissimilar pairs. The margin is introduced as regularization to avoid the system from minimizing the loss just pulling two vector as far apart as possible. The embedding is then learned to make positive pairs as close as possible and negative pairs at least at distance m.

Network architecture of this type can be traced back to the work of Schmidhuber and Prelinger [32] on problems of predictable classification. In [14], siamese networks were used to learn an invariant mapping of tiny images directly from pixel representation, whereas in [36] a similar approach is used to learn a model that is highly effective at matching people in similar pose which exhibits invariance to identity, clothing, background, lighting, shift and scale. An advantage of such architecture is that one can create arbitrarily complex embeddings by simply stacking many layers in the network. In all our experiments, in order to make a fair comparison to other hashing methods, we adopt a simple single layer architecture, wherein $\mathrm{y}(\mathrm{x}) = \mathrm{sign}(\mathbf{P}\mathrm{x} + \mathbf{t})$. Network training attempts to find $\mathbf{P}, \mathbf{t}$ that minimize L_{nn} (which is a regularized version of L_{d}). Since we solve a non-linear problem without introducing any simplification or relaxation, the results are expected to be better compared to hashing methods described in Section 3. In the following, we refer to our method as *NNhash*.

Since a binary output is required, we adopt $\tanh(\beta t) \approx \mathrm{sign}(t)$ as the non-linear activation function for our siamese network, which enforces binary vectors when either m or the steepness β of the function is increased. Since the problem is highly non-convex, it is liable to local convergence, and thus there is no theoretical guarantee to find the global minimum. However, by initializing $\mathbf{P}, \mathbf{t}$ by the solution obtained by one of the standard hashing methods, we have a good initial point that can be improved by network optimization,

5 Results

5.1 Data

In our experiments, we used the Rawseeds dataset [4, 9]. The dataset contained video sequences of a robot equipped with an omnidirectional camera system based on a parabolic mirror moving in an indoor and outdoor scene. The image undergoes significant distortion since different parts of the scene move from the central part of the mirror to the boundaries.

We used the toolbox of Vedaldi [39] to compute SIFT features in each frame of the video. Since the robot movement is slow, the change between two adjacent frames in the dataset is infinitesimal, and SIFT features can be matched reliably. Tracking features for multiple frames, we constructed the positive set as the transitive closure of these adjacent feature descriptor pairs. This way, the positive set included also descriptors distant in time, and, as a result of robot motion located at different regions in the image and thus subject to strong distortions. As negatives, we used features not belonging to the same track.

In addition to the Rawseeds dataset, we created synthetic omnidirectional datasets using panorama images that were warped simulating the effect of a parabolic mirror. The warping intentionally was not the same as in Rawseeds dataset. By moving the panorama image, we created synthetic motion with known pixel-wise groundtruth correspondence (Figure 5). The positive and negative sets for synthetic data were constructed as described above.

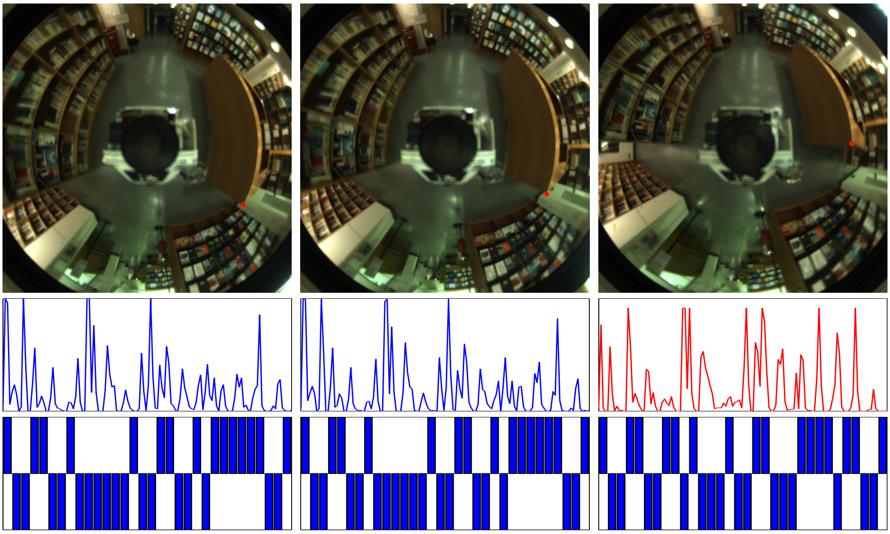


Fig. 1 A few frames from the Rawseeds dataset examplifying how a descriptor changes over time due to camera motion throughout the scene. First row: omnidirectional images of the indoor dataset, shown at times 1 (left), 5 (middle) and 50 (right). Second row: SIFT descriptors at point indicated in red. Third row: binary descriptors of length 32 produced by NNhash trained on outdoor images.

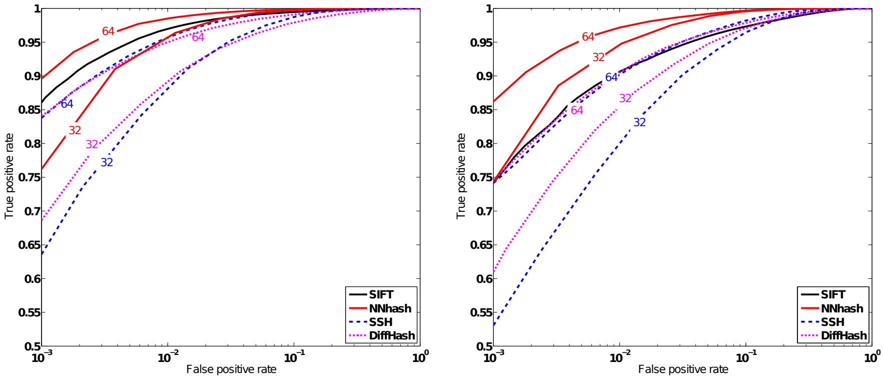
5.2 *Methods*

We compared the SSH [33], diff-hash [34], and our NNhash methods. For the NNhash training we used scaled conjugate gradient over the whole batch of descriptors, which we normalize in the range $[-1..1]$. We used a margin $m = 5$ in all cases. The steepness factor for tanh is $\beta = 1$ in the case of 32 bit while for 64 bit we gradually increased it up to 3 so to have a smooth binarization. We reached convergence in about 50 epochs in all cases.

5.3 *Performance Degradation in Time*

For this experiment, we constructed the training set using descriptors extracted from about 300 consecutive frames of the outdoor sequence (similar results were obtained when using outdoor or synthetic data for training). We considered descriptors that could be tracked for at least 60 consecutive frames and selected as positives pairs of descriptors belonging to these tracks.

To avoid bias, we selected pairs of descriptors in frames t_i, t_j in such a way that the time difference $\Delta t = |t_i - t_j|$ between the frames was uniformly distributed. The training was performed on a positive set of size 10^5 and on a negative set of size 10^6 to produce hashes of length 32 and 64 bits.



$10 \leq \Delta t \leq 30$ $20 \leq \Delta t \leq 40$

Fig. 2 ROC curve for the outdoor dataset, with frames taken at various distance Δt. Each hashing method is shown with 32 and 64 bits. Note significant performance degradation of SIFT and only minor performance degradation of NNhash.

Table 1 Descriptor matching performance using different methods and descriptor size for frames with time range $10 \leq \Delta t \leq 30$.

	m	**EER**	**FPR@1%**	**FPR@0.1%**
SIFT	1024	1.91%	3.08%	13.87%
NNhash	32	**1.66%**	3.77%	23.81%
	64	**1.31%**	**1.92%**	**9.48%**
DiffHash	32	4.41%	9.36%	29.95%
	64	2.57%	5.17%	18.30%
SSH	32	4.02%	15.64%	36.41%
	64	2.22%	4.90%	16.74%

Table 2 Descriptor matching performance using different methods and descriptor size for frames with time range $20 \leq \Delta t \leq 40$.

	m	**EER**	**FPR@1%**	**FPR@0.1%**
SIFT	1024	3.31%	7.47%	27.94%
NNhash	32	**2.70%**	**6.98%**	**24.98%**
	64	**2.38%**	**4.54%**	**14.22%**
DiffHash	32	5.17%	12.55%	37.49%
	64	3.69%	8.75%	27.34%
SSH	32	5.52%	24.10%	47.29%
	64	3.46%	9.48%	27.66%

Testing was performed on a different portion of the same sequence, where frames at distance $10 \leq \Delta t \leq 30$ (Figure 2, left) and $20 \leq \Delta t \leq 40$ (Figure 2, right) were used. A few phenomena can be observed in Figure 2 showing the ROC curves of straightforward SIFT matching using the Euclidean distance and matching of learned binary descriptors using the Hamming distance. First, we can see that even

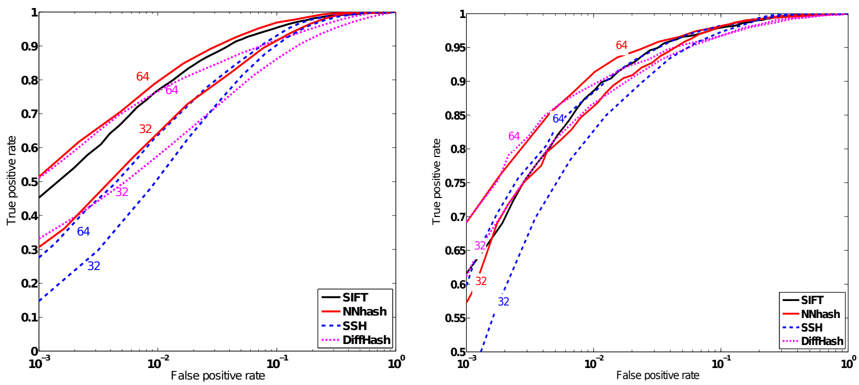
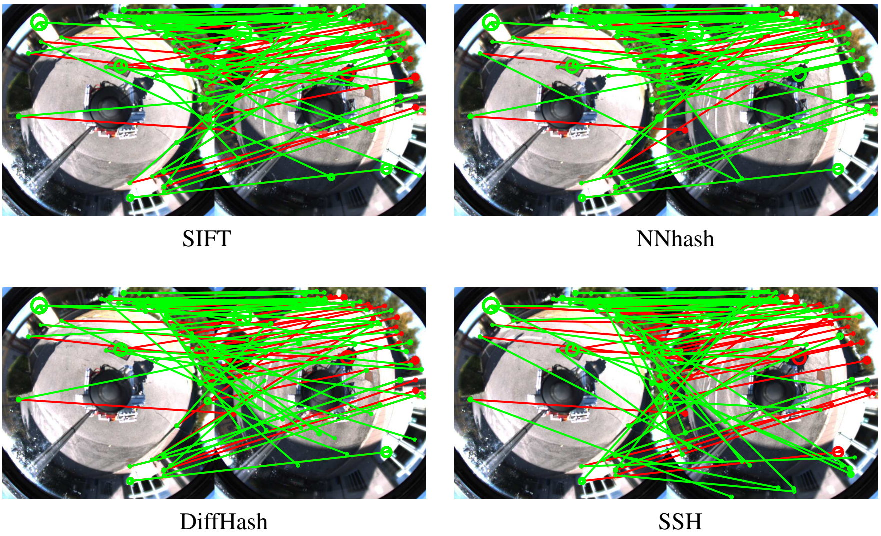


Fig. 3 Left: ROC curve for the models trained on outdoor data and tested on indoor data with descriptors taken at $35 \leq \Delta t \leq 60$. Right: ROC curve for synthetic trained models. Testing performed on indoor real descriptors.



SIFT

NNhash

DiffHash

SSH

Fig. 4 Visual comparison of the matches produced on outdoor data with $\Delta t = 70$. Ground truth matches are plotted in red and descriptor matches (1-closest) in green. Ideally (if matching completely coincides with the groundtruth), only green lines should be visible. Interesting matches appear on the bottom-left portion of the image where NNhash learns invariance to high distortions.

with very compact descriptors (as small as 64 bit, compared to 1024 bit required to represent SIFT) we match or outperform SIFT. These results are consistent with the study in [34]. Second, we observe that NNhash significantly outperforms other hashing methods for the same number of bits. This is a clear indication that SSH

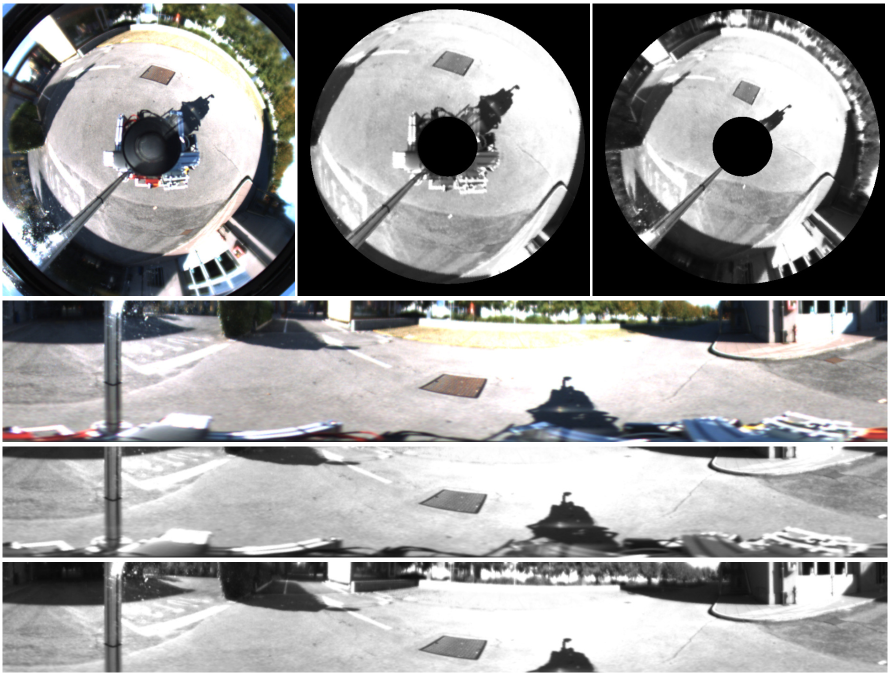


Fig. 5 Illustrative example of how synthetic data is generated. First row from left to right: the original omnidirectional image, the synthetic image from the first shift of 5 pixels, the synthetic image after 14 vertical shifts. Second to fourth rows: unwarped panorama images generated from images in the first row.

and diff-hash methods are finding a suboptimal solution by solving a relaxed problem, while NNhash attempts to solve the full non-linear non-convex optimization problem.

Comparing Figure 2 (left and right) and Tables 1–2, we can observe how the matching performance degrades if we increase the time between the frames (from $10-30$ frames to $20-40$ frames). Because of significant distortions caused by the parabolic mirror, objects moving around the scene appear differently. This phenomenon is especially noticeable when the distance between the frames (Δt) is large. SIFT shows significant degradation, while NNhash, trained on a dataset including positive pairs at distances up to $\Delta t = 60$ degrades only slightly (even a 32-bit NNhash performs better than SIFT). This is a clear indication that we are able to learn feature invariance.

Finally, Figure 4 shows a visual example of feature matching using different methods. NNhash produces matches most similar to the groundtruth (shown in green).

5.4 Generalization

To test for generalization we perform experiments of transfer learning from outdoor data to indoor data and from synthetic data to real data.

Figure 3-left shows the performance of descriptors trained on outdoor and tested on indoor data. We can see that even though the data used for training is very different from the one used for testing (i.e. see Figure 1 and Figure 4 for a visual comparison) we achieve better performance than SIFT with just 64 bits. Figure 3-right shows the performance of descriptors trained on synthetic and tested on indoor data. All learning methods perform better than SIFT. The discrepancy between NNhash and the other algorithms is less pronounced that in the real case.

6 Discussion, Conclusions, and Future Work

We presented a new approach for feature matching in omnidirectional images based on similarity-sensitive hashing and inspired by the recent work [34]. We learn a mapping from the descriptor space to the space of binary vectors that preserves the similarity of descriptors on a training set. By carefully constructing the training set, we account for descriptor variability, e.g. due to optical distortions. The resulting descriptors are compact and are compared using the Hamming metric, offering significant computational advantage over other traditional metrics such as L_2. Though tested with SIFT descriptors, our approach is generic and can be applied to any feature descriptor.

We compared several existing similarity-preserving hashing methods, as well as our NNhash method based on a neural network. Experimental results show that NNhash outperforms other approaches. An explanation to this behavior is the fact that of today's state-of-the-art similarity-preserving hashing algorithms like SSH or LDAHash solve a simplified optimization problem, whose solution does not necessarily coincide with the solution of the "true" non-linear non-convex problem. We showed that using a neural network, we can solve the "true" problem and yield better performance.

Finally, our discussion in this paper was limited to simple embeddings of the form $\mathrm{sign}(\mathbf{P}x + \mathbf{t})$ which in some cases are too simple. The neural network framework seems to us a very natural way to consider more generic embeddings using multi-layer network architectures.

Acknowledgements. M. B. is partially supported by the Swiss High Performance and High Productivity Computing (HP2C) grant. J. M. is supported by Arcelor Mittal Maizieres Research SA. D. M. is partially supported by the EU projects FP7-ICT-IP-231722 (IM-CLeVeR).

References

1. Athitsos, V., Alon, J., Sclaroff, S., Kollios, G.: Boostmap: a method for efficient approximate similarity ranking. In: Proc. CVPR (2004)
2. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: SURF: Speeded Up Robust Features. Computer Vision and Image Understanding 10(3), 346–359 (2008)

3. Bogdanova, I., Bresson, X., Thiran, J.P., Vandergheynst, P.: Scale space analysis and active contours for omnidirectional images. Trans. Image Processing 16(7), 1888–1901 (2007), doi:10.1109/TIP.2007.899008
4. Bonarini, A., Burgard, W., Fontana, G., Matteucci, M., Sorrenti, D.G., Tardos, J.D.: Rawseeds: Robotics advancement through web-publishing of sensorial and elaborated extensive data sets. In: Proc. IROS Workshop on Benchmarks in Robotics Research (2006), http://www.robot.uji.es/EURON/en/iros06.htm
5. Briggs, A., Li, Y., Scharstein, D., Wilder, M.: Robot navigation using 1d panoramic images. In: Proc. ICRA (2006)
6. Bronstein, A., Bronstein, M., Ovsjanikov, M., Guibas, L.: Shape Google: geometric words and expressions for invariant shape retrieval. ACM TOG (2010)
7. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Video genome. Tech. Rep. arXiv:1003.5320v1 (2010)
8. Bronstein, M.M., Bronstein, A.M., Michel, F., Paragios, N.: Data fusion through cross-modality metric learning using similarity-sensitive hashing. In: Proc. CVPR (2010)
9. Ceriani, S., Fontana, G., Giusti, A., Marzorati, D., Matteucci, M., Migliore, D., Rizzi, D., Sorrenti, D.G., Taddei, P.: Rawseeds ground truth collection systems for indoor self-localization and mapping. Autonomous Robots 27(4), 353–371 (2009), http://www.springerlink.com/content/k924032g72818h53/, doi:10.1007/s10514-009-9156-5
10. Cruz, J., Bogdanova, I., Paquier, B., Bierlaire, M., Thiran, J.P.: Scale invariant feature transform on the sphere: Theory and applications. Tech. rep. (2009), http://transp-or2.epfl.ch/technicalReports/CruzBogdPaquBierThir09.pdf
11. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In: European Conference on Computational Learning Theory, pp. 23–37 (1995)
12. Geyer, C., Stewenius, H.: A nine-point algorithm for estimating para-catadioptric fundamental matrices. In: Proc. CVPR (2007)
13. Gionis, A., Indik, P., Motwani, R.: Similarity Search in High Dimensions via Hashing. In: International Conference on Very Large Databases (2004)
14. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: Proc. CVPR (2006)
15. Hansen, P.I., Corke, P., Boles, W.: Wide-angle visual feature matching for outdoor localization. Int. J. Robotics Research 29(2/3), 267–297 (2010), http://eprints.qut.edu.au/33736/
16. Jain, P., Kulis, B., Grauman, K.: Fast image search for learned metrics. In: CVPR (2008)
17. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)
18. Jégou, H., Douze, M., Schmid, C.: Packing Bag-of-Features. In: Proc. ICCV (2009)
19. Jégou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. Trans. PAMI (2010)
20. Kimmel, R., Zhang, C., Bronstein, A.M., Bronstein, M.M.: Are mser features really interesting? Trans. PAMI 32(11), 2316–2320 (2011)
21. Kulis, B., Darrell, T.: Learning to hash with binary reconstructive embeddings. In: Proc. NIPS, pp. 1042–1050 (2009)
22. Lowe, D.: Distinctive Image Features from Scale-Invariant Keypoints. IJCV 20(2), 91–110 (2004)

23. Mauthner, T., Fraundorfer, F., Bischof, H.: Region matching for omnidirectional images using virtual camera planes. Technology (2006)
24. Mei, C., Rives, P.: Single view point omnidirectional camera calibration from planar grids. In: Proc. ICRA (2007)
25. Micusik, B., Pajdla, T.: Structure from motion with wide circular field of view cameras. Trans. PAMI 28(7), 1135–1149 (2006), doi:10.1109/TPAMI.2006.151
26. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. IJCV 65(1/2), 43–72 (2005)
27. Nayar, S.K.: Catadioptric Omnidirectional Camera. In: Proc. CVPR (1997)
28. Puig, L., Guerrero, J.J.: Scale space for central catadioptric systems. towards a generic camera feature extractor. In: Proc. ICCV (2011)
29. Raginsky, M., Lazebnik, S.: Locality-Sensitive Binary Codes from Shift-Invariant Kernels. In: Proc. NIPS (2009)
30. Salakhutdinov, R., Hinton, G.: Semantic hashing. In: SIGIR Workshop on Information Retrieval and applications of Graphical Models (2007)
31. Scaramuzza, D., Siegwart, R., Martinelli, A.: A robust descriptor for tracking vertical lines in omnidirectional images and its use in mobile robotics. Int. J. Robotics Research 28(2), 149–171 (2009), `http://ijr.sagepub.com/cgi/doi/10.1177/0278364908099858`
32. Schmidhuber, J., Prelinger, D.: Discovering predictable classifications. Neural Computation 5(4), 625–635 (1993)
33. Shakhnarovich, G.: Learning Task-Specific Similarity. Ph.D. thesis, MIT (2005)
34. Strecha, C., Bronstein, A.M., Bronstein, M.M., Fua, P.: LDAHash: improved matching with smaller descriptors. Trans. PAMI (2011)
35. Svoboda, T., Pajdla, T.: Matching in catadioptric images with appropriate windows, and outliers removal. In: Skarbek, W. (ed.) CAIP 2001. LNCS, vol. 2124, pp. 733–740. Springer, Heidelberg (2001)
36. Taylor, G.W., Spiro, I., Bregler, C., Fergus, R.: Learning invariance through imitation. In: Proc. CVPR (2011)
37. Tola, E., Lepetit, V., Fua, P.: Daisy: an Efficient Dense Descriptor Applied to Wide Baseline Stereo. Trans. PAMI 32(5), 815–830 (2010)
38. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: a large dataset for nonparametric object and scene recognition. Trans. PAMI 30(11), 1958–1970 (2008)
39. Vedaldi, A.: An open implementation of the SIFT detector and descriptor. Tech. Rep. 070012, UCLA CSD (2007)
40. Wang, J., Kumar, S., Chang, S.F.: Semi-supervised hashing for scalable image retrieval. In: CVPR (2010)
41. Wang, J., Kumar, S., Chang, S.F.: Sequential projection learning for hashing with compact codes. In: ICML (2010)
42. Weiss, Y., Torralba, A., Fergus, R.: Spectral hashing (2009)

Visual Correspondence, the Lambert-Ambient Shape Space and the Systematic Design of Feature Descriptors

Stefano Soatto and Jingming Dong

Abstract. In this expository article, we justify the use of sparse local descriptors for correspondence, and illustrate a systematic method for their design. Correspondence is the process that allows using image data to infer properties of the "scene," where the scene can refer to a specific object or landscape, or can be abstracted into a category label to take into account intra-class variability. As the generality increases, the complexity of nuisance factors does too, so global pixel-level correspondence is not viable, and one has to settle instead for sparse descriptors. These should be co-designed with the classifier, and for a given classifier family, one can design the descriptors to be invariant to uninformative nuisances that are explicitly modeled, insensitive to other nuisances that are not explicitly modeled, and maximally discriminative, relative to the chosen family of classifiers. Existing descriptors are interpreted in this framework, where their limitations are illustrated, together with pointers on how to improve them.

1 Visual Correspondence

Correspondence is the process of attributing properties of different images to the same underlying "scene." Depending on what one means by "scene," various nuisance factors contribute to the variability that different images exhibit, so establishing correspondence largely entails dealing with the effects of such nuisances.

In the simplest case where the "scene" is some static layout of surfaces ("objects") and the images are taken from nearby vantage points, under constant illumination and camera rotation, different images are related by a simple (global projective) transformation. Other than around the boundary, every point in one image can be put into one-to-one correspondence with a point in another image, and multiple images can be collated into a *mosaic* (Fig. 1). The same goes for arbitrary

Stefano Soatto · Jingming Dong
University of California, Los Angeles
e-mail: soatto@ucla.edu

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_4, © Springer-Verlag Berlin Heidelberg 2014

camera motion in three-dimensional (3-D) space when the scene is planar [15] (Fig. 1). In both cases, the only nuisance variability is a projective deformation of the image domain.[1]

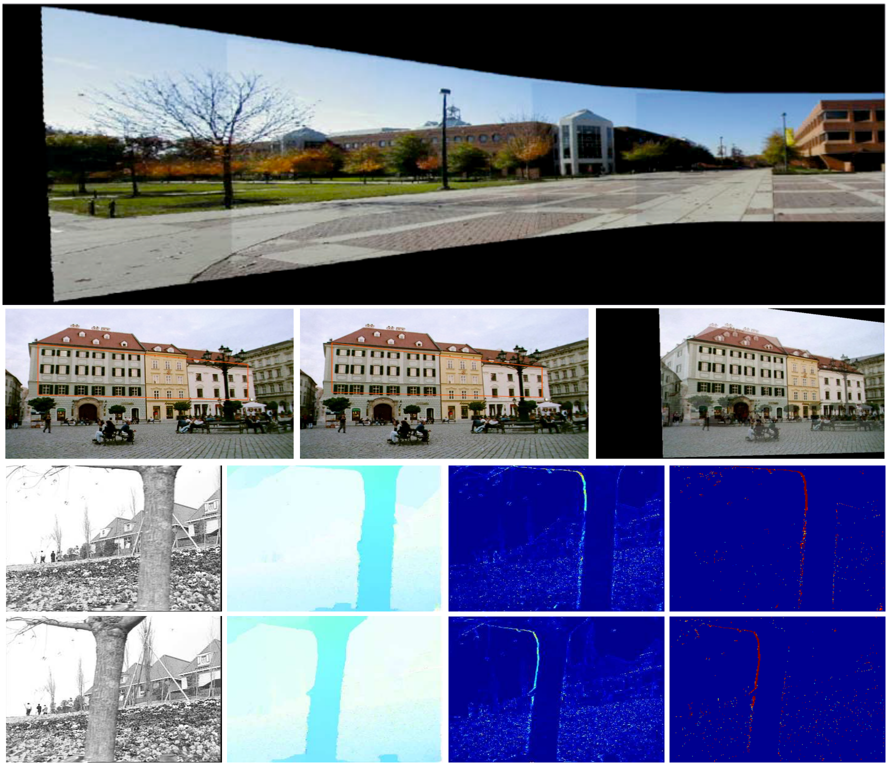


Fig. 1 Viewpoint changes induce relatively simple nuisance transformations of the image only when the viewpoint changes are small relative to the distance from the scene (for instance, pure rotational motion, top from [15]), or when the scene is planar (middle, from [15]). In both cases, occlusion phenomena are negligible or absent, and the deformation of the image domain can be represented by a global projective transformation. Even when parallax is present (bottom), one can establish (possibly non-unique) correspondence between every point on the co-visible domain (from [2]).

However, as soon as there is some *parallax* (the translation of the optical center, or *baseline*, is non-zero) and the scene is not flat, the domain deformation cannot be captured by a simple (global, finite-dimensional, invertible) transformation. Nevertheless, for sufficiently simple scenes and sufficiently small motions, it is possible

[1] For tutorial material on projective transformations, see Chapter 3 of [15].

to determine a domain deformation that brings *co-visible* points into one-to-one correspondence (Fig. 1 bottom). In fact, typically there are infinitely many such deformations, and one can impose additional conditions to choose among them (*regularization*). The linear component of such a deformation is called *optical flow*, and can be determined along with the collection of points that are visible in one image but *occluded* in another [2] (Fig. 2). If the scene is static, establishing correspondence is equivalent to reconstructing a range (depth) map of the scene [22].

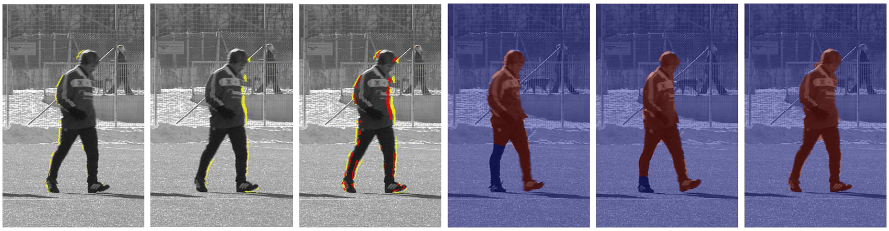


Fig. 2 Occluded regions are regions of one image that back-project onto portions of the scene that are not visible in another image. They can be determined by comparing multiple images of the same scene, and inferred together with the domain deformation w or its linear component (optical flow) by solving a convex variational optimization problem [3].

When the baseline is large,[2] however, occlusion phenomena become ubiquitous and the optical flow becomes increasingly complex.[3] Rather than modeling correspondence as a very complex function defined globally on the image domain, as customary in functional approximation one can partition the domain into regions or *segments* each of which independently transforms via a relatively simple map, for instance a projective, affine, or similarity transformation. Because correspondence cannot be established for large portions of the image domain [9] (the *aperture problem*, addressed by regularization in the short-baseline case), such correspondence can typically be established only for a relatively small set of regions, usually of relatively small size.

Even when not susceptible to the aperture problem, because global consistency is not enforced, there are typically multiple regions that could potentially correspond, in the sense that they "look similar" – up to a small-dimensional transformation – in different images. More importantly, the actual correspondent may not be one of them. Because of changes of illumination, or violation of the Lambertian assumption implicit in correspondence [2], actual intensity values at or around corresponding points can be rather different. So, it is customary to include among nuisance factors not only (geometric) transformations of the domain of the image, but also

[2] The distance between the optical centers is large relative to the distance to the scene.

[3] In fact, as complex as an arbitrary infinite-dimensional homeomorphism of the image domain [23]. A homeomorphism is a continuous and invertible transformation of the plane, with continuous inverse.

(photometric) transformations of its range. For instance, photometric changes can be modeled locally as a monotonic continuous (a.k.a. *contrast*) transformation.

The situation is even less predictable when the "scene" is not static but can instead deform. In this case, correspondence based on local properties of the image can at best be hypothesized, but any hypothesis would have to be validated against a model of the intrinsic deformation of the scene or objects within. Worse yet, such a "deformation" can be geometric (as in a deformable object) or photometric, when the "scene" is not a specific object, but a category label, where each object within the category can exhibit intrinsic photometric variability.

In this case, two or more images are in correspondence if they portray the same "scene," that is if they portray (possibly different) objects belonging to the same category. In addition to nuisance variability, which we seek to discount or eliminate, we then have intrinsic variability, often represented in the form of a training set, against which we must test our correspondence hypothesis.

Correspondence then is naturally framed as a statistical hypothesis testing problem, whereby images provide a mechanism for hypothesis generation (bottom-up) and models or assumptions provide a mechanism for validation (top-down). Among the nuisance variability that we seek to eliminate in the hypothesis generation process are domain deformations due to changes of viewpoint, and range deformations due to illumination. Among the assumptions that we may want to enforce in the validation stage are *co-visibility* (a basic requirement of all forms of correspondence), rigidity and photometric equivalence up to contrast (if we are interested in a specific static object or scene) and intrinsic photometric and geometric variability (if we are interested in a deforming object or in a category of objects).

In this expository paper we formalize the notions above – that serve to motivate the necessity for local feature detection and invariant description – in a manner that allows the reader not only to *use* existing feature descriptors, but also compare them on analytical grounds, and hopefully design better ones.

2 Nuisance Groups, Orbits and Shape Spaces

In this tutorial section, we deal with transformations that exhibit the structure of a group, acting either on the domain or the range of the image. We start with the simplest case of Euclidean (planar) shape spaces, and show how the concepts can be extended to more complex groups.

A group is a set G with an operation (composition) that combines two elements g_1 and g_2 into an element $g_1 \circ g_2 \in G$ of the same set, has an element $e \in G$ (the "null" or "identity") that leaves the operand unchanged $g \circ e = g$, and each element g has an *inverse*, that is another element g^{-1} of the group that, composed with it, gives the null element. A group can *act* on a space X by transforming an element $x \in X$ into another element of the same space. We indicate the action with $g(x)$ or $g \circ x$ or simply gx. Examples include the general linear group $\mathbb{GL}(n)$ of $n \times n$ invertible matrices, acting on $\mathbb{R}^n$, where composition is represented by matrix multiplication.

There are many groups that can act on an image. The simplest is planar translation, due for instance to camera calibration or to a change of viewpoint, that can however generate domain deformations that are as complex as a generic (infinite-dimensional) diffeomorphism.[4] Similarly, a linear or affine transformation of the pixel values can be due to camera auto-gain control or to changes of illumination. Such nuisance variability is typically uninformative, although there may be applications where it is not [21]. The reason for studying group transformations is that they organize the data into *orbits* that are equivalence classes, and can therefore be represented by any one element of the class. In other words, the entire variability due to a group transformation, no matter how complex or high-dimensional, can be collapsed to a single element by choosing a suitable (canonical) representative for each class.

We will first illustrate the concept with triangles, as a special case of classical finite-dimensional shape space [11]. Each triangle can be described by the coordinates of its three vertices, $x_1, x_2, x_3 \in \mathbb{R}^2$, or equivalently by a 2×3 matrix $x = [x_1, x_2, x_3] \in \mathbb{R}^{2\times 3} \sim \mathbb{R}^6$. Therefore, a triangle can be thought of as a "point" in six-dimensional space $X = \mathbb{R}^6$. However, depending on the reference frame with respect to which the coordinates are expressed, we have different coordinates $x \in X$. Indeed, if we "move" the triangle around the plane, its coordinates will describe a trajectory in X, and yet we want to capture the fact that it is the *same* triangle. Shape Spaces are designed to capture precisely this concept: The shape of a configuration is what is preserved regardless of the choice of coordinates, or equivalently regardless of the motion of the object.

Now, even on the plane, one can consider different kinds of coordinates, or equivalently different kinds of motions, or different kind of groups acting on the triangles. For instance, one can consider Euclidean coordinates, or correspondingly rigid motions, whereby the triangles are transformed in such a way as to preserve distances, angles and orientation. In this case, the matrix of coordinates $x \in X$ is transformed via multiplication by a rotation matrix $R \in SE(2)$, that is a matrix of the form $R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ for some $\theta \in [0, 2\pi)$ and the addition of an offset $T \in \mathbb{R}^2$ (a translation vector). So if we indicate with $g = (R, T) \in SE(2)$ the group of rigid motions, and with gx the action of the group on the coordinates, we have that $x' = gx \in \mathbb{R}^{2\times 3}$ and the transformed coordinates are $x_i' = Rx_i + T$. However, we could also consider the *similarity* group where the rectangles are allowed to be scaled, while retaining the angles, in which case $g = (\alpha R, T)$ and points are transformed via $x_i' = \alpha R x_i + T$ for some $\alpha > 0$, or the *affine* group where $R \in \mathbb{GL}(2)$ is an arbitrary 2×2 invertible matrix. In any case, what we want to capture as the "essence" of the triangle $x \in S$ is what remains unchanged as the group acts on the object of interest by transforming its coordinates.

[4] A diffeomorphism of the plane is a continuous and differentiable transformation that has a differentiable inverse. The set of all planar diffeomorphisms is a group.

2.0.1 Orbits

Geometrically, we can think of the group $G \ni g$ acting on the space X by generating *orbits*, that is equivalence classes

$$[x] \doteq \{gx \mid g \in G\}.$$

Different groups will generate different orbits. Remember that an equivalence class is a set that can be *represented* by any of its elements, from which we can construct the entire orbit by acting with the group. As we change the group element g, the coordinates gx change, but what remains constant is the entire orbit $[gx] = [x]$ for any $g \in G$. Therefore, the entire orbit is the object we are looking for; it is the *maximal invariant* to the group G. We now need an efficient way to represent this orbit algebraically, and to *compare* different orbits.

2.0.2 Max-Out

The simplest approach consists of using *any* point along the orbit to represent it. For instance, if we have two triangles we simply describe their "shape" by their coordinates $x, y \in \mathbb{R}^{2\times 3}$. However, when comparing the two triangles we cannot just use any norm in $\mathbb{R}^{2\times 3}$, for instance $d(x,y) = ||x-y||$, lest the same triangle, written in two different reference frames, would have non-zero distance, for instance if $y = gx$, we have $d(x,y) = ||x-gx|| = ||e-g||\,||x||$ which is non-zero so long as the group element g is not the identity e. Instead, when comparing two triangles we have to compare all points on their two orbits,

$$d(x,y) = \min_{g_1,g_2} ||g_1x - g_2y||_{\mathbb{R}^6}.$$

This procedure is called *max-out*, and entails solving an optimization problem every time we have to compute a distance (Fig. 3)

2.0.3 Canonization

As an alternative, since we can represent any orbit with one of its elements, if we can find a *consistent way* to choose a representative of each orbit, perhaps we can then simply compare such representatives, instead of having to search for the shortest distance along the two orbits. The choice of a consistent representative, also known as a *canonical element*, is called canonization. The choice of canonization is not unique, and it depends on the group. The important thing is that it must depend on *each orbit* independently of others. So, if we have two orbits $[x]$ and $[y]$, the canonical representative of $[x]$, call it $\hat{x}$, cannot depend on $[y]$. To gain some intuition on the canonization process, consider again the case of triangles. If we take one of its vertices, for instance x_1, to be the origin, so that $x_1 = 0$, or equivalently $T = -x_1$, and transform the other points to the same reference frame, we have that all triangles are now of the form $[0, x_2 - x_1, x_3 - x_1] = [0, x_2', x_3']$. What we have done is to *canonize* the translation group. The result is that we have eliminated two degrees of freedom,

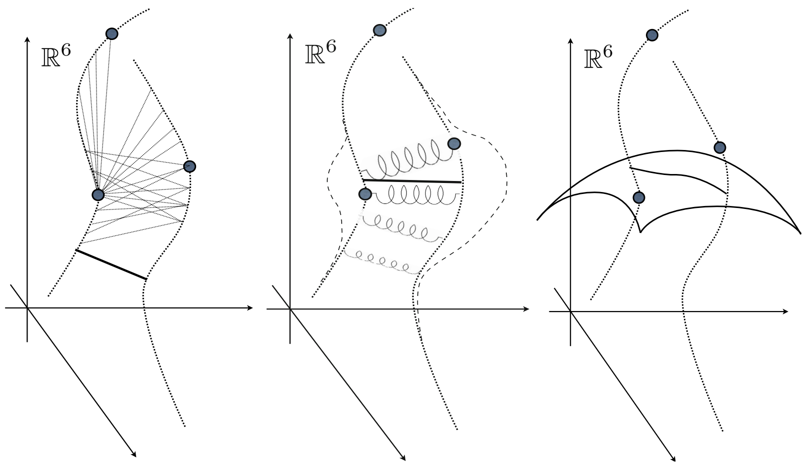


Fig. 3 Pictorial representation of equivalence classes (orbits), and their comparison via max-out (left), marginalization (middle), and canonization (right). In max-out, one has to search along each orbit for the smallest distance between two points. In marginalization, one has to average all distances relative to the strength imposed by the prior. In canonization, each orbit is represented by a canonical element, and all canonical elements live on the "base space" of the orbit space, where they are compared using either a cordal distance (in the embedding space $\mathbb{R}^6$, or more correctly using a geodesic distance, which is the length of the shortest path on the base space.

and now every rectangle can be represented in a *translation-invariant* manner by a 2×2 matrix $[x'_2, x'_3] \in \mathbb{R}^{2\times 2}$.

We can now repeat the procedure to canonize the rotation group, for instance by applying the rotation $R(\theta)$ that brings the point x'_2 onto the horizontal axis. By doing so, we have canonized the rotation group. We can also canonize the scale group, by multiplying by a scale factor α so that the point x_2 not only is on the horizontal axis, but is at distance one from the origin, or equivalently has coordinates $[1,0]^T$. By doing so, we have canonized the scale group, and now every triangle is represented by $x = \left[\begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix}\; \frac{1}{\alpha}R^T(\theta)x'_3\right]$. So, every triangle is represented by a two-dimensional vector $\hat{x} = \frac{1}{\alpha}R^T(\theta)[x_3 - x_1] \in \mathbb{R}^2$. With this procedure, we have canonized the similarity group. Note that all three vertices contribute to these two coordinates, as x_2 is used to compute θ and α. However, each triangle is now represented by just *two* numbers, instead of six that we started with. These two numbers can be easily visualized in a graph (Fig. 4).

If we now want to compare triangles, we can just compare their canonical representative, without solving an optimization problem:

$$d(x,y) = \|\hat{x} - \hat{y}\|_{\mathbb{R}^2}.$$

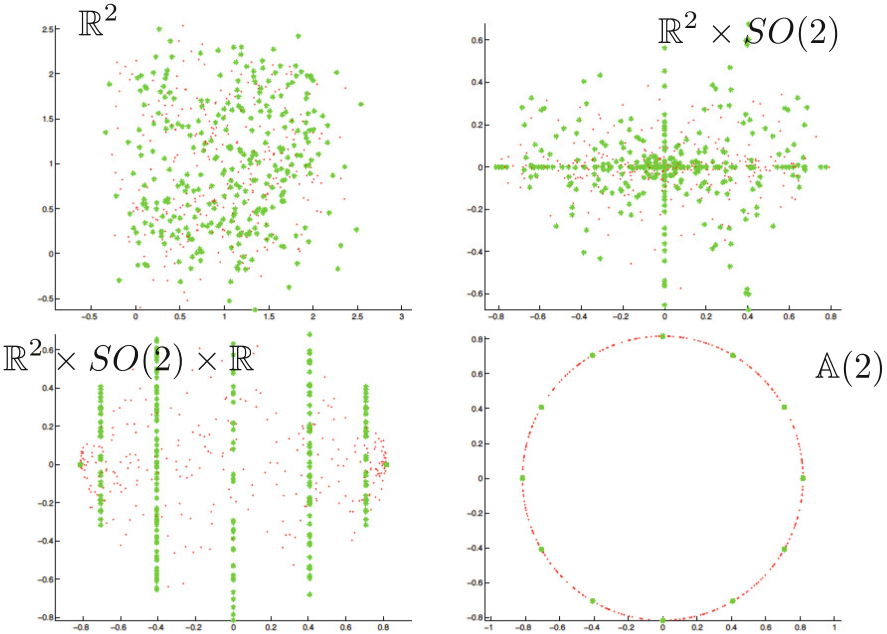


Fig. 4 Similarity shape space of triangles. We show the vertices of random triangles belonging to two classes: Isosceles (green) and scalene (red). On the top-left, we show the coordinates of the vertices after canonizing translations (so one of the vertices is always at the origin). They do not exhibit any discernible characteristic. On the top-right we show the same after canonizing rotation and translation. Here the isosceles appear to exhibit some regularity, as one vertex is always at zero (not shown) and the other is either on the horizontal or vertical axis (the third vertex can be anywhere on the plane). On the bottom-right we show the same after canonizing translation, rotation and scale. Here it is clear that isosceles are distributed on a subset of measure zero and can be easily discriminated. On the bottom-right we show the case where an affine transformation is quotiented out. Both sets distribute on a subset of measure zero of the plane. The multiple periodic repetitions (symmetries) are due to the fact that canonization mechanism chosen is not invariant to permutations.

This is a so-called *cordal distance*; we will describe the more appropriate notion of *geodesic distance* later.

Note that choosing a canonical representative of the orbit $\hat{x}$ is done by choosing a canonical element of the group, that depends on x, $\hat{g} = \hat{g}(x)$, and then *un-doing* the group action,

$$\hat{x} = \hat{g}^{-1}(x)x.$$

It is an easy exercise to show that $\hat{x}$ is now invariant to the group, in the sense that $\widehat{g'x} = \hat{x}$ for any $g' \in G$. This procedure is very general, and we will repeat it in different contexts below. Note that the larger the group that we are canonizing (this procedure is also called *quotienting out*), the smaller the quotient, to the point where the quotient collapses into a single point. Consider for instance the case of triangles where we try to canonize the affine group. By doing so all triangles would become identical, since it is always possible to transform any triangle into any other triangle with an affine transformation.

Geometrically, the canonization process corresponds to choosing a *base* of the orbit space, or computing the *quotient* of the space X with respect to the group G. Consequently, the base space is often written as X/G. Note that the canonical representative $\hat{x}$ lives in the base space that has a dimension equal to the dimension of X minus the dimension of the group G. So, the quotient of triangles relative to the translation group is 4-dimensional, relative to the group of rigid motion it is 3-dimensional, relative to the similarity group it is 2-dimensional, and relative to the affine group it is 0-dimensional. By a similar procedure one could consider the quotient of the set of quadrilaterals $x = [x_1, x_2, x_3, x_4] \in \mathbb{R}^{2\times 4}$ with respect to the various groups. In this case, the quotient with respect to the affine group is an $8-6=2$-dimensional space. However, the quotient of quadrilaterals with respect to the *projective group* is 0-dimensional, as vertices of different quadrilaterals can be mapped onto one another by a projective transformation [15].

One could continue the construction for an arbitrary number of points on the plane, and quotient out the various group: translation, Euclidean, similarity, affine, projective, …where does it stop? Unfortunately, the next stop is infinite-dimensional, the group of diffeomorphisms [23], and a diffeomorphism can send any finite collection of points to arbitrary locations on the plane. Therefore, just like affine transformations for triangles, and projective transformations for quadrilaterals, the quotient with respect to diffeomorphisms collapses any arbitrary (finite) collections of N points into one element of $\mathbb{R}^N$. However, as we will see, there are infinite-dimensional spaces that are not destroyed by the diffeomorphic group [23].

2.0.4 Comparing Canonical Elements: Procrustes and Geodesic Distances

As we have seen, the canonization procedure enables us to reduce the dimension of the space by the dimension of the group. For instance, triangles live in a 6-dimensional space, but once we quotient (or "mod-out") similarities, they can be represented by $\hat{x} \in \mathbb{R}^2$. That is, the canonical representatives can be displayed on a planar plot. Consider, for instance, two collections of random triangles: One is made of isosceles triangles, one is made of scalenes. If visualized as triangles, it is very difficult to separate them. Visualizing them in their native 6-dimensional space is obviously a challenge. However, if we visualize the quotient, their structure emerges clearly (Fig. 4).

Of course, the mod-out operation (canonization) alters the geometry of the space. For instance, triangles belong to the linear space $\mathbb{R}^{2\times 3} \sim \mathbb{R}^6$. In that space, one can sum triangles, multiply them by a scalar, and still obtain triangles. In other words, X is a linear space. However, the quotient X/G is not necessarily a linear space, in the sense that summing or scaling canonical representative may not yield a valid canonical representative. Indeed, the quotient space X/G is in general a *homogeneous space* that is non-linear (curved) even when the native space X and the group G are linear. Therefore, when considering a distance in the base space as we have done above, one should in principle choose a *geodesic distance* (the length of the shortest path between two points that remains in the space) as opposed to a cordal distance that is the distance in the embedding space as we have done above.

In addition, the canonization procedure significantly distorts the original space. Consider in fact a collection of triangles that is represented by a Gaussian distribution in the space $X = \mathbb{R}^6$. Once we canonize each of them with respect to the Similarity group, the resulting distribution in the quotient space X/G is not Gaussian, but rather part of what are known as *Procrustean* distributions [11].

2.0.5 Not All Canonizations Are Created Equal

It is important to notice that the canonization mechanism is not unique. To canonize translation, instead of choosing x_1, we could have chosen x_2, or x_3, or any combination of them, for instance the centroid. Similarly, for rotation we fixed the direction of the segment x_1x_2, but we could have chosen the principal direction (the singular vector of the matrix x corresponding to its largest singular value).

In principle, no canonization mechanism is better than the other, in the sense that they all achieve the goal of quotienting-out (eliminating) the group. However, consider two triangles that are *equivalent* under the similarity group (i.e., they can be transformed into one another by a similarity transformation), but where the order of the three vertices in x is scrambled: $x = [x_1, x_2, x_3]$ and $\tilde{x} = [x_2, x_1, x_3]$. Once we follow the canonization procedure above, we will get two canonical representatives $\hat{x} \neq \hat{\tilde{x}}$ that are different. What happens is that we have eliminated the similarity group, but *not* the permutation group. So, we should consider not one canonical representative, but 6 of them, corresponding to all possible reorderings of the vertices. One can easily see how this procedure becomes unworkable when we have large collection of points, $x \in \mathbb{R}^{2\times N}, N >> 2$.

If, however, we had canonized translation using the centroid, rotation using the principal direction (singular vector corresponding to the largest singular value), and scale using the largest singular value, then we would only have to consider symmetries relative to the principal direction, so that choice of canonization mechanism is more desirable.

2.0.6 Structural Stability of the Canonization Mechanism

Requiring that the canonization mechanism be *unique* is rather stiff. Geometrically, it corresponds to requiring that the homogeneous space X/G admits a global coordinate chart, which is in general not possible, and one has instead to be content with an atlas of local coordinate charts.

However, what is desirable is to make sure that, as we travel smoothly along an orbit $[x]$ via the action of a group g, the canonical representative $\hat{x} = \hat{x}(gx)$ does not all of a sudden "jump" to another chart.

Consider, again, the example of triangles. Suppose that we choose as a canonical representative for translation the point that has the smallest abscissa (the "left-most" point). As we rotate the triangle around, the canonical representative switches, which is undesirable. A more “stable” canonization mechanism is to choose the centroid as canonical representative, as it is invariant to rotations. The notion of “structural stability” is critical to the canonization process [21], and involves the relation between the group that is being canonized and all the other nuisances (which may or may not be groups). The design of a suitable canonization mechanism should take such an issue into account.

2.1 Extension to Infinite-Dimensional Shape Spaces

The general intuition behind the process of eliminating the effects of the group G from the space X is not restricted to finite-dimensional spaces, nor to finite-dimensional groups. We can mod-out finite-dimensional groups from infinite-dimensional spaces, and then infinite-dimensional groups from infinite-dimensional spaces. When we talk about infinite-dimensional spaces we refer to function spaces, that are characterized by a (finite-dimensional) domain X, a finite-dimensional range, and a map from the former to the latter.

As an example, we will consider images as elements of the function space $\mathscr{I}$ that maps the plane onto the positive reals, $I : \mathbb{R}^2 \to \mathbb{R}$.

2.1.1 Transformations of the Range of a Function (Left Action)

In the previous section we have considered affine transformations of $\mathbb{R}^2$. We now consider affine transformation of $\mathbb{R}$, and apply them to the range of the function $I : \mathbb{R}^2 \to \mathbb{R};\ x \mapsto I(x)$. For simplicity we assume that I is smooth, defined on a compact subset $D \subset \mathbb{R}^2$, and has a bounded range. An affine transformation is defined by two scalars α, β, with $\alpha \neq 0$, and transforms the range of the function $I(\cdot)$ via $g \circ I \doteq \alpha I + \beta$. Therefore, the orbits we consider are of the form $[I] = \{\alpha I + \beta, \alpha > 0, \beta \in \mathbb{R}\}$, and the function $g \circ I$ is defined by $g \circ I(x) = \alpha I(x) + \beta$.

As in the finite-dimensional case, there are several possible canonization mechanisms. The simplest consists in choosing the canonical representative of β to be the smallest value taken by I, $\beta = \min_{x \in D \subset \mathbb{R}^2} I(x)$ and the canonical representative of α to be the largest value $\alpha = \max_{x \in D \subset \mathbb{R}^2} I(x)$. However, one could also choose the mean for β and the standard deviation for α. This is no different than if I was an element of a finite-dimensional vector space. In either case, the canonical group element $\{\hat{\alpha}, \hat{\beta}\} = \hat{g}(I)$ is determined from the function I, and is then “un-done” via $\hat{g}^{-1} \circ I = \frac{I - \hat{\beta}}{\hat{\alpha}}$. Again, we have that the canonical element is $\hat{I} = \hat{g}^{-1}(I) \circ I$.

More interesting is the case when the group acting on the range is infinite-dimensional. Consider for instance all *contrast functions*, that is functions $k : \mathbb{R} \to \mathbb{R}$ that are continuous and monotonic. These form a group, and indeed an infinite-dimensional one. The equivalence class we consider is now

$$[I] = \{k \circ I, k \in \mathscr{H}\},$$

and $g \circ I(x) = k(I(x))$, where $\mathscr{H}$ is the set of contrast transformations.

A canonization procedure for contrast transformations is equivalent to a *"dynamic time warping"* [12] of the range of the function[5] that is chosen in a way that depends on the function itself. The affine range transformation was a very special case. It has been shown in [1] that the quotient of real-valued functions with respect to contrast transformations is the equivalence class of iso-contours of the function. So, by substituting the value of each pixel with the curvature of the iso-intensity curves, one has effectively canonized contrast transformations. Equivalently, because the iso-contours are normal to the gradient direction, one can canonize contrast transformations by considering, instead of the value of I at x, the direction of the gradient of the image at x. This explains the popularity of the use of gradient direction histograms in image analysis [14].

Note that in all these cases, the canonical element of the group $g = \{\alpha, \beta\}$ or $g = k(\cdot)$, is chosen in a way that depends only on the function $I(\cdot)$ in question, so we can write the canonical element as $\hat{g} = \hat{g}(I)$ and, as usual, we have

$$\hat{I} = \hat{g}^{-1}(I) \circ I.$$

2.1.2 Transformations of the Domain of a Function (Right Action)

We have already seen how to canonize finite-dimensional groups of the plane for the case of triangles (or three points on the image domain). The orbits we consider are of the form $I \circ g(x) \doteq I(gx)$. So, if we want to canonize domain transformations of the function I, that is if we want to represent the quotient space of the equivalence class

$$[I] = \{I \circ g, g \in G\},$$

we need to find canonical elements $\hat{g}$ that depend on the function itself. In other words, we look for canonical elements $\hat{g} = \hat{g}(I)$. As a simple example, we could canonize translation by choosing the highest value of I (location of the brightest pixel), and rotation using the principal curvatures of the function I at the maximum, or the direction to the second-brightest pixel. However, instead of the maximum of the function I we could choose the maximum of any operator (functional) acting on I, for instance the Hessian of Gaussian $\nabla^2 G * I(x) \doteq \int_D \nabla^2 G(x-y) I(y) dy$, where D is a neighborhood around the extremum. Similarly, instead of choosing the principal curvature of the function I, we could choose the principal directions of the second-moment matrix $\int_D \nabla I^T \nabla I(x) dx$. In either case, once we have a canonical representative for translation and rotation, we have $\hat{g}$, and everything proceeds just like in the finite-dimensional case.

More interesting is the case when the group g is infinite-dimensional, for instance the set $\mathscr{W}$ of planar diffeomorphisms $w : \mathbb{R}^2 \to \mathbb{R}^2$. In this case we consider the orbit $[I] = \{I \circ g, g \in \mathscr{W}\}$ where the function $I \circ g(x) \doteq I(w(x))$. It has been shown in [27] that this is possible. Below we discuss the case of jointly eliminating domain and range diffeomorphisms.

[5] The name is misleading, because that there is nothing dynamic about dynamic time warping, and there is no time involved.

2.1.3 Joint Domain and Range Transformations (Left and Right Actions)

So far we have considered functions with groups acting either on the range $g \circ I(x) \doteq k(I(x))$ or on the domain $I \circ g(x) \doteq I(w(x))$. However, there is nothing that prevents us from considering groups acting simultaneously on the domain and range, so long as their joint action can be considered as the action of the product group. In this case, we consider orbits of the kind

$$[I] = \{g_1 \circ I \circ g_2,\ g_1 \in G_1,\ g_2 \in G_2\}.$$

The canonization mechanism is the same, leading to $\hat{g}_1(I), \hat{g}_2(I)$, from which we can obtain the canonical element

$$\hat{I} \doteq \hat{g}_1^{-1}(I) \circ I \circ \hat{g}_2^{-1}(I).$$

In [23] it is shown that the quotient of images – interpreted as integrable functions and approximated with Morse functions – modulo contrast transformations and diffeomorphisms (the group closure of viewpoint-induced domain deformations) is the Attributed Reeb Tree (ART) of the image. In Section 3 we elaborate further on the geometry of the shape space of the Lambert-Ambient model.

The functional $\hat{g}(I)$ that chooses the canonical element of the group is also called a *co-variant detector*, in the sense that it varies with the group. Once a co-variant detector has been determined, the canonical representative automatically determines a statistic, $\hat{I} \doteq I \circ \hat{g}^{-1}(I)$, that is a function of the image known as *invariant descriptor*. It is invariant in the sense that, as the group g changes, the canonical element changes with it, but the image referred to the canonical reference frame does not.

As far as eliminating the effects of the nuisance group G, any co-variant detector function is equivalent. Where they differ is in how they address all the other (non group) nuisances such as noise an quantization. Ideally, one would want the canonization procedure to *commute* with non-group nuisances, so that the results of canonization before and after noise or quantization are "the same." We will articulate later what "same" means in this context.

3 The Geometry of the Lambert-Ambient Model

The previous section introduced the notion of a shape space as the quotient of a space X modulo the action of a group G, or X/G. We have seen that both the space and the group can be rather complex, including infinite-dimensional. For instance, X can be the set of radiance functions defined on surfaces in space, and G the (group closure of the) set of domain deformations induced by a viewpoint change. In this section we instantiate the specific case of a Lambertian scene viewed under ambient illumination, to highlight the geometry of the quotient of the set of resulting images under changes of viewpoint and contrast.

We consider an object of interest that is *static* and *Lambertian*, so it can be described by its geometry, a surface $S : D \subset \mathbb{R}^2 \to \mathbb{R}^3;\ x_0 \mapsto S(x_0)$ and its photometry, the albedo $\rho : S \to \mathbb{R}^+; p \mapsto \rho(p)$. We assume an ambient illumination model that modulates the albedo with a simple contrast transformation $k : \mathbb{R} \to \mathbb{R}; I \mapsto k(I)$. The scene is viewed from a vantage point determined by $(R,T) \in SE(3)$, so that the point p projects onto the pixel with coordinate $x = \pi(Rp + T)$, with $R \in SO(3)$ an orthogonal matrix with unit determinant, and $T \in \mathbb{R}^3$. In the absence of occlusions, regardless of the shape of S, the map from x_0 to x is a homeomorphism, $x = \pi(RS(x_0) + T) \doteq w^{-1}(x_0)$; the choice of name w^{-1} is to highlight the fact that it is invertible. If we assume (without loss of generality given the visibility assumption) that p is the radial graph of a function $Z : \mathbb{R}^2 \to \mathbb{R}$ (the range map), so that is $p = \bar{x}Z(x)$, where $\bar{x} = [x,\ y,\ 1]^T$ are the homogeneous coordinates of $x \in \mathbb{R}^2$, we have that

$$w(x) = \frac{[\mathbf{e}_1 \mathbf{e}_2]^T R^T (\bar{x}Z(x) - T)}{\mathbf{e}_3^T R^T (\bar{x}Z(x) - T)} \quad \text{and} \quad w^{-1}(x) = \frac{[\mathbf{e}_1 \mathbf{e}_2]^T (RS(x) + T)}{\mathbf{e}_3^T (RS(x) + T)}. \tag{1}$$

where $\mathbf{e}_i$ are the i-th coordinate vectors. Putting all the elements together we have a model that is valid under assumptions of Lambertian reflection, ambient illumination, and co-visibility:

$$I(x) = k \circ \rho \circ S \circ w(x) + n(x), \qquad x \in D. \tag{2}$$

In relating this model to the previous discussion on canonization, a few considerations are in order:

- There is an additive term n, that collects the results of all unmodeled uncertainty. Therefore, one has to require not only that left- and right- canonization commute, but also that the canonization process be *structurally stable* with respect to such uncertainty (often referred to as "noise"). If we are canonizing the group g (either k or w), we cannot expect that $\hat{g}(I) = \hat{g}(I - n)$, but we want $\hat{g}$ to depend continuously on n, and not to exhibit jumps, singularities, bifurcations and other topological accidents. This goes to the notion of *structural stability* and *proper sampling* addressed in [13].
- If we neglect the "noise" term n, we can think of the image as a point on the orbit of the "scene" $\rho \circ S$. Because the two are entangled, in the absence of additional information we cannot determine either ρ or S, but only their composition. This means that if we canonize contrast k and domain diffeomorphisms w, we obtain an invariant descriptor that lumps into the same equivalence class all objects that are homeomorphically equivalent to one another [27]. The fact that $w(x)$ depends on the scene S (through the function Z) shows that when we canonize viewpoint g we lose the ability to discriminate objects by their shape (although see later on occlusions and occluding boundaries). Thus, with an abuse of notation, we indicate with ρ the composition $\rho \circ S$.

We now show that the planar isometric group $SE(2)$ can be isolated from the diffeomorphism w, in the sense that

$$w(x) = \tilde{w} \circ g(x) = \tilde{w}(gx) \tag{3}$$

for a planar rigid motion $g \in SE(2)$ and a residual planar diffeomorphism $\tilde{w}$, in such a way that the residual diffeomorphism $\tilde{w}$ can be made (locally) independent of planar translations and rotations. More specifically, if the spatial rigid motion $(R,T) \in SE(3)$ has a rotational component R that we represent using angles θ for *cyclo-rotation* (rotation about the optical axis), and ω_1, ω_2 for rotation about the two image coordinate axes, and translational component $T = [T_1, T_2, T_3]^T$, then the residual diffeomorphism $\tilde{w}(x) = \tilde{w}(x)$ can be made *locally* independent of T_1, T_2 and θ. To see that, note that $R_i(\theta) = \exp(\widehat{\mathbf{e}}_3\theta)$ is the *in-plane* rotation, and $R_o(\omega_1, \omega_2) = \exp(\widehat{\mathbf{e}}_2\omega_2)\exp(\widehat{\mathbf{e}}_1\omega_1)$ is the *out-of-plane* rotation, so that $R = R_iR_o$. In particular,

$$R_i = \begin{bmatrix} R_1(\theta) & 0 \\ 0 & 1 \end{bmatrix} \quad \text{where} \quad R_1(\theta) \doteq \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

We write R_o in blocks as

$$R_o = \begin{bmatrix} R_2 & \mathbf{r}_3 \\ \mathbf{r}_4^T & r_5 \end{bmatrix}$$

where $R_2 \in \mathbb{R}^{2\times 2}$ and $r_5 \in \mathbb{R}$. We can then state the claim:

Theorem 0.1. *The diffeomorphism $w : \mathbb{R}^2 \to \mathbb{R}^2$ corresponding to a vantage point $(R,T) \in SE(3)$ can be decomposed according to (3) into a planar isometry $g \in SE(2)$ and a residual diffeomorphism $\tilde{w} : \mathbb{R}^2 \to \mathbb{R}^2$ that can be made invariant to $R_1(\theta)$ and arbitrarily insensitive to T_1, T_2, in the sense that $\forall\ \varepsilon\ \exists\ \delta$ such that $\|x\| \leq \delta \Rightarrow \|\frac{\partial \tilde{w}}{\partial T_i}\| \leq \varepsilon$ for $i = 1, 2$.*

This means that, by canonizing a planar isometry, one can quotient out spatial translation parallel to the image plane, and rotation about the optical axis, at least in a neighborhood of the origin.

Proof: *We write the diffeomorphism explicitly as*

$$w(x) = \frac{[\mathbf{e}_1\mathbf{e}_2]^T\,(R_iR_o\bar{x}Z(x) + T)}{\mathbf{r}_4^T xZ(x) + r_5 + T_3} = \frac{R_2R_1(\theta)xZ(x) + \mathbf{r}_3Z(x) + [T_1, T_2]^T}{\mathbf{r}_4^T xZ(x) + r_5 + T_3} \tag{4}$$

and define the disparity $d(x) = 1/Z(x)$*, so the above expression becomes*

$$w(x) = \frac{R_2R_1(\theta)x + \mathbf{r}_3 + [T_1, T_2]^T d(x)}{\mathbf{r}_4^T x + (r_5 + T_3)d(x)} \tag{5}$$

We can now apply a planar isometric transformation $\hat{g} \in SE(2)$ defined in such a way that $\tilde{w}(x) = w \circ \hat{g}^{-1}(x)$ satisfies $\tilde{w}(0) = 0$, and $\tilde{w}(x)$ does not depend on $R_1(\theta)$. To this end, if $\hat{g} = (\hat{R}, \hat{T})$, we note that

$$\tilde{w}(x) \doteq w \circ \hat{g}^{-1}(x) = \frac{R_2R_1(\theta)\hat{R}^T(x - \hat{T}) + \mathbf{r}_3 + [T_1, T_2]^T\tilde{d}(x)}{\mathbf{r}_4^T x + (r_5 + T_3)\tilde{d}(x)} \tag{6}$$

and $\tilde{d}(x) = d \circ \hat{g}^{-1}(x) = d(\hat{R}^T(x - \hat{T}))$ is an unknown function, just like d was. We now see that imposing[6]

[6] It may seem confusing that the definition of $\hat{T}$ is "recursive" in the sense that $\hat{T} = R_2^{-1}[T_1, T_2]^T\tilde{d}(x) = R_2^{-1}[T_1, T_2]^T d(\hat{R}^T(x - \hat{T}))$. This, however, is not a problem because $\hat{T}$ is chosen *not* by solving this equation, but by an independent mechanism of imposing that $\tilde{w}(0) = 0$, that is a "translation co-variant detector."

$$\hat{R} \doteq R_1(\theta) \quad \text{and} \quad \hat{T} = R_2^{-1}[T_1, T_2]^T \tilde{d}(0) \tag{7}$$

we have that the residual diffeomorphism is given by

$$\tilde{w}(x) = \frac{R_2 x + [T_1, T_2]^T (\tilde{d}(x) - d_0)}{\mathbf{r}_4^T x + (r_5 + T_3)\tilde{d}(x)}. \tag{8}$$

Note that $\tilde{w}$ does not depend on $R_1(\theta)$; because of the assumption on visibility and piecewise smoothness of the scene, $\tilde{d}$ is a continuous function, and therefore the function $\tilde{d}(x) - d_0$ in a neighborhood of the origin $x = 0$ can be made arbitrarily small; such a function is precisely the derivative of $\tilde{w}$ with respect to T_1, T_2.

In the limit where the neighborhood is infinitesimal, or when the scene is fronto-parallel, so that $\tilde{d}(x) = \text{const.}$, we have that

$$\tilde{w}(x) \simeq \frac{R_2 x}{\mathbf{r}_4^T x + (r_5 + T_3)\tilde{d}(x)}, \quad x \in \mathscr{B}_\varepsilon(0) \tag{9}$$

where $\mathscr{B}_\varepsilon$ is a neighborhood ("ball") of radius ε around the point x. A canonization mechanism can be designed to choose $\hat{R}$, for instance so that the ordinate axis of the image is aligned with the projection of the gravity vector onto the image.

The consequence of this theorem, and the previous observation that S and ρ cannot be disentangled, are that we can represent the Lambert-Ambient model (2) as

$$I(x) = k \circ \rho \circ \tilde{w} \circ g(x) + n(x). \tag{10}$$

In the absence of noise n, the canonization process would enable us to mod-out $k, \tilde{w}$ and g, and would yield a canonical element $\hat{I}$ that belongs to the equivalence class $[\rho]$ under viewpoint and contrast transformations. This is precisely the ART introduced in [23].

In the presence of noise, the group g acts linearly on the image, in the sense that $(I_1 + I_2) \circ g = I_1 \circ g + I_2 \circ g$. So, the canonization process effectively eliminates the dependency on g:

$$I \circ \hat{g}^{-1}(x) = k \circ \rho \circ \tilde{w}(x) + \tilde{n}(x) \tag{11}$$

where $\tilde{n}(x) \doteq n \circ \hat{g}^{-1}(x)$. Because $\hat{g}$ is an isometry, $\tilde{n}$ will be a transformed realization of a random process that has the same statistical properties (*e.g.* mean and covariance) of n. Although $\tilde{w}$ also acts linearly on the image, $\tilde{n} \circ \tilde{w}^{-1}$ does *not* have the same statistical properties of $\tilde{n}$, because the diffeomorphism $\tilde{w}$ alters the distribution of $\tilde{n}$. Therefore, *the canonization process for $\tilde{w}$ does not commute with the additive noise* and cannot be performed in an exact fashion.

Similarly, the general contrast transformation k does *not* act linearly on the image, in the sense that $k^{-1} \circ (I_1 + I_2) \neq k^{-1} \circ I_1 + k^{-1} \circ I_2$. Similarly to what we have done for w, we can isolate the affine component of k, that is the contrast transformation $I \mapsto \alpha I + \beta$, and canonize that. For simplicity, we just assume that k is not a general contrast transformation, but instead an affine contrast transformation. By canonizing it we have

$$\hat{k}^{-1} \circ I \circ \hat{g}^{-1}(x) = \rho \circ \tilde{w}(x) + n'(x) \tag{12}$$

where now $n'(x)$ has a statistical description that can be easily derived as a function of the statistical description of n and the values of α, β in the contrast canonization (if μ and σ are the mean and standard deviation of n', then $(\mu - \beta)/\alpha$ and σ/α are the mean and standard deviation of n'). If we summarize the canonization process as a functional ϕ acting on the image, and forgo all superscripts, we have

$$\phi(I(x)) = \rho \circ w(x) + n(x). \tag{13}$$

When the noise is "small" one can think of $\phi(I)$ as a small perturbation of a point on the base space of the orbit space of equivalence classes $[\rho \circ w]$ under the action of planar isometries and affine contrast transformations.

So, even if domain deformations under a general viewpoint change in front of a non-planar scene induce a (subset of a) group transformation, only a small subgroup can be canonized without a loss. We will therefore have to deal with the residual transformation $\tilde{w}$, which we will do later, after we have discussed other nuisances that not only cannot be canonized without a loss, but cannot be canonized at all since they are not groups.

3.1 Occlusions

In the presence of occlusions, including self-occlusions, the map w is not, in general, a diffeomorphism. Indeed, it is not even a function, in the sense that for several locations in the image, $x \in \Omega$, it is not possible to find *any* transformation $w(x)$ that maps the radiance ρ onto the image I. In other words, if D is the image-domain, we only have that

$$I(x) = k \circ \rho \circ w(x), \quad x \in D \backslash \Omega. \tag{14}$$

The image in the *occluded region* Ω can take arbitrary values that are unrelated to the radiance ρ in the vicinity of the point $S(w(x)) \in \mathbb{R}^3$; we call these values $\beta(x)$. Therefore, neglecting the role of the additive noise $n(x)$ for now, we have

$$I(x) = k \circ \rho \circ w(x)(1 - \chi_\Omega(x)) + \beta(x)\chi_\Omega(x), \quad x \in D \tag{15}$$

where $\chi_A(x)$ is the characteristic function of the set A; that is, $\chi_A(x) = 1$ if $x \in A$ and zero otherwise. The canonization mechanism acts on the image I, and has no knowledge of the occluded region Ω. Therefore, $\phi(I)$ *may* eliminate the effects of the nuisances k and w, if it only depends on the values of the image in the visible region, or it may *not* – if it depends on the values of the image in an occluded region. If $\phi(I)$ is computed in a region $\hat{R}^T \mathscr{B}_\sigma(x - \hat{T})$, then the canonization mechanism is successful if

$$\hat{R}^T \mathscr{B}_\sigma(x - \hat{T}) \subset D \backslash \Omega. \tag{16}$$

And fails otherwise. Whether the canonization process succeeds or fails can only be determined by comparing the statistics of the canonized image $\phi(I)$ with the statistics of the radiance, ρ, which is of course unknown. However, under the

Lambertian assumption, this can be achieved by comparing the canonical representation of *different images*.

3.1.1 Determining Co-Visibility

If range maps were available, one could test for co-visibility as follows: Let $Z : D \subset \mathbb{R}^2 \to \mathbb{R}^+; x \mapsto Z(x;S)$ be defined as the distance of the point of first intersection of the line through x with the surface x:

$$Z(x;S) = \min\{Z > 0 \mid \bar{x}Z \in S\}. \tag{17}$$

When the surface is moved, the range map changes, not necessarily in a smooth way because of self-occlusions:

$$Z(x;RS+T) = \min\{Z > 0 \mid R\bar{x}Z + T \in S\}. \tag{18}$$

A point with coordinates x_0 on an image is *co-visible* with a point with coordinates x in another image taken by a camera that has moved by $(R,T) \in SE(3)$ if

$$\bar{x}Z(x;RS+T) = R\bar{x}_0 Z(x_0;S) + T. \tag{19}$$

An alternative expression can be written using the third component of the equation above, that is

$$Z(x;RS+T) = \mathbf{e}_3^T(RS(x_0) + T). \tag{20}$$

Therefore, the visible domain $D \backslash \Omega$ is given by the set of points x that are co-visible with any point $x_0 \in D$. Vice-versa, the occluded domain is given by points that are not visible, i.e.

$$\Omega = \{x \in D \mid Z(x;RS+T) \neq \mathbf{e}_3^T(R\bar{x}_0 Z(x_0;S) + T), \quad x_0 \in D\}. \tag{21}$$

The region Ω can be inferred from multiple images of the same scene, along with the diffeomorphism w, under the assumption of Lambertian reflection and ambient illumination, by solving a variational optimization problem [2].

To summarize, from the shape space of the Lambert-Ambient model we have:

- In the absence of noise, $n = 0$ and occlusions $\Omega = \emptyset$, the shape space of the Lambert-Ambient model is the set of Attributed Reeb Trees [23].
- In the presence of additive noise $n \neq 0$, but no occlusions, $\Omega = \emptyset$, the shape space is the collection of radiance functions composed with domain diffeomorphisms with a fixed point (*e.g.* the origin) and a fixed direction (*e.g.* gravity).
- In the presence of noise and occlusions, the shape spaces is broken into local patches that are the domain of attraction of covariant detectors. The size of the region depends on scale and visibility and cannot be determined from one datum only. Co-visibility must be tested as part of the correspondence process, by testing for geometric and topological consistency, as well as photometric consistency [2].

This construction justifies the use of local descriptors, that would otherwise be detrimental in light of the Data Processing Inequality [6].

4 Co-variant Detectors, Invariant Descriptors

The previous section described a procedure to eliminate the effects of a nuisance group by (i) determining a *canonical element* of the group based on one datum $\hat{g}(I)$, and (ii) inverting the canonical element so that the datum, in the (moving) reference frame of the canonical element, $I \circ \hat{g}^{-1}(I)$ does not change with the group. So, the data itself, in the canonical reference frame, is by construction *invariant* to the group.

In this section, we will give specific examples of functions that select a canonical element in (i), that are *co-variant detectors*. They are all equivalent as far as eliminating the effects of the nuisance group (ii). However, they all differ in the way in which they handle *all other* nuisances. Therefore, the data in the canonical frame is often further processed to generate statistics that, in addition to being invariant to the nuisance group, are also *insensitive* to all other nuisances, including those not explicitly modeled. These statistics (functions of the data) are called *invariant descriptors*.

We will first formalize the notion of co-variant detector, then show that most of the commonly used detectors can be understood as special cases of canonization. Then we will show how to handle the residual variability, and frame existing invariant descriptors in this framework, where it becomes evident that there are systematic ways of handling residual nuisances, which we do in Section 5.

We consider the set of digital images $\mathscr{I}$ to be (piece-wise constant) functions $I : \mathbb{R}^2 \to \mathbb{R}^2;\ x \in \mathscr{B}_\varepsilon(x_{ij}) \mapsto I_{ij}$, that can be identified with the set of matrices $\mathbb{R}^{N\times M}$. A differentiable functional $\psi : \mathscr{I} \times G \to \mathbb{R}; (I,g) \mapsto \psi(I,g)$ is said to be *local*, with *effective support* σ if its value at g only depends on a neighborhood of the image of size $\sigma > 0$, up to a residual that is smaller than the mean quantization error. For instance, for a translational frame g, if we call $I_{|\mathscr{B}_\sigma(g)}$ an image that is identical to I in a neighborhood of size σ centered at position $g \equiv T$, and zero otherwise, then $\psi(I_{|\mathscr{B}_\sigma(g)}, g) = \psi(I,g) + \tilde{n}$, with $|\tilde{n}| \leq \frac{1}{NM}\sum_{i,j}|n_{ij}|$. For instance, a functional that evaluates the image at a pixel $g \equiv T = x \in \mathscr{B}_\varepsilon(x_{ij})$, is local with effective support ε. For groups other than translation, we consider the image in the reference frame determined by g, or equivalently consider the "transformed image" $I \circ g^{-1}$, in a neighborhood of the origin, so $\psi(I,g) = \psi(I \circ g^{-1}, e)$, where e is the identity element of the group G.

If we call $\nabla\psi \doteq \frac{\partial\psi}{\partial g}$ the gradient of the functional ψ with respect to (any) parametrization of the group,[7] then under certain (so-called "transversality") conditions on ψ, the equation $\nabla\psi = 0$ locally determines g a function of I, $g = \hat{g}(I)$, via the Implicit Function Theorem. Such conditions are independent of the

[7] The following discussion is restricted to finite-dimensional groups, but it could be extended with some effort to infinite-dimensional ones.

parametrization and consist of the Hessian matrix $H(\psi) \doteq \nabla\nabla\psi$ being non-singular, $\det(H(\psi)) \neq 0$. The function $\hat{g}$ is unique in a neighborhood where the transversality condition is satisfied, and is called a (local) *canonical representative* of the group. If the canonical representative co-varies with the group, in the sense that $\hat{g}(I \circ g) = (\hat{g} \circ g)(I)$, then the functional ψ is called a *co-variant detector*. Each co-variant detector determines a local reference *frame* so that, if the image is transformed by the action of the group, a hypothetical observer attached to the co-variant frame would see no changes. We summarize this in the following definition:

Definition 1 (Co-variant detector). *A differentiable functional* $\psi : \mathscr{I} \times G \to \mathbb{R}$; $(I, g) \mapsto \psi(I, g)$ *is a co-variant detector if*

1. *The equation* $\det(H(\psi(I, g))) = 0$ *locally determines a unique isolated extremum in the frame* $g \in G$*, and*
2. *if* $\nabla\psi(I, \hat{g}) = 0$*, then* $\nabla\psi(I \circ g, \hat{g} \circ g) = 0 \ \forall \ g \in G$*,* i.e.*,* ψ *co-varies with G.*

The first "transversality" condition [8] corresponds to the Jacobian of $\nabla\psi$ with respect to g being non-singular:

$$|J_{\nabla\psi}| \neq 0. \tag{22}$$

In words, a co-variant detector is a functional that determines an isolated group element in such a way that, if we transform the image, the group elements is transformed in the same manner. We have already seen simple examples of co-variant detectors; more realistic examples will follow shortly.

The transversality condition (22) guarantees that $\hat{g}$, the canonical element, is an isolated (Morse) critical point [17] of the derivative of the function ψ via the Implicit Function Theorem [8]. So a co-variant detector is a statistic (a feature) that "extracts" a group element $\hat{g}$. With a co-variant detector we can easily construct an invariant descriptor, or *local invariant feature*, by considering the data itself in the reference frame determined by the detector:

Definition 2 (Canonized descriptor). *For a given co-variant detector* ψ *that fixes a canonical element* $\hat{g}$ *via* $\nabla\psi(I, \hat{g}(I)) = 0$ *we call the statistic*

$$\boxed{\phi(I) \doteq I \circ \hat{g}^{-1}(I) \ \mid \ \nabla\psi(I, \hat{g}(I)) = 0.} \tag{23}$$

an invariant descriptor.

Where they differ is in how they behave relative to all other nuisances. Later we will give more examples of detectors that are designed to "behave well" with respect to other nuisances. In the meantime, however, we state more precisely the fact that, as far as dealing with a group nuisance, all co-variant detectors do the job.

Theorem 0.2 (Canonized descriptors are complete features). *Let* ψ *be a co-variant detector. Then the corresponding canonized descriptor (23) is an invariant sufficient statistic.*

Proof: *To show that the descriptor is invariant we must show that* $\phi(I \circ g) = \phi(I)$*. But* $\phi(I \circ g) = (I \circ g) \circ \hat{g}^{-1}(I \circ g) = I \circ g \circ (\hat{g}g)^{-1} = I \circ g \circ g^{-1}\hat{g}^{-1}(I) = I \circ \hat{g}^{-1}(I)$*. To show that it is complete it suffices to show that it spans the orbit space* $\mathscr{I}/G$*, which is evident from the definition* $\phi(I) = I \circ g^{-1}$*.*

Example 0.1 (SIFT detector and its variants). To construct a simple translation-covariant detector, consider an isotropic bi-variate Gaussian function $\mathscr{N}(x;\mu,\sigma^2) = \frac{1}{2\pi\sigma}\exp(-\frac{\|x-\mu\|^2}{2\sigma^2})$; then for any given scale σ, the Laplacian-of-Gaussian (LoG) $\psi(I,g) \doteq \nabla^2\mathscr{N}(x;g,\sigma^2) * I(x)$ is a linear translation-covariant detector. If the group includes both location and scale, so $\bar{g} = (g,\sigma^2)$, then the same functional can be used as a translation-scale detector. Other examples are the difference-of-Gaussians (DoG) $\psi(I,g) \doteq \frac{\mathscr{N}(x;g,\sigma^2)-\mathscr{N}(x;g,k^2\sigma^2)}{k-1} * I(x)$, with typically $k = 1.6$, and the Hessian-of-Gaussian (HoG) is $\psi(I,g) = \det H(\mathscr{N}(x;g,\sigma^2))$. Among the most popular detectors, SIFT uses the DoG, as an approximation of the Laplacian.

Example 0.2 (Harris' corner and its variants). Harris' corner and its variants (Stephens, Lucas-Kanade, etc.) replace the Hessian with the second-moment matrix:

$$\psi(I,g) \doteq \det\left(\int_{\mathscr{B}_\sigma(g)} \nabla^T I \nabla I(x)dx\right). \tag{24}$$

One can obtain generalizations to groups other than translation in a straightforward manner by replacing $\mathscr{N}(x;g,\sigma^2)$ with $\frac{1}{2\pi\sigma\det J}\exp(-\frac{\|g^{-1}(x)\|^2}{2\sigma^2\det(J_g)^2})$ where J_g is the Jacobian of the group. For instance, for the affine group $g(x) = Ax + T$, we have that $\psi(I,g) = \nabla^2\left(\frac{1}{2\pi\sigma\det A}\exp(-\frac{\|A^{-1}(x-T)\|^2}{2\sigma^2\det(A)^2})\right)$ is an affine-covariant (Laplacian) detector. One can similarly obtain a Hessian detector or a DoG detector. The Euclidean group has $A \in SO(2)$, so that $\det A = 1$, and the similarity group has $\tilde{\sigma}A$, with determinant $\tilde{\sigma}$.

Unlike the Laplacian of Gaussian or Hessian of Gaussian, this is not a linear functional, and therefore it does not commute with additive nuisances such as quantization or noise [21].

Example 0.3 (Harris-Affine). The only difference from the standard Harris' corner is that the region where the second-moment matrix is aggregated is not a spherical neighborhood of location g with radius σ, but instead an ellipsoid represented by a location $T \in D \subset \mathbb{R}^2$ and an invertible 2×2 matrix $A \in \mathbb{GL}(2)$. In this case, $g = (T,A)$ is the affine group, and the second-moment matrix is computed by considering the gradient with respect to all 6 parameters T,A, so the second-moment matrix is 6×6. However, the general form of the functional is identical to (24), and shares the same limitations.

Although these detectors are not local, their effective support can be considered to be a spherical neighborhood of radius a multiple of the standard deviation $\sigma > 0$, so they are commonly treated as local.[8]

The assumption of differentiability in a co-variant detector is not necessary; in [21] it is shown how to construct co-variant detectors that are not differentiable. Indeed, canonization itself is not necessary to design invariant descriptors. We have already mentioned "blurring" as a way to reduce (if not eliminate) the dependency of a statistic on a group, although that does not yield a sufficient statistic in general [18] (however, some hierarchical multi-scale blurring method is lossless in the limit [5]).

Indeed, even the first condition in the definition of a co-variant detector is not necessary in order to define an invariant descriptor: Assume that the image I is such that for any functional ψ, the equation $\nabla\psi(I,g) = 0$ does *not* uniquely determine $\hat{g} = \hat{g}(I)$. That means that $|J_{\nabla\psi}| = 0$ for all ψ, and therefore all statistics are already (locally) invariant to G. More in general, where the structure of the image allows a "stable" and "repeatable" detection[9] of a frame $\hat{g}$, this can be inverted and canonized $\phi(I) = I \circ \hat{g}^{-1}$. Where the image does *not* enable the detection of a frame $\hat{g}$, it means that the image itself is already invariant to G.

We emphasize that detectors' only purpose is to avoid marginalizing the invertible component of the group G. However, at best such detectors can yield no improvement over marginalizing the action of G, that is *using no detector at all.* Therefore, one should always marginalize or max-out the nuisances if this process is viable given resource constraints such as the need to minimize processing at decision time. This is a design choice that has been explored empirically: In visual category recognition, some researchers prefer to use features selected around "keypoints," whereas others prefer to compute "dense descriptors" at each pixel, or at a regular sub-sampling of the pixel grid, and let the classifier sort out which are informative, at decision time.

Remark 1 (Aliasing and Proper Sampling). *Canonization entails the computation of the Jacobian, which is a differential operation on the image. However, images are* discrete*, merely a sampled version of the underlying signal (assuming* that *is piecewise differentiable), that is the radiance of the scene. In any case, the differentiable approximation, or the computation of the Jacobian, entails a choice of* scale*, depending on which any given "structure" may or may not exist: A differential*

[8] Varying the scale parameter σ produces a *scale-space*, whereby the locus of extrema of ψ describes a *graph* in $\mathbb{R}^3$, via $(x,\sigma) \mapsto \hat{x} = \hat{g}(I;\sigma)$. Starting from the finest scale (smallest σ), one will have a large number of extrema; as σ increases, extrema will merge or disappear. Although in two-dimensional scale space extrema can also appear as well as split, such *genetic* effects (births and bifurcations) have been shown to be increasingly rare as scale increases, so the locus of extrema as a function of scale is well approximated by a tree, which is the *co-variant detection tree* [13].

[9] "Stability" will be captured by the notion of Structural Stability, and "repeatability" by the notion of Proper Sampling.

operator such as the Jacobian could be invertible at a certain scale, and not invertible at a different scale at the same location. *Because the "true" scale is unknown (and it could be argued that it does not exist), canonizability alone is not sufficient to determine whether a region can be* meaningfully *canonized. "Meaningful" in this context indicates that a structure detected in an image corresponds to some structure in the* scene*, and is not instead a* sampling *artifact (*"aliasing"*) due to the image formation process, for instance quantization and noise. Therefore, an additional condition must be satisfied for a region to be "meaningfully" canonized. This is the condition of* Proper Sampling *described in [21].*

In [20] it is shown that the only nuisances that can be canonized without a loss are planar isomorphisms an affine contrast changes; all other nuisances have to either be properly marginalized, that requires solving a complex integration or optimization at decision time, or averaged out as described above. What we have in the end, for each image I, is a set of multiple descriptors (or templates), one per each canonical translation and, for each translation, multiple scales, canonized with respect to rotation and contrast, but still dependent on deformations, complex illumination and occlusions:

$$\begin{aligned}\phi(I) = \{&k_{ij}\circ\rho(S_jR_{ij}x+T_iv_{ij}(x))+n_{ij}(x),\\ &i,j=1,\dots N_T,N_S|\mathscr{B}_{\sigma_j}(x+T_i)\cap D=\emptyset\}\end{aligned} \tag{25}$$

where v_{ij} is the residual of the diffeomorphism $w(x)$ after the similarity transformation $\alpha x+T$ has been applied, *i.e.* , $v_{ij}(x)=w(x)-\alpha_jR_{ij}x-T_i$. If we call the frame determined by the detector $\hat{g}_{ij}=\{\alpha_j,T_i,R_{ij},k_{ij}\}$, we have that

$$\phi(I)=\{I\circ\hat{g}_{ij}^{-1}\}_{i,j=1}^{N_T,N_S}. \tag{26}$$

Note that the selection of occluded regions, which is excluded from the descriptor, is not known a-priori and will have to be determined as part of the matching process.

In the case of video data, $\{I_t\}_{t=1}^T$, one obtains a *time series* of descriptors,

$$\phi(\{I_t\}_{t=1}^T)=\{I_t\circ\hat{g}_{ij}^{-1}(t)\}_{i,j,t=1}^{N_T,N_S,T} \tag{27}$$

where the frames $\hat{g}_{ij}(t)$ are provided by the feature detection mechanism that, in the case of video, consists of a tracking procedure. Note that if we want to canonize a nuisance, in the process of making the feature invariant to the nuisance, we may end up making it also invariant to some components of the scene. In other words, by abusing canonization we may end up throwing away the baby (scene) with the bath water (nuisances).

The simplest example is the interaction of *viewpoint and shape*. In the model (1), we see immediately that the viewpoint (R,T) and shape S interact in the motion field $w(x)=\pi(R\pi^{-1}(x)+T)$, where $p=\pi^{-1}(x)\in S$ depends on the shape of the scene. It is shown in [23] that the group closure of domain warpings w spans the entire group of diffeomorphisms, which can therefore be canonized – if we exclude the effects of occlusion and quantization. However, necessarily the canonization process *eliminates the effects of the shape S in the resulting descriptor*, which is the ART. This means that if we want to perform recognition using a (true) viewpoint-invariant, no matter how it is constructed, then we will lump all objects that have the same radiance, modulo a diffeomorphism, into the same class. That means that, for instance, all white objects are indistinguishable using a viewpoint invariant statistic, regardless of how such an invariant is constructed. Of course, as pointed out in [27], this does not mean that we cannot recognize different objects that have the same radiance. It just means that we cannot do it *with a viewpoint invariant,* and instead we have to resort to marginalization or extremization.

The same phenomenon occurs with reflectance (a property of the scene) and illumination (a nuisance), as discussed in the appendix of [21]. Deciding how to manage the scene-nuisance interaction is ultimately a modeling choice, that should be guided by two factors. The first is the priority in terms of speed of execution (biasing towards canonizing nuisances) *vis-a-vis* discriminative power (biasing towards marginalization to avoid multiple scenes collapsing into the same invariant descriptor). The second is a thorough understanding of the interaction of the various factors and the ambiguities in the image formation model. This means that one should understand, given a set of images, what is the set of all possible scenes that, under different sets of nuisances, can have generated those images. This is the set of *indistinguishable scenes*, that therefore cannot be discriminated from their images. This issue is largely still an open problem.

If we are given a sequence of images $\{I_t\}$ of a static scene, or a rigid object, then the only temporal variability is due to viewpoint g_t, which is a nuisance for the purpose of recognition, and therefore should be either marginalized/max-outed or canonized. In other words, there is no "information" in the time series $\{\hat{g}_t\}$ (of course, this is not the case if the purpose os navigation, or another task where viewpoint is informative). Once we have the tracking sequence available, the temporal ordering is irrelevant. This is not the case when we have a deforming object, say a human, where the time series contains information about the particular action or activity, and therefore temporal ordering is relevant. In this manuscript we focus on rigid scenes, where S does not change over time, or rigid objects, which are just a simply connected component of the scene S_i (detachable objects [3]).

The simplest descriptor that aggregates the temporal data is the (class-conditional) mean or median [13]. However, after we canonize the invertible-commutative nuisances, via the detected frames $\hat{g}_t$, we do not need to blur them, and instead we can construct the template below, where averaging is only performed with respect to the nuisances ν, rather than all nuisances. The prior $dP(\nu)$ is generally not known, and neither is the class-conditional density $dQ_c(\xi)$. However, if a sequence of frames[10] $\{\hat{g}_k\}_{k=1}^T$ has been established in multiple images $\{I_k\}_{k=1}^T$, with $I_k = h(g_k, \xi_k, \nu_k)$, then it is easy to compute the best (local) template via[11]

$$\phi(\hat{I}_c) = \int_{\mathscr{I}} \phi(I) dP(I|c) = \sum_{\substack{\nu_k \sim dP(\nu) \\ \xi_k \sim dQ_c(\xi)}} \phi \circ h(\hat{g}_k \xi_k, \nu_k) = \sum_k I \circ \hat{g}_k = \sum_{k,i,j} \phi_{ij}(I_k) \tag{28}$$

where $\phi_{ij}(I_k)$ are the components of the descriptor defined in eq. (25) for the k-th image I_k that come as a byproduct of a *tracking* procedure. Note that we are tracking reference frames $\hat{g}_k$, not just their translational component (points) x_i. *The template above $\hat{I}_c$, therefore, is an averaging of the gradient direction, in a region determined by $\hat{g}_k$, according to the nuisance distribution $dP(\nu)$ and the class-conditional distribution $dQ_c(\xi)$, as represented in the training data.* This *"best-template descriptor"* (BTD) is implemented in [13]. It is related to [7, 4, 24] in that it uses gradient orientations, but instead of performing spatial averaging by coarse binning, it uses the actual (data-driven) measures and average gradient directions weighted by their standard deviation over time. The major difference is that composing the template *requires local correspondence*, or tracking, of regions $\hat{g}_k$, in the training set. Of course, it is assumed that a *sufficiently exciting sample* is provided, lest the sample average on the right-hand side of (28) does not approximate the expectation on the left-hand side. Sufficient excitation is the goal of active exploration [26].

Note that, once the template descriptor is learned, with the entire scale semigroup spanned in $dP(\nu)$,[12] recognition can be performed by computing the descriptors ϕ_{ij} *at a single scale* (that of the native resolution of the pixel). This significantly improves the computational speed of the method, which in turn enables real-time implementation even on a hand-held device [13]. It should also be noted that, once a template is learned from multiple images, recognition can be performed on a *single* test image.

It should be re-emphasized that the best-template descriptor is only the best among templates, and only relative to a chosen family of classifiers (*e.g.* nearest neighbors with respect to the Euclidean norm). For non-planar scenes, the descriptor can be made viewpoint-invariant by averaging, but that comes at the cost of losing the ability to discriminate based on shape. If we want to recognize by shape, we can marginalize viewpoint, but that comes at a (computational) cost as it corresponds to performing (implicit) reconstruction [21].

It should also be emphasized that the template above is a first-order statistic (mean) from the sample distribution of canonized frames. Different statistics, for instance the median, can also be employed [13], as well as multi-modal

[10] We use k as the index, instead of t, to emphasize the fact that the temporal order is not important in this context.

[11] This notation assumes that the descriptor functional acts linearly on the set of images $\mathscr{I}$; although it is possible to compute it when it is non-linear, we make this choice to simplify the notation.

[12] Either because of a sufficiently rich training set, or by extending the data to a Gaussian pyramid in post-processing.

descriptions of the distribution [28] or other dimensionality reduction schemes to reduce the complexity of the samples.

5 Aggregating Residual Variability

In the previous section we have seen that, after canonization, in general we still have residual variability due to nuisance factors that are not canonized. In the absence of occlusions, this is the residual illumination variability due to the non-affine components of contrast transformations, and the residual viewpoint-induced variability due to the non-affine component of domain deformations. Since occlusions can be *guessed*, but not determined in a single image, a decision on co-visibility is deferred to correspondence time, and co-variant detectors, and their corresponding invariant descriptors, are computed *locally*. The question that remains, then, is how to deal with such residual variability.

Let us assume, for the moment, that a scene or object is viewed from a moving camera, and that (short-baseline) correspondence has been established for a local region. This means that, for that region, a certain point has been *tracked* through the image sequence $\{I_t\}_{t=0}^T$, or equivalently translation has been canonized, in the sense that, relative to the moving frame that has its origin at the tracked point, the region does not translate. We can also canonize rotation, scale, and the affine group [25], as well as local contrast transformations [10], if so desired. If we now imagine taking the regions around the canonized frame at each instant of time (a "stack" of image regions, or patches, around the local moving frame), depending on the scene's shape, reflectance and illumination, these regions can exhibit more or less variability. If these regions back-project onto a planar Lambertian surface in constant ambient illumination, there will be minimal if any variability. If the regions back-project onto a highly curved specular surface, there will be significant variability among the local regions (Fig. 5).

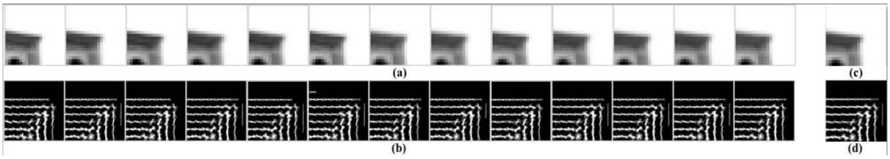


Fig. 5 Tracking provides a collection of local frames and their temporal correspondence. Each frame in a track can be interpreted as a sample from the class-conditional distribution that is, by construction, invariant to the nuisance group. Local descriptors can then be constructed by computing statistics from such a distribution, for instance first-order statistics such as the mean or median [13]. Panel (a) shows image patches around a tracked feature point; (b) shows a contrast invariant (gradient orientation) in the corresponding image patches; (c) shows the mean patch; (d) shows the aggregated orientation histogram. In this case, the portion of the scene being tracked is nearly fronto-parallel and planar, so the different patches exhibit modest variability along the track.

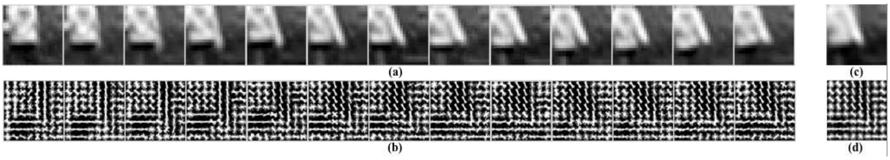


Fig. 6 The same panels as in Figure 5 are shown for a portion of the scene that is not planar and fronto-parallel, thus showing patches that exhibit significant variability.

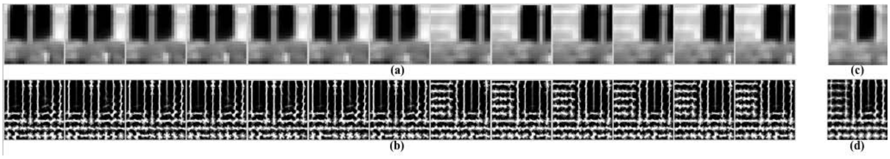


Fig. 7 When the tracking mechanism fails, because the assumptions underlying it (co-visibility, constant illumination, Lambertian reflection) are violated, the resulting patches can exhibit significant variability, including discontinuous variability. In the specific case of this example, there is jump between frame 7 and 8.

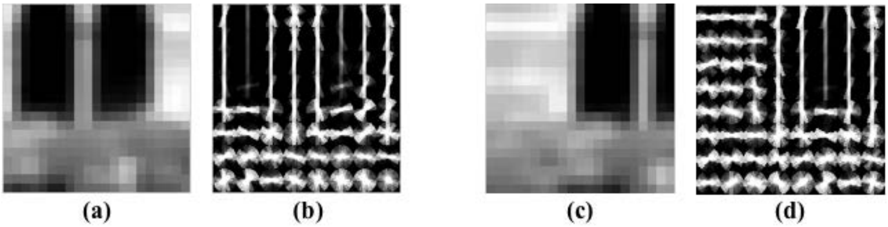


Fig. 8 For the case in Fig. 7 a simple (first-order) statistic such as the mean or the median is not representative of the class-conditional distribution. However, aggregating means for each mode of th distribution yields multi-modal descriptors such as the one corresponding to the first 7 frames in Fig. 7(a); (c) shows the mean patch for the first feature point corresponding to the last 6 frames in Fig. 7(a).

Now, if we have a video with short-baseline correspondence established ("training set"), we could consider these regions $\psi(I_t)$ as *samples* from the *object-specific* distribution $p(\psi|c)$ where c is the object or scene or "class" label. Knowing this distribution would allow us to establish correspondence by computing the likelihood of a region in a new image $\tilde{I}$ ("test set") via $p(\psi(\tilde{I})|c)$. Thus, the object-specific residual variability has been learned, and correspondence is a statistical classification problem. One can then aggregate the class-conditional distribution $p(\psi|c)$ into a classifier. One of the simplest classifiers is the distance to the mean of the distribution: We would first compute the mean $\hat{\psi} = \int \psi dP(\psi|c)$ from the training set, assuming that ψ lives in some vector space, and then compute some distance from the test set to the mean $d(\psi(\tilde{I}), \hat{\psi})$.

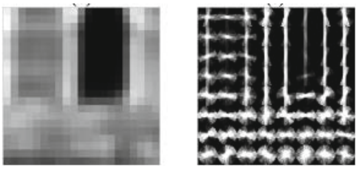


Fig. 9 Construction of the mean and median template for the example in the previous figure. If the distribution is not partitioned into its multiple modes, the mean is a poor representative, and the mean template looks “blurry”. If the distribution is broken into its modes, each template is sharper and more representative of the distribution around the mode (Fig. 8).

Instead of the mean, one could compute the median, or mode, or other statistic of the class-conditional distribution. This process is illustrated in Fig. 9.

What if we do not have a real training set (a video with short-baseline correspondence established), but instead *only have one image*? We could select a descriptor computed *on that image* as the mean $\hat{\psi} \doteq \psi(I)$. However, this may not be representative of the distribution. Instead, one can “guess” or hypothesize a distribution $dP(\psi)$, and average with respect to that distribution, instead of the true (object-specific) one. Of course, in this process necessarily the distribution is not object-specific, so we would be blurring in the same way all descriptors, regardless of whether the data exhibits small or large variability (Fig. 5). This is essentially what most existing *single-view descriptors* do, although the distribution is not explicitly described, but is instead implicit in the choice of quantization, binning, or discretization employed by the algorithm.

In the presence of multiple views, one can do better by averaging relative to the object-specific distribution. In [13], the mean, median, and mode of the marginals were used to develop a template descriptors, whereas in [16], kernel PCA was used to aggregate the class conditional (with PCA being a special case). However, other choices of dimensionality reduction are possible, and can be exercised as part of the design process. Fig. 5 illustrates the process of aggregating statistics from a training set.

Example 0.4 (SIFT and HOG revisited). If instead of a sequence $\{I_k\}$ one had only one image available, one could generate a pseudo-training set by duplicating and translating the original image in small integer intervals. The procedure of building a temporal histogram described above then would be equivalent to computing a *spatial histogram* of gradient orientations. Depending on how this histogram is binned, and how the gradient direction is weighted, this procedure is essentially what SIFT [14] and HOG [7] do. So, one can think of SIFT and HOG as a special case of template descriptor where the nuisance distribution $dP(\nu)$ is *not* the real one, but a simulated one, for instance a uniform scale-dependent quantized distribution in the space of planar translations.

We call the distributional aggregation, rather than the averaging, of $\{\phi_{ij}(I_k)\}$ in (28) the Time HOG or Time SIFT, depending on how the samples are aggregated and binned into a histogram. Although a step up from template descriptors, Time SIFT and Time HOG still discard the temporal ordering in favor of a static descriptor. In cases where the temporal ordering is important, as in the recognition of temporal events, one should instead retain the time series $\{\phi(I_t)\}$ and compare them using dynamic time warping [19], which corresponds to marginalizing, or max-outing, time. This process is considerably more onerous, computationally, at decision time, and well beyond the scope of this expository paper.

6 Conclusion

Local descriptors are ubiquitous in visual recognition and categorization of both objects and scenes. They can either be paired with a *detector* and associated to a sparse collection of "informative" locations, or attached to every pixel in the image. Their design has, so far, been largely driven by intuition and some biological inspiration, but never framed analytically and designed according to some optimality principle.

In this manuscript, we have shown that local descriptors arise from the need to establish correspondence under changes of viewpoint that induce (self) occlusions. While the decision as to whether a region of the scene is *co-visible* in multiple images, or occluded, can only be performed at decision time, marginalizing occluded regions is complex and time consuming. It can be simplified by *analyzing* the image (breaking it down into pieces) and describing it locally, thus leaving the decision as to whether a region is co-visible or not to decision time, where it becomes a combinatorial matching process. Such analysis process would seem to violate the Data Processing Inequality, but it can be justified if the resulting descriptors are sufficient statistics [21]. This, unfortunately, is not the case unless the data formation process can be actively controlled, or a "sufficiently exciting" training input is provided. This is clearly not the case when descriptors are computed from a single image. Nevertheless, this is the most common practice in the literature. Since the true (class-specific) distribution is not available, blurring is performed relative to a *generic* distribution, often implicit in the choice of algorithm.

When multiple views are available and a tracking procedure provides correspondence of local frames, such a blurring can be computed relative to the class-specific distribution. Thus, the use of multiple views *during training* (construction of the descriptors) is beneficial; if these views are temporally adjacent, correspondence of local frames can be easily obtained through tracking.

Acknowledgements. This work has been conducted in support of the Reading Group at ICVSS 2011-2012. We are grateful to Sebastiano Battiato, Roberto Cipolla and Giovanni Maria Farinella for establishing and supporting the reading group. The research described in this manuscript was supported by AFOSR, ONR and the DARPA MSEE project. This research also exploited collaborative facilities renovated with funds from the National Science Foundation under Grant No. 0963183, an award funded under the American Recovery and Reinvestment Act of 2009 (ARRA).

References

1. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations of image processing. Arch. Rational Mechanics 123 (1993)
2. Ayvaci, A., Raptis, M., Soatto, S.: Sparse occlusion detection with optical flow. Intl. J. of Comp. Vision (2012)
3. Ayvaci, A., Soatto, S.: Detachable object detection. IEEE Trans. on Patt. Anal. and Mach. Intell. (2011)
4. Berg, A., Malik, J.: Geometric blur for template matching. In: Proc. CVPR (2001)
5. Bruna, J., Mallat, S.: Classification with scattering operators. In: Proc. IEEE Conf. on Comp. Vision and Pattern Recogn. (2011)
6. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley (1991)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. IEEE Conf. on Comp. Vision and Pattern Recogn. (2005)
8. Guillemin, V., Pollack, A.: Differential Topology. Prentice-Hall (1974)
9. Huang, J., Mumford, D.: Statistics of natural images and models. In: Proc. CVPR, pp. 541–547 (1999)
10. Jin, H., Favaro, P., Soatto, S.: Real-time feature tracking and outlier rejection with changes in illumination. In: Proc. of the Intl. Conf. on Computer Vision, pp. 684–689 (2001)
11. Kendall, D.G.: Shape manifolds, procrustean metrics and complex projective spaces. Bull. London Math. Soc. 16 (1984)
12. Keogh, E.J., Pazzani, M.J.: Dynamic time warping with higher order features. In: Proceedings of the 2001 SIAM Intl. Conf. on Data Mining (2001)
13. Lee, T., Soatto, S.: Video-based descriptors for object recognition. Image and Vision Computing (2011)
14. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV (1999)
15. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.: An invitation to 3D vision, from images to geometric models. Springer (2003)
16. Meltzer, J., Yang, M.-H., Gupta, R., Soatto, S.: Multiple view feature descriptors from image sequences via kernel principal component analysis. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3021, pp. 215–227. Springer, Heidelberg (2004)
17. Milnor, J.: Morse Theory. Annals of Mathematics Studies no. 51. Princeton University Press (1969)
18. Poggio, T.: How the ventral stream should work. Technical report, Nature Precedings (2011)
19. Soatto, S.: On the distance between non-stationary time series. In: Chiuso, A., Ferrante, A., Pinzoni, S. (eds.) Modeling, Estimation and Control. LNCIS, vol. 364, pp. 285–299. Springer, Heidelberg (2007)
20. Soatto, S.: Actionable information in vision. In: Proc. of the Intl. Conf. on Comp. Vision (October 2009)
21. Soatto, S.: Steps Toward a Theory of Visual Information. Technical Report UCLA-CSD100028 (September 13, 2010), `http://arxiv.org/abs/1110.2053`
22. Soatto, S., Yezzi, A.J., Jin, H.: Tales of shape and radiance in multiview stereo. In: Intl. Conf. on Comp. Vision, pp. 974–981 (October 2003)
23. Sundaramoorthi, G., Petersen, P., Varadarajan, V.S., Soatto, S.: On the set of images modulo viewpoint and contrast changes. In: Proc. IEEE Conf. on Comp. Vision and Pattern Recogn. (June 2009)
24. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: Proc. CVPR. Citeseer (2008)

25. Tomasi, C., Shi, J.: Good features to track. In: IEEE Computer Vision and Pattern Recognition (1994)
26. Valente, L., Tsai, R., Soatto, S.: Information gathering control via exploratory path panning. In: Proc. of the Conf. on Information Sciences and Systems (CISS) (March 2012)
27. Vedaldi, A., Soatto, S.: Features for recognition: viewpoint invariance for non-planar scenes. In: Proc. of the Intl. Conf. of Comp. Vision, pp. 1474–1481 (October 2005)
28. Wnuk, K., Soatto, S.: Multiple instance filtering. In: Proc. of NIPS (2011)

Classemes: A Compact Image Descriptor for Efficient Novel-Class Recognition and Search

Lorenzo Torresani, Martin Szummer, and Andrew Fitzgibbon

Abstract. In this chapter we review the problem of object class recognition in large image collections. We focus specifically on scenarios where the classes to be recognized are not known in advance. The motivating application is ”object-class search by example” where a user provides at query time a small set of training images defining an arbitrary novel category and the system must retrieve images belonging to this class from a large database. This setting poses challenging requirements on the system design: the object classifier must be learned efficiently at query time from few examples; recognition must have low computational cost with respect to the database size; finally, compact image descriptors must be used to allow storage of large collections in memory. We review a method that addresses these requirements by learning a compact image descriptor – *classemes* – yielding good categorization accuracy even with efficient linear classifiers. We also study how data structures and methods from text-retrieval can be adapted to enable efficient search of an object-class in collections of several million images.

1 Introduction

The accuracy of object category recognition is improving rapidly, particularly if the goal is to retrieve or label images where the category of interest is the primary subject of the image. However, existing techniques do not scale well to searching in large image collections. This chapter identifies three requirements for such scaling, and describes representations and retrieval methods that satisfy them.

Lorenzo Torresani
Dartmouth College, 6211 Sudikoff Lab, Hanover, NH 03755, U.S.A.
e-mail: lorenzo@cs.dartmouth.edu

Martin Szummer · Andrew Fitzgibbon
Microsoft Research, 7 JJ Thomson Avenue, Cambridge, CB3 0FB, U.K.
e-mail: {szummer,awf}@microsoft.com

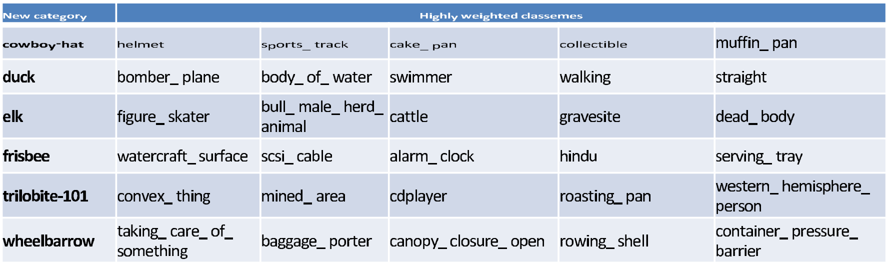
R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_5, © Springer-Verlag Berlin Heidelberg 2014

1. Interesting large-scale applications must support recognition of **novel categories**. This means that a new category can be presented as a set of training images, and a classifier learned from these new images can be run efficiently against the large database. Depending on the application, the user may define the query category either by supplying a set of image examples of the desired class, by performing relevance feedback on images retrieved for predefined tags, or perhaps by bootstrapping the recognition via text-to-image search [12]. In all these cases, the classifiers cannot be precomputed during an offline stage and thus both training and testing must occur efficiently at query-time in order to be able to provide results in reasonable time to the user.
2. Large-scale recognition benefits from a **compact descriptor** for each image, for example allowing databases to be stored in memory rather than on disk.
3. The ideal descriptor also provides good results with **linear classifiers**, such as linear SVMs, or tf-idf rankers [24], as these can be evaluated efficiently even on large databases.

Although a number of systems satisfy these desiderata for recognition of specific object-instances [27, 17], places [6] and whole scenes [43], we argue that these requirements cannot be addressed by traditional systems in the context of object-category recognition. This is due to the large computational and storage complexities of modern object-classifiers, which rely on high-dimensional image descriptors and expensive non-linear decision functions. For example, the current state-of-the-art in categorization is represented by multiple kernel combiners, such as the LP-β classifier [13], which compute non-linear (kernel-based) functions of multiple low-level features. These nonlinearities are critically necessary to achieve good classification accuracy: for example, compare in figure 2 the difference in accuracy between LP-beta13 and Xsvm, which represent, respectively, a multiple kernel combiner and a linear SVM trained on the same combination of low-level features. However, kernel-based classifiers cannot be used in our search setting, since the classes to recognize are not known at the time of the creation of the database and thus the kernel-distances cannot be precomputed: novel-class recognition with non-linear models would require evaluating the kernel-distance between each database image and (a subset of) the training images provided at query-time, which clearly cannot be accomplished in the real-time demanded by a search application. Furthermore, the multiple, high-dimensional image descriptors needed by LP-β would pose challenging storage requirements for large databases.

In this chapter we describe a system that addresses these requirements by using multiple-kernel combiners as an image representation instead of as a classification model: the idea is to use an image descriptor containing as entries the outputs of a set of *predefined* category-specific classifiers applied to the image. Because these basis-classifiers provide a rich coding of the image, simple *linear* models (e.g., linear SVMs) trained on this representation can approach state-of-the art accuracy, satisfying the requirements listed above. The obvious (but only partially correct) intuition is that a novel category, say `duck`, can be effectively expressed in terms of the outputs of the basis-classifiers (which we refer to as "classemes"), describing either objects similar to ducks, or objects seen in conjunction with ducks.

Table 1 Highly weighted classemes. Five classemes with the highest LP-β weights for the retrieval experiment, for a selection of Caltech256 categories. Some may appear to make semantic sense, but it should be emphasized that our goal is simply to create a useful feature vector, not to assign semantic labels. The somewhat peculiar classeme labels reflect the ontology used as a source of base categories.



New category	Highly weighted classemes				
cowboy-hat	helmet	sports_ track	cake_ pan	collectible	muffin_ pan
duck	bomber_ plane	body_ of_ water	swimmer	walking	straight
elk	figure_ skater	bull_ male_ herd_ animal	cattle	gravesite	dead_ body
frisbee	watercraft_ surface	scsi_ cable	alarm_ clock	hindu	serving_ tray
trilobite-101	convex_ thing	mined_ area	cdplayer	roasting_ pan	western_ hemisphere_ person
wheelbarrow	taking_ care_ of_ something	baggage_ porter	canopy_ closure_ open	rowing_ shell	container_ pressure_ barrier

In practice, the reason this descriptor will work is slightly more subtle. It is not required or expected that these base categories will provide useful semantic labels, of the form water, sky, grass, beak. On the contrary, the assumption is that modern category recognizers are essentially quite dumb; so a swimmer recognizer looks mainly for water texture, and the bomber␣plane recognizer contains some tuning for "C" shapes corresponding to the airplane nose, and perhaps the "V" shapes at the wing and tail. Even if these recognizers are perhaps overspecialized for recognition of their nominal category, they can still provide useful building blocks to the learning algorithm that learns to recognize the novel class duck. Table 1 lists some highly-weighted classemes used to describe an arbitrarily selected subset of the Caltech256 categories. Each row of the table may be viewed as expressing the category as a weighted sum of building blocks; however the true building blocks are not the classeme labels that we can see, but their underlying dumb components, which we cannot. To complete the duck example, it is a combination of body␣of␣water, bomber␣plane, swimmer, as well as walking and straight. To gain an intuition as to what these categories actually represent, Figure 1 shows the training sets for the latter two. Examining the training images, we suggest that walking may represent "inverted V outdoors" and straight might correspond to "clutter and faces".

2 Background

Before describing the details of the system, and experimental investigations, we shall briefly summarize related literature.

The closest existing approach is probably image representation via *attributes* [11, 19]. Here object categories are described by a set of boolean attributes, such as "has beak", "no tail", "near water". Classifiers for these attributes are built by acquiring labels using Amazon's Mechanical Turk. In contrast, classemes are not designed to

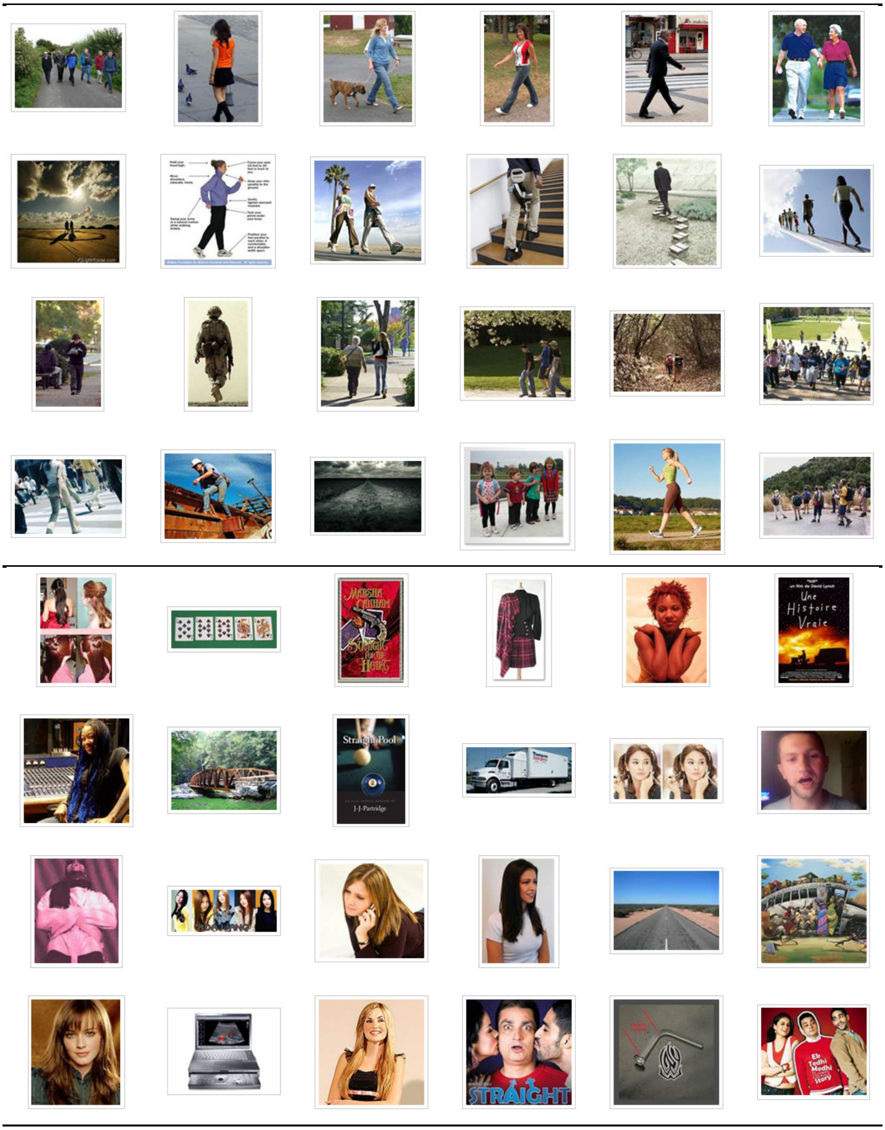


Fig. 1 Classeme training images. A subset of the training images for two of the 2659 classemes: `walking`, and `straight`. The top 150 training images are downloaded from Bing image search with no filtering or reranking. As discussed in the text, we do not require classeme categories to have a semantic relationship with the novel class; but to contain some building blocks useful for classification.

have specific semantic meanings, but rather to capture intersections of properties. Furthermore, they are trained using data directly obtained from web image search, without human cleanup. In addition, most prior attribute-based methods have relied on a “zero-shot” learning approach: instead of *learning* a classifier for a novel category from training examples, a user designs the classifier by listing attributes, limiting such systems to categories for which humans can easily extract attributes, and increasing the workload on the user even for such categories. A related idea is the representation of images in terms of distances to basis classes, which has been previously investigated as a way to define image similarities [42], to perform video search [15], or to enable natural scene recognition and retrieval [41].

The approach considered here is also evocative of Malisiewicz and Efros’s “Recognition by Association” [23], in which object classes are represented by sets of object *instances* to which they are associated. In contrast, classemes represent object classes as a combination of other object *classes* to which they are related. This change of viewpoint enables the use of powerful kernel-based classifiers.

Because classemes represent images by a (relatively) low-dimensional feature vector, the approach is related to dimensionality reduction techniques and methods to learn compact codes for images [43, 36, 33, 31, 9]. These data-driven techniques find low-dimensional, typically nonlinear, projections of a large feature vector representing each image, such that the low-dimensional vectors are an effective proxy for the original. These techniques can achieve tremendous compressions of the image (for example to 64 bits [43]), but are of course correspondingly lossy, and have not been shown to be able to retain category-level information.

It is also useful to make a comparison to existing categorization systems in terms of how far they meet the requirements we have set out. In the discussion below, let N be the size of the test set (i.e. the image database, which may in principle be very large). Let n be the number of images in the training set, typically in the range $5-100$ per class. Let d be the dimensionality of the representation stored for each image. For example, if a histogram of visual words is stored, d is the minimum of the number of words detected per image and the vocabulary size. For a GIST descriptor [28], d is of the order of 1000. For multiple-kernel techniques [13], d might be of the order of $20,000$. For the system in this paper, d can be as low as 1500, while still leveraging all the descriptors used in the multiple-kernel technique. Note that although we shall later be specific about the number of bits per element of d, this is not required for the current discussion.

Boiman *et al.* [4] shows one of the most intriguing results on the Caltech256 benchmark: a nearest-neighbour-like classifier on low-level feature descriptors produces excellent performance, especially with small training sets. Its training cost is effectively zero: assemble a bag of descriptors from the supplied training images (although one might consider building a kd-tree or other spatial data structure to represent large training sets). However, the test-time algorithm requires that each descriptor in the *test* image be compared to the bag of descriptors representing the class, which has complexity $O(nd)$. It may be possible to build a kd-tree for the test set, and reverse the nearest-neighbor lookups, but the metric is quite asymmetric, so it is not at all clear that this will preserve the properties of the method.

For the multilple-kernel system of Gehler and Nowozin [13] the complexity is again $O(nd)$, but with large d, and a relatively large constant compared to the nearest-neighbor approach.

Another class of related techniques is the use of classifier combination other than multiple-kernel approaches. Zehnder *et al.* [44] build a classifier cascade which encourages feature sharing, but again requires the set of classes to be predefined, as is true for Griffin and Perona [14] and Torralba *et al.* [37]. Heitz *et al.* [16] propose to learn a general cascade similar to classemes (although with a different goal). However, the classeme approach simplifies training by pre-training the first layer, and simplifies testing by successfully working with simple top-layer classifiers.

3 Method Overview

The approach is now described precisely, but briefly, with more details supplied in §4. There are two distinct stages: once-only classeme learning; followed by any number of object-category-related learning tasks. Note that there are distinct training sets in each of the two stages.

3.1 Classeme Learning

A set of C category labels is drawn from an appropriate term list. For each category $c \in \{1..C\}$, a set of training images is gathered by issuing a query on the category label to an image search engine.

A one-versus-all classifier ϕ_c is trained for each category. The classifier output is real-valued, and is such that $\phi_c(\mathbf{x}) > \phi_c(\mathbf{x}')$ implies that $\mathbf{x}$ is more similar to class c than $\mathbf{x}'$ is. Given an image $\mathbf{x}$, then, the feature vector (descriptor) used to represent $\mathbf{x}$ is the *classeme vector* $\mathbf{f}(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_C(\mathbf{x})]$.

Given the classeme vectors for all training images, it may be desired to perform some feature selection on the descriptors. We shall assume this has been done in the sequel, and simply write the classeme vector in terms of a reduced dimensionality $d \le C$, so $\mathbf{f}(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_d(\mathbf{x})]$. Where d is not specified it may be assumed that $d = C$.

Given the parameters of the ϕ_c, the training examples used to create the classemes may be discarded. We denote by $\boldsymbol{\Phi}$ the set of functions $\{\phi_c\}_{c=1}^{d}$, which encapsulates the output of the classeme learning, and properly we shall write $\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}; \boldsymbol{\Phi})$.

3.2 Using the Classemes

Given $\boldsymbol{\Phi}$, the rest of the approach is conventional. A typical situation might be that a new object category, or set of categories, is defined by a set of training images (note again that this is a new set of training images, unrelated to those used to build $\boldsymbol{\Phi}$). The training images are converted to classeme vectors, and then any classifier can be trained taking the classeme vectors as input. As shown in experiments, the features are sufficiently powerful that simple linear classifiers applied to the classemes can give accuracies commensurate with much more expensive classifiers applied to the low-level image features. Useful candidate classifiers might be those which make a sparse linear combination of input features, so that the test cost is a small fraction of d per image; or predicate-based classifiers so that test images with nonzero score can be retrieved rapidly using inverted files [27, 38], achieving test complexity sublinear in N, the size of the test set.

4 Further Details

Several details are now expanded.

4.1 Selecting Category Labels

The set of category labels used to build the classemes should consist primarily of visual concepts. This will include concrete nouns, but may also include more abstract concepts such as “person working”. The category labels should be chosen to be representative of the type of applications in which one plans to use the descriptors. As the focus of this study is general-category recognition, here we consider concepts selected from the Large Scale Concept Ontology for Multimedia (LSCOM) [26]. The LSCOM categories were developed specifically for multimedia annotation and retrieval, and have been used in the TRECVID video retrieval series. This ontology includes concepts selected to be useful, observable and feasible for automatic detection, and as such are likely to form a good basis for image retrieval and object recognition tasks. The LSCOM CYC ontology dated 2006-06-30 [22] was selected as the reference data set of concepts. From the initial 2832 unique concepts, the following categories were removed: 97 classes denoting abstract groups of other categories (marked in angle brackets in [22]); plural categories that also occurred as singulars; some people-related categories which were effectively near-duplicates. A total of $C = 2659$ categories were preserved by this filtering: the final list of concepts is available in [40]. Some examples have already been seen in table 1. This filtering was intentionally conservative in removing categories because, as discussed in the introduction, it is not easy to predict *a priori* what categories will be useful.

4.2 Gathering Category Training Data

For each category label, a set of training images was gathered by taking the top 150 images from the `bing.com` image search engine. For a general application these examples would not need to be manually filtered in any way, but in order to perform fair comparisons against the Caltech image database, near duplicates of images in that database were removed by a human-supervised process. Conversely, we did not remove overlap between the classeme *terms* and the Caltech categories

(28 categories overlap, see data on [40]), as a general-purpose system can expect to see overlap on a small number of queries. We also ran a test, not reported here, where classemes overlapping with Caltech256 labels were removed; the resulting performance was essentially unchanged.

4.3 Learning Classifiers ϕ_c

The classification model used for the $\phi_c(\cdot)$ is the LP-β kernel combiner of Gehler and Nowozin [13]. While they used 39 kernels, the experiments presented in this chapter are based on a set of 13 kernels. The kernels are defined in terms of the χ^2 distance between feature vectors as follows: $k(\mathbf{x},\mathbf{x}') = \exp(-\chi^2(\mathbf{x},\mathbf{x}')/\gamma)$, where γ is a hyper-parameter set as in [13] to be the average of the χ^2 distances in the training set. The following 13 feature types were used:

- *Kernel 1: Color GIST, d_1 = 960.* The GIST descriptor [28] is applied to color images. The images were resized to 32×32 (aspect ratio is not maintained), and then orientation histograms were computed on a 4×4 grid. Three scales were used with the number of orientations per scale being $8,8,4$.
- *Kernels 2-5: Pyramid of Histograms of Oriented Gradients, $d_{2..5}$* $= 1700$. The PHOG descriptor [7] is computed using 20 bins at four spatial pyramid scales.
- *Kernels 6-9: PHOG (2π unwrapped), $d_{6..9}$* $= 3400$. These features are obtained by using unoriented gradients quantized into 40 bins at four spatial pyramid scales.
- *Kernels 10-12: Pyramid self-similarity, $d_{10..12}$* $= 6300$. The Shechtman and Irani self-similarity descriptor [34] was computed as described by Bosch [5]. This gives a 30-dimensional descriptor at every 5th pixel. We quantized these descriptors into 5000 clusters using k-means, and a pyramid histogram was recorded with three spatial pyramid levels.
- *Kernel 13: Bag of words.* $d_{13} = 5000$ SIFT descriptors [21] were computed at interest points detected with the Hessian-Affine detector [25]. These descriptors were then quantized using a vocabulary of size 5000, and accumulated in a sparse histogram.

A binary LP-β classifier was trained for each classeme, using a setup following the one described in section 7 of [13] in terms of kernel functions, kernel parameters, values of ν and number of cross validations. The only difference is that the objective of their equation (4) was modified in order to handle the uneven training set sizes. We used $N_+ = 150$ images as positive examples, and one image chosen at random from each of the other training sets as negative examples, so $N_- = C-1$. The objective was modified by scaling the positive entries in the cost vector by (νN_+) and the negative entries by (νN_-). The cross-validation yields a per-class validation score which is used for feature selection.

4.4 Feature Selection

In order to perform feature selection on the classeme vectors $\mathbf{f}$, the classemes were first sorted in increasing order of cross-validation error. Given a desired feature dimensionality, d, the reduced classeme vector was obtained by selecting the first d components $\mathbf{f}(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_d(\mathbf{x})]$. Again in situations where d is not specified it may be assumed that $d = C$

4.5 Classeme Quantization

For a practical system, the classeme vectors should not be stored in double precision, but instead an explicit quantization of the values should be used. This may be achieved by a simple quantization, or by defining binary "decision stumps" or predicates. Quantization can be performed either at novel-category learning time (i.e. on the novel training set) or at classeme-learning time. For 1-bit quantization, simple thresholding at 0 was used. For higher quantization numbers, the following "histogram-equalized" quantization was used. Given a training set of classeme vectors $\{\mathbf{f}_i\}_{i=1}^n$, write $\mathbf{f}_i = [\phi_{ik}]_{k=1}^d$. Write the rows of the matrix $[\mathbf{f}_1, \ldots, \mathbf{f}_n]$ as $\mathbf{r}_k = [\phi_{ik}]_{i=1}^n$. To quantize to Q levels, quantization centres z_{iq} are chosen as follows: $\mathbf{r}'_k = \mathrm{sort}(\mathbf{r}_k)$, defining a matrix ϕ'_{ik}. Then make the set $Z_k = \{\phi'_{\lceil nq/(Q+1) \rceil, k}\}_{q=1}^Q$, and each value ϕ_{ik} is replaced by the closest value in Z_k.

5 Experiments

Given the simplicity of the approach, the first question that naturally arises is how it compares to the state-of-the-art recognition approaches. Here we compare to the LP-β kernel combiner as this is the current front-runner. Note that the key metric here is performance drop with respect to LP-β with the same 13 kernels used by classemes. As the classeme classifiers introduce an extra step in the recognition pipeline, performance might be expected to suffer from a "triangle inequality": the raw kernel combiner can optimize kernels for the final classes to recognize, while the classifiers using classemes as representation are forced to use the kernels trained on the LSCOM classes. The experiments show that this does happen, but to a small enough extent that the classemes remain competitive with the state of the art, and are much better than the closest "efficient" system.

There are two main experiments. In the first, we wish to assess the representational power of classemes with respect to existing methods, so we use the standard Caltech256 accuracy measure, with multiclass classifiers trained on all classes. In the second, we want to test classemes in a framework closer to their intended use, so we train one-vs-all classifiers on each class separately, and then report precision on ranking a set of images including distractors from the other classes.

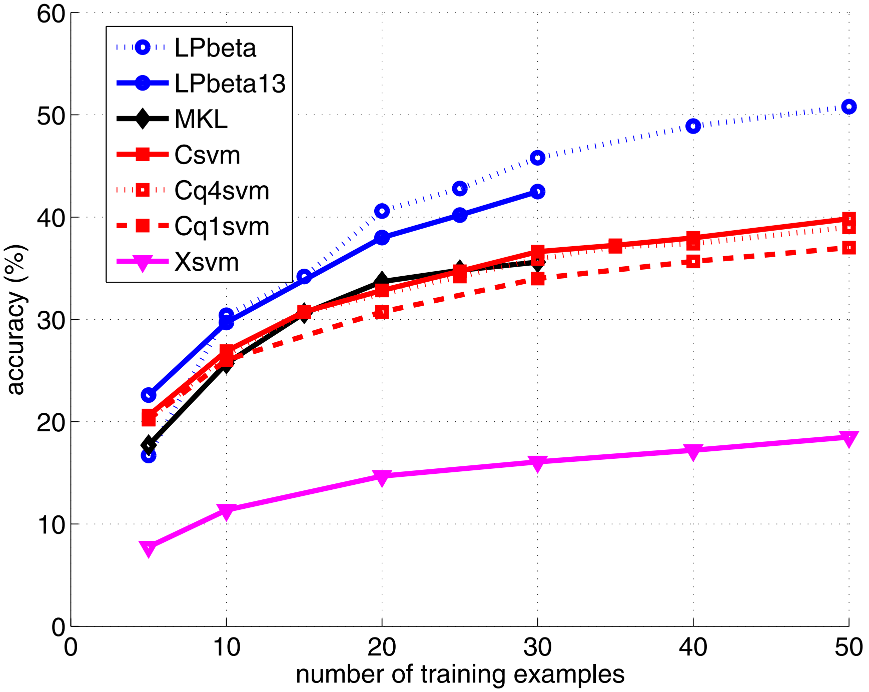


Fig. 2 **Caltech256**. A number of classifiers are compared on the Caltech256 dataset. **LPbeta** [13], **MKL**: Multiple Kernel learning [1], as implemented in [13], **LPbeta13**: LP-β on our low-level features (§4.3); **Xsvm** SVM trained on the concatenation of our low-level features. The classeme-based classifiers are: **Csvm**: SVM, floating point, $d = 1500$; **Cq4svm**: SVM, input quantized to 4 bits per channel (bpc), $d = 1500$; **Cq1svm**: SVM, input quantized to 1 bit, $d = 1500$. The key-result is this: on 30 training examples, and using the same underlying features, Csvm has 36% accuracy, and LPbeta13 has 42% accuracy, but the classeme-based system is orders of magnitude faster to train and test.

5.1 Experiment 1: Multiclass Classification

In this experiment we study the performance of classemes using the multiclass linear SVM of Joachims [18] as classification model, since this is an efficient classifier to train and test and thus it is well suited to our motivating problem. The SVM regularization parameter was set to be $\lambda = 3000$. All classeme-based results are presented for the case $d = 1500$, as using more than 1500 classemes was found to yield no further improvements.

Figures 2 shows the multi-class accuracy for different classifiers as a function of the number of training examples per class, using 25 test examples per category. It can be seen that the classeme-based SVM (Csvm) greatly outperforms an SVM directly trained on the same low-level features (Xsvm) and it matches the accuracy of the nonlinear classifier trained using multiple kernel learning [1]. Only LPbeta13

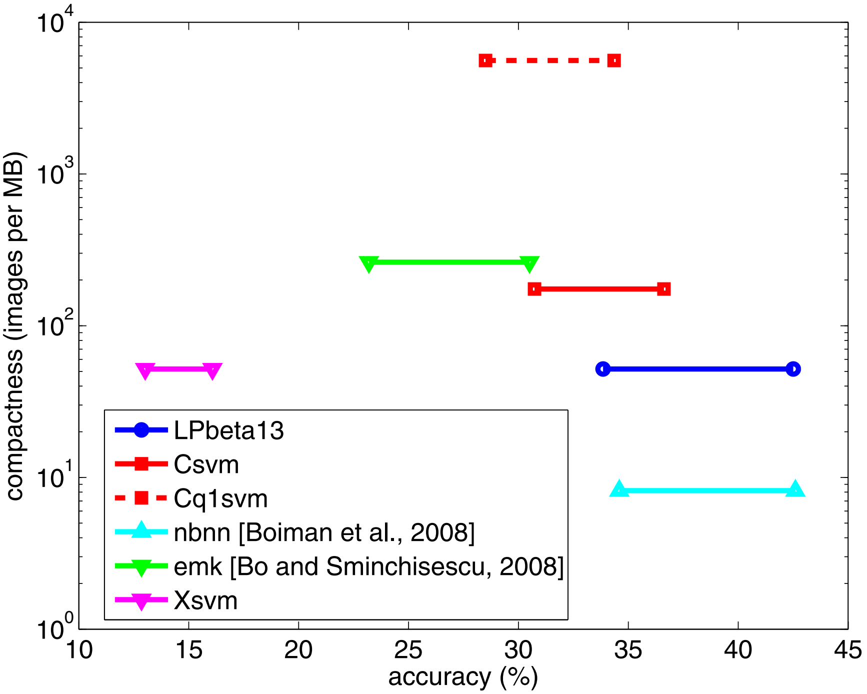


Fig. 3 Accuracy versus compactness of representation on Caltech-256. On both axes, higher is better. (Note logarithmic *y*-axis). The lines link performance at 15 and 30 training examples.

(the version of LP-β using the same low-level features exploited by classemes) provides higher accuracy. However the size of the representation is considerably reduced for classemes compared to LP-β: $2.5KB$ versus $23KB$. Furthermore, the training and test times of our approach are considerably lower than LP-β: training the multiclass classifier Csvm with 5 examples for each Caltech class takes about 9 minutes on a AMD Opteron Processor 280 2.4GHz while the method of [13] requires more than 23 hours on the same machine; predicting the class of a text example takes 0.18ms with our model and 37ms with LP-β.

In addition, when moving from floating point classemes (Csvm) to a quantization of 4 bits per channel (Cq4svm) the change in accuracy is negligible. Accuracy drops by only 2–4 percentage points using a 1 bit per channel SVM (Cq1svm, $d = 1500$, 187.5 bytes per image). However, this representation increases the number of images that can be stored in an index by a factor of 100 over LP-β, which is especially significant for RAM-based indices.

Figure 3 shows accuracy versus compactness for different classification systems. In this plot we include also the performance of Naive Bayes Nearest Neighbor (nbnn) [4] and Efficient Match Kernel (EMK) [3]. It can be seen that classemes

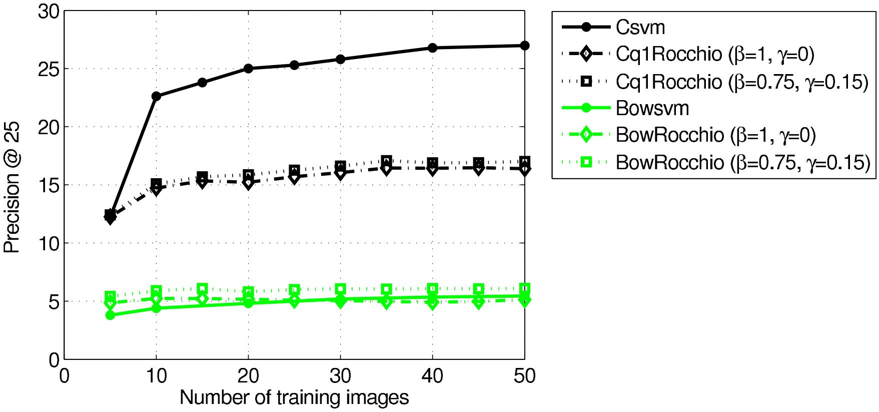


Fig. 4 Class retrieval in Caltech256. Percentage of the top 25 in a 6400-document set which match the query class. Random performance is 0.4%.

using 1 bit per channel provide a significant saving in terms of storage requirement compared to all other methods, while still yielding near state-of-the-art accuracy.

5.2 Experiment 2: Retrieval

The retrieval experiment attempts to gain insight into the behaviour of classemes in a class-retrieval task. A query against the database is specified by a set of training images taken from one category, and the retrieval task is to order the database by similarity to the query.

Evaluation on Caltech256. We start by studying performance on the Caltech256 data set. The test database is formed by sampling 25 images from each Caltech category. Success is measured as precision at 25: the proportion of the top 25 images which are in the same category as the query (training) set. The maximum score is 1, obtained if all the matching images are ranked above all the distractors. For this experiment, we compare classemes with bags of visual words (BOW), which are a popular model for efficient image retrieval. We use as BOW features the quantized SIFT descriptors of Kernel 13.

We consider two different retrieval methods. The first method is a linear SVM learned for each of the Caltech classes using the one-vs-the-rest strategy. We compare these classifiers to the Rocchio algorithm [24], which is a classic information retrieval technique for implementing relevance feedback. In order to use this method we represent each image as a document vector $\mathbf{d}(\mathbf{x})$. In the case of the BOW model, $\mathbf{d}(\mathbf{x})$ is the traditional *tf-idf*-weighted histogram of words. In the case of classemes instead, we define $\mathbf{d}(\mathbf{x})_i = [\phi_i(\mathbf{x}) > 0] \cdot \mathrm{idf}_i$, i.e. $\mathbf{d}(\mathbf{x})$ is computed by multiplying the binarized classemes by their inverted document frequencies. Given, a set of relevant training images D_r, and a set of non-relevant examples D_{nr}, Rocchio's algorithm

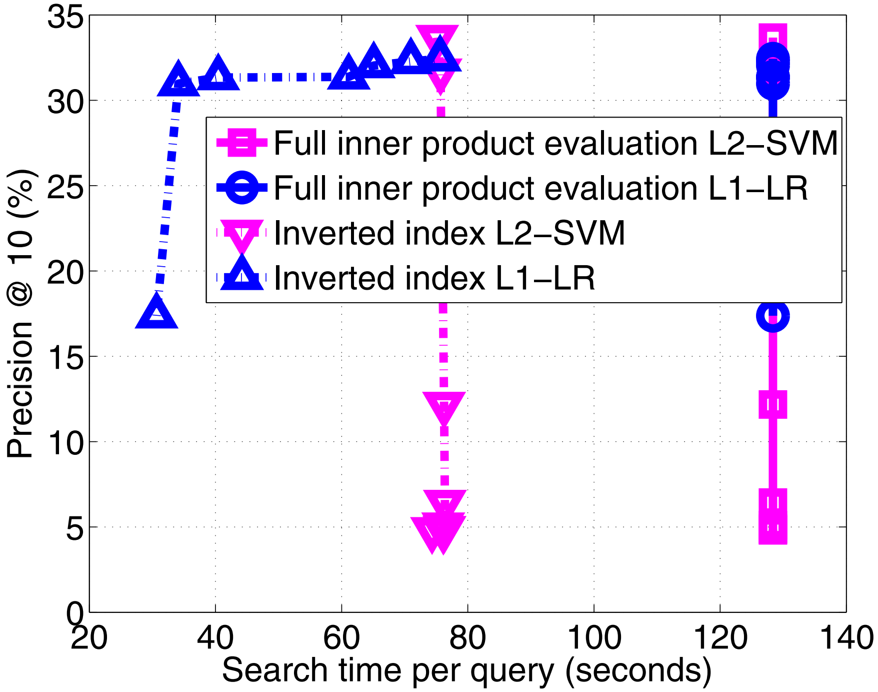


Fig. 5 Class-retrieval precision versus search time for the 10-million ImageNet database: x-axis is search time; y-axis shows percentage of true positives ranked in the top 10 (for each query class, the database contains $n^{-}_{test} = 9,671,611$ distractors and $n^{+}_{test} = 450$ true positives). The curve for each method is obtained by varying the hyperparameter in the learning objective of the classifier, thus producing different accuracy-speed tradeoffs (see details in the text).

computes the document query

$$\mathbf{q} = \beta \frac{1}{|D_r|} \sum_{\mathbf{x}_r \in D_r} \mathbf{d}(\mathbf{x}_r) - \gamma \frac{1}{|D_{nr}|} \sum_{\mathbf{x}_{nr} \in D_{nr}} \mathbf{d}(\mathbf{x}_{nr}) \tag{1}$$

where β and γ are scalar values. The algorithm then retrieves the database documents having highest *cosine similarity* with this query. In our experiment, we set D_r to be the training examples of the class to retrieve, and D_{nr} to be the remaining training images. We report results for two different settings: $(\beta,\gamma) = (0.75, 0.15)$, and $(\beta,\gamma) = (1,0)$ corresponding to the case where only positive feedback is used.

Figure 4 shows that methods using classemes consistently outperform the algorithms based on traditional BOW features. Furthermore, SVM yields much better precision than Rocchio's algorithm when using classemes.

Evaluation on ImageNet (10M images). We now move on to present results on the large-scale ImageNet dataset [8], which includes about 10-million images representing over 15,000 categories (in this experiment we used 15,203 classes).

We randomly selected 400 categories as query classes. For each of these classes we capped the number of true positives in the database to be $n_{test}^{+} = 450$. The total number of distractors for each query is $n_{test}^{-} = 9,671,611$. Due to the size of the collection, we restrict our analysis to the binary version of classemes (1 bit per channel), using $d = 2659$.

We use this large-scale database to evaluate the speed-up achievable by implementing linear classification via inverted lists [24]. Inverted lists (also known as inverted indices) have been widely used for image search but predominantly for retrieval of near-duplicates or particular object-instances [35, 27, 30]. Instead here we adopt them to efficiently calculate the inner product between the weight vector learned at query-time and the binary classeme vector associated to each database image. This can be achieved by storing an inverted list for each classeme feature, enumerating the database images containing that particular classeme entry. The inverted lists allow the ranker to skip over classemes having value zero. A further speedup can be obtained by using a sparse classification model where the weight vector is constrained to have very few non-zero entries so that the evaluation cost will be a small fraction of the number of features (d). We use an ℓ_1-regularized logistic regression [10] (L1-LR) to test the advantages of a sparse classifier over the traditional ℓ_2-regularized SVM (L2-SVM).

For each query category we trained these two classification models using the one-vs-the-rest strategy, with a training set consisting of $n^{+} = 10$ positive examples and $n^{-} = 15,202$ negative images obtained by sampling one training image for each of the negative classes. The results are summarized in figure 5. The x-axis shows average retrieval time per query, measured on a single-core computer with 16GB of RAM and an Intel Core i7-930 CPU @ 2.80GHz. The y-axis reports precision at 10 which measures the proportion of true positives in the top 10. The performance curve of each method was generated by varying the regularization hyperparameter λ in the learning objective of the classifier. While λ is traditionally viewed as controlling the bias-variance tradeoff, for the L1-LR classifier it can be interpreted as a parameter balancing generalization accuracy versus sparsity, and thus retrieval speed. It can be seen that inverted indices speed up considerably the retrieval, particularly in the case of L1-LR which tends to generate sparser weight vectors for which inverted indexing is especially advantageous: using this model ranking the entire 10-million dataset takes about 30 seconds, with an average precision@10 above 30%. As a reference, random retrieval would produce precision@10 roughly equivalent to 0.005%. Learning a L1-LR or an L2-SVM classifier for a query category in this experiment takes roughly 2 seconds.

We would like also to comment on the memory usage. Representing the database as a bit-map of all classemes would require a space of $(2659/8) \times N$ bytes for a database containing N images, which in this case amounts to about 3GB. The inverted list architecture requires more space. We represented the image IDs in inverted files using one byte per image: we achieve this by storing only ID displacements (which in our experiment happened to be always smaller than 255) between consecutive images in the list. Using this encoding, the total storage requirement for the 10M data set was roughly 9GB.

6 Discussion

In this chapter we have describe the learning of the classeme descriptor which is a representation intended to be useful for efficient high-level object recognition. By using the noisy training data from web image search in a novel way – to train "category-like" classifiers – the descriptor is essentially given access to knowledge about what humans consider "similar" when they search for images on the web (note that most search engines are considered to use "click-log" data to rank their image search results, so the results do reflect human preferences). The experiments have shown that this knowledge is effectively encoded in the classeme vector, and that this vector, even when quantized to below 200 bytes per image, gives competitive object category recognition performance.

A natural question is whether the weakly trained classemes actually do contain any semantic information, although we have emphasized that this is not the main motivation for their use.

We have focused here on object category recognition as characterized by the Caltech256 training data, which are adequate for clip-art search, but which will not be useful for, for example, home photo retrieval, or object indexing of surveillance footage. It should be straightforward to retrain the classemes on images such as the PASCAL VOC images, but a sliding-window approach would probably be required in order to achieve good performance.

Classemes were originally introduced in [39]. A further extension of this idea was presented in [2] where the classeme classifiers were trained jointly (as opposed to independently) by directly optimizing an objective measuring linear classification accuracy. A related approach is proposed by Li et al. [20] where the location-dependent output of object detectors evaluated on the image is used as a representation. The advantage of this descriptor is that it encodes spatial information; furthermore, object detectors are more robust to clutter and uninformative background than classifiers evaluated on the entire image. In [9] classemes were empirically shown to be useful also for low-level retrieval tasks such as finding images of the same scene as the query, particularly when used in conjunction with local-appearance descriptors [21, 29]. Also in [9], several binary encoding methods are presented to further compress the size of classemes while preserving their good retrieval properties. Some of these compression methods as well as a top-*k* ranking scheme are explored in [32] to further boost the efficiency of object-class retrieval in large databases using classemes.

Additional material including the list of classeme labels, the classeme training images, precomputed feature vectors for standard datasets, as well as software to extract this descriptor may be obtained from [40].

References

1. Bach, F.R., Lanckriet, G.R.G., Jordan, M.I.: Multiple kernel learning, conic duality, and the SMO algorithm. In: ICML (2004)
2. Bergamo, A., Torresani, L., Fitzgibbon, A.: Picodes: Learning a compact code for novel-category recognition. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K. (eds.) Advances in Neural Information Processing Systems 24, pp. 2088–2096 (2011)

3. Bo, L., Sminchisescu, C.: Efficient Match Kernel between Sets of Features for Visual Recognition. Adv. in Neural Inform. Proc. Systems (December 2009)
4. Boiman, O., Shechtman, E., Irani, M.: In defense of nearest-neighbor based image classification. In: Proc. Comp. Vision Pattern Recogn (CVPR) (2008)
5. Bosch, A.: Image classification using rois and multiple kernel learning (2010), `http://eia.udg.es/~aboschr/Publicacions/bosch08a_preliminary.pdf`
6. Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total recall: Automatic query expansion with a generative feature model for object retrieval. In: Intl. Conf. Computer Vision (2007)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (1), pp. 886–893 (2005)
8. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
9. Douze, M., Ramisa, A., Schmid, C.: Combining attributes and fisher vectors for efficient image retrieval. In: Proc. Comp. Vision Pattern Recogn, CVPR (2011)
10. Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., Lin, C.-J.: Liblinear: A library for large linear classification. J. of Machine Learning Research 9, 1871–1874 (2008)
11. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: Proc. Comp. Vision Pattern Recogn. (CVPR), pp. 1778–1785 (2009)
12. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google's image search. In: ICCV, pp. 1816–1823 (2005)
13. Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV (2009)
14. Griffin, G., Perona, P.: Learning and using taxonomies for fast visual categorization. In: Proc. Comp. Vision Pattern Recogn. (CVPR) (2008)
15. Hauptmann, A.G., Yan, R., Lin, W.-H., Christel, M.G., Wactlar, H.D.: Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. IEEE Transactions on Multimedia 9(5), 958–966 (2007)
16. Heitz, G., Gould, S., Saxena, A., Koller, D.: Cascaded classification models: Combining models for holistic scene understanding. In: Advances in Neural Information Processing Systems (NIPS), pp. 641–648 (2008)
17. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)
18. Joachims, T.: An implementation of support vector machines (svms) in c (2002)
19. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: Proc. Comp. Vision Pattern Recogn. (CVPR) (2009)
20. Li-Jia Li, E.P.X., Su, H., Fei-Fei, L.: Object bank: A high-level image representation for scene classification semantic feature sparsification. In: NIPS (2010)
21. Lowe, D.: Distinctive image features from scale-invariant keypoints. Intl. Jrnl. of Computer Vision 60(2), 91–110 (2004)
22. LSCOM (2006), `http://lastlaugh.inf.cs.cmu.edu/lscom/ontology/LSCOM-20060630.txt`, `http://www.lscom.org/ontology/index.html` (Cyc ontology dated June 30, 2006)
23. Malisiewicz, T., Efros, A.A.: Recognition by association via learning per-exemplar distances. In: Proc. Comp. Vision Pattern Recogn. (CVPR) (2008)
24. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge Univ. Press (2008)

25. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. Intl. Jrnl. of Computer Vision 60(1), 63–86 (2004)
26. Naphade, M., Smith, J.R., Tesic, J., Chang, S.-F., Hsu, W., Kennedy, L., Hauptmann, A., Curtis, J.: Large-scale concept ontology for multimedia. IEEE MultiMedia 13(3), 86–91 (2006)
27. Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: Proc. Comp. Vision Pattern Recogn. (CVPR), pp. 2161–2168 (2006)
28. Oliva, A., Torralba, A.: Building the gist of a scene: The role of global image features in recognition. Visual Perception, Progress in Brain Research 155 (2006)
29. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 143–156. Springer, Heidelberg (2010)
30. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR (2008)
31. Raginsky, M., Lazebnik, S.: Locality-sensitive binary codes from shift-invariant kernels. In: Advances in Neural Information Processing Systems (NIPS) (2010)
32. Rastegari, M., Fang, C., Torresani, L.: Scalable object-class retrieval with approximate and top-k ranking. In: ICCV, pp. 2659–2666 (2011)
33. Salakhutdinov, R., Hinton, G.: Semantic hashing. Int. J. Approx. Reasoning 50, 969–978 (2009)
34. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: Proc. Comp. Vision Pattern Recogn. (CVPR) (June 2007)
35. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: ICCV (2003)
36. Torralba, A., Fergus, R., Weiss, Y.: Small codes and large image databases for recognition. In: Proc. Comp. Vision Pattern Recogn. (CVPR) (2008)
37. Torralba, A., Murphy, K.P., Freeman, W.T.: Sharing visual features for multiclass and multiview object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(5), 854–869 (2007)
38. Torresani, L., Szummer, M., Fitzgibbon, A.: Learning query-dependent prefilters for scalable image retrieval. In: Proc. Comp. Vision Pattern Recogn. (CVPR), pp. 2615–2622 (2009)
39. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 776–789. Springer, Heidelberg (2010)
40. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes, web page (2010), http://www.cs.dartmouth.edu/~lorenzo/projects/classemes
41. Vogel, J., Schiele, B.: Semantic modeling of natural scenes for content-based image retrieval. Intl. Jrnl. of Computer Vision 72(2), 133–157 (2007)
42. Wang, G., Hoiem, D., Forsyth, D.: Learning image similarity from flickr using stochastic intersection kernel machines. In: Intl. Conf. Computer Vision (2009)
43. Weiss, Y., Torralba, A., Fergus, R.: Spectral hashing. In: NIPS (2008)
44. Zehnder, P., Koller-Meier, E., Gool, L.V.: An efficient shared multi-class detection cascade. In: British Machine Vision Conf. (2008)

The Enhanced Flock of Trackers

Tomáš Vojíř and Jiří Matas

Abstract. The paper presents contributions to the design of the Flock of Trackers (FoT). The FoT estimates the pose of the tracked object by robustly combining displacement estimates from a subset of local trackers that cover the object and has been. The enhancements of the Flock of Trackers are: (i) new reliability predictors for the local trackers - the Neighbourhood consistency predictor and the Markov predictor, (ii) new rules for combining the predictions and (iii) introduction of a RANSAC-based estimator of object motion. The enhanced FoT was extensively tested on 62 sequences. Most of the sequences are standard and used in the literature. The improved FoT showed performance superior to the reference method. For all 62 sequences, the ground truth is made publicly available.

1 Introduction

The term "visual tracking" covers a broad range of methods for estimation of the pose and state of some entity in a sequence of images assuming temporal dependence of the estimated quantities. The complexity of the tracked entity may range from a rectangular region to a deformable or articulated object like human or animal body. The pose refers to geometric parameters of the entity, in 2D tracking typically a position, often with scale and rotation. The state represents all other information about the object, e.g. its past appearance, dynamics or even a discriminative classifier for redection [8, 6] or pointers to objects in the image with correlated motion [5].

Short-term frame-to-frame tracking is the most widely used form of visual tracking. It formulates the problem as a sequential casual estimation of the pose of an object in the next frame given the pose in the current frame. Short term trackers do not consider the problems of object re-detection after occlusion or disappearance -

Tomáš Vojíř · Jiří Matas
The Center for Machine Perception, FEE CTU, Prague, Karlovo namesti 13, 121 35 Praha 2, Czech Republic
e-mail: {vojirtom,matas}@cmp.felk.cvut.cz

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_6, © Springer-Verlag Berlin Heidelberg 2014

some pose parameters are always output, regardless of the fact the tracked entity is no more (visible) in the field of view. Prominent examples of short term trackers are the Lucas-Kanade [11] and mean-shift [3] trackers. The popularity of short-term trackers stems from their simplicity and, consequently, high speed and applicability in a wide range of conditions.

The Flock of Trackers (FoT). Recently, Kolsch and Turk [9] and Kalal et al. [8, 6] have shown that a very robust short-term tracker is obtained if a collection (a ”flock”) of local short-term trackers covering the object is run in parallel and the object motion is estimated from the displacements or, more generally, from transformation estimates of the local trackers. Each local tracker is attached to a certain area specified in the object coordinate frame. Following [8, 6, 14], we adopted the Lucas-Kanade [11] algorithm for local tracking.

The block structure of the Flock of Trackers is illustrated in Fig. 1. In its simplest form, the FoT requires only two components: a local short-term tracker, multiple instance of which are run on different areas of the object and provide image-to-image correspondence, and a (global) object motion estimation module robustly combining the local estimates.

The FoT is a very attractive short-term tracker. In comparison to many recently published methods, it is relatively simple and transparent and yet its performance is close to the state of the art [14]. Its internal structure allows handling heavy partial occlusion and local non-rigid changes and it makes the pose estimation robust, since it does not depend on a single global property of the object but rather on a composition of many local (weak) features. The FoT is slower then a monolithic short-term tracker, but not by orders of magnitude since the local trackers operate on small patches are thus fast.

In this chapter we show that the performance of the FoT is significantly improved if the object motion module is provided with a confidence measure in the reliability of the local tracker motion estimates. We propose (i) new reliability predictors for the local trackers, (ii) new rules for combining the predictions and (iii) introduce a new, RANSAC-based estimator of the object motion.

The local tracker reliability predictors presented in the chapter fall into two groups. The first group contains methods that are applicable to any short-term tracker and includes estimators based on the apparent magnitude of the intra-frame appearance change like the sum of squared intensity differences (SSD), the normalized cross-correlation (NCC) and the forward-backward procedure (FB). The forward-backward procedure runs the Lucas-Kanade tracker [11] twice, once in the forward direction time, as in a standard implementation, and then a second (extra) run is made in the reverse direction. The probability of being an oulier, i.e. of tracker failure, is a function of the distance of the initial position and the position reached by the FB procedure.

The second group of local tracker reliability predictors includes two estimators applicable only to trackers running multiple local trackers, such as the FoT. One, a new predictor based on the consistency of motion estimates in a local neighbourhood (P_N), exploits the fact that it is unlikely for a local motion estimate to be correct if it differs significantly from other motion estimates in its neighbourhood. The second new predictor reflects past performance of the local tracker. If a local tracker motion estimate has (often) been an outlier in the (recent) past, i.e. it was inconsistent with the global motion estimate, it is not likely to be correct in the current frame. This occurs for instance when the area covered by the local tracker is occluded or because the area is not suitable for tracking (e.g. it has near constant intensity). This local predictor of tracker reliability is called the Markov predictor (P_M), since it models the sequence of predicted states (either inlier or outlier) as a Markov chain.

The Markov predictor uses the global object motion estimates as ground truth in judging the correctness of local tracker motion. Naturally, the global motion estimate may be correct or incorrect, but the latter case need not be considered since the FoT has failed anyway.

Combination of predictors. With the exception of the forward-backward procedure, the evaluation of the reliability prediction is fast in comparison with the time it takes to calculate the local motion estimate. It is therefore reasonable to combine all fast predictors to achieve high accuracy and avoid, if possible, the FB procedure.

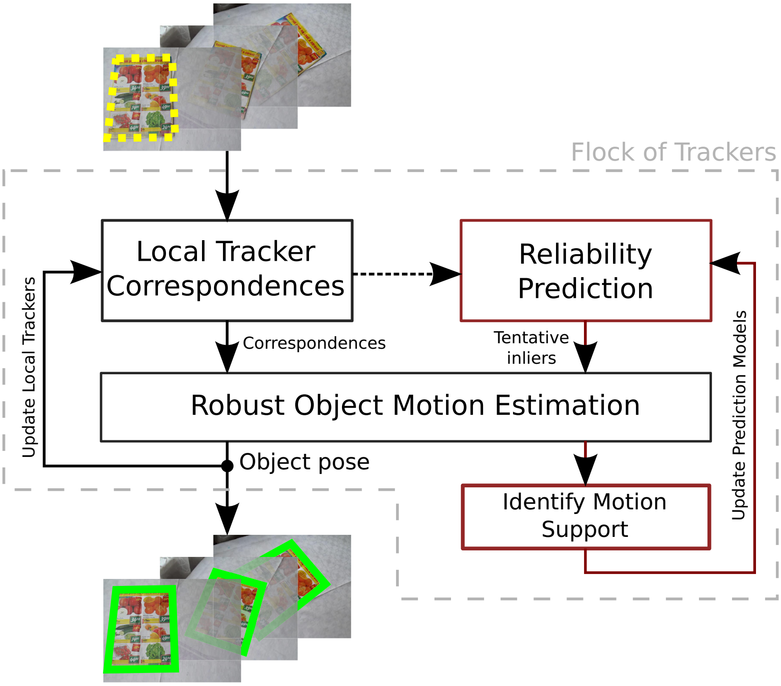


Fig. 1 Block structure of the Flock of Trackers (FoT). Correspondences (motion estimates) between two images, given the previous object pose and two consecutive images, are produced by local trackers. Simultaneously, reliability is estimated for each motion estimate. The object pose in the next frame is robustly estimated from a subset of most reliable motion estimates called tentative inliers.

We show that the Markov and Neighbourhood predictors, both on their own and when combined with the normalized cross-correlation predictor P_ρ, are more reliable than the normalized cross-correlation predictor combined with the FB procedure used in the reference method [7]. The new predictors are computed efficiently at a cost of about 10% of the complete FoT procedure whereas the forward-backward procedure slows down tracking approximately by a factor of two, since the most time consuming part of the process, the Lucas-Kanade local optimization, is run twice. With the proposed combination of reliability predictors, a FoT with much higher robustness to local tracker problems is obtained with negligible extra computational cost.

We introduce and compare two *predictor combination* schemes: a predictor combination method approximating a likelihood-based decision (denoted as $\mathscr{P}_\Theta$) and a simple ad-hoc predictor combination (denoted as $\mathscr{P}_\wedge$ combination). The ad-hoc combination sets a reliability threshold for each predictor (i.e. P_ρ, P_M, P_N) and the local tracker has to satisfy all of the condition to be used for pose estimation. The likelihood-based method orders the local trackers based on their likelihood of being correct. It allows choosing either the n best local trackers or a variable size subset that on average maintains a certain level of the inlier ratio for robust object pose estimation. In experiments, we set the operating point of the $\mathscr{P}_\Theta$ combination so that the number of the local trackers in the predicted inlier set (i.e. points, from which the object pose is estimated) is the same in each frame for the $\mathscr{P}_\wedge$ and the $\mathscr{P}_\Theta$ methods. The methods are evaluated by inlier prediction precision and by how many true inliers were in a predicted set.

Finally, we turn our attention to *robust object motion estimation* that takes as input the local motion estimates equipped with their reliability predictions.

The reference method is the Median-Flow (MF) [7] tracker which was shown to be comparable to the state-of-the-art where object motion, which is assumed to be well modelled by translation and scaling, is estimated by the median of a subset of local tracker responses.

Theoretically, the median with the breakdown point 0.5 is robust up to 50% of corrupted data. Since a single displacement vector give an estimate of the translation, the median as a translation estimator is robust up to 50% of incorrect local trackers. For scale estimation a ratio of pairwise distances of local trackers is used as an estimate of scale change, therefore a median is robust up to $100 \times (1-\sqrt{0.5})\% \doteq 29\%$ of incorrect local trackers for scale estimation step.

In practice, the outlier tolerance is often lower since the outliers ”conspire”. The outlier motion estimates originate from occluded or background areas. All local motion estimates in such areas are typically consistent with a motion of the occluding object or the background, i.e. they are higher or lower than the tracked object motion and bias the median based estimate. In challenging tracking scenarios presented in Section 6, the inlier percentage was often not sufficient for the median-based estimation of global motion and it failed when used without local tracker reliability prediction.

We show that RANSAC [2, 4] followed by least square fitting of inliers (LS) as model estimator is a preferable alternative to the median estimator. There are three main advantages of using the RANSAC+LS estimator: the model is estimated consistently (i.e. translation estimation is not separated from scale estimation), the motion model is not constrained to translation, scale and rotation; affine transformation or a homography requires only to change the sample size and it handles higher outlier percentages.

The rest of the paper is structured as follows. Section 3 proposes two new predictors of local tracker failure and discusses the predictor parameters selection. Section 4 discusses predictor combinations. Section 5 introduce RANSAC as a model estimator. Finally, Section 6 evaluates the proposed improvements. Conclusions are given in Section 7. This paper is an extension of a workshop paper [14].

2 Related Work

The work presented in the chapter builds on Kalal et al. [7] who mainly used the FoT as a tracking component of the powerful Tracking-Learning-Detection system, or TLD in short, long-term tracker [8]. Interestingly, with the improvements in presented in the chapter, the FoT with the combined new reliability prediction of local trackers approaches performance of the TLD framework on sequences where redetection is not needed, and yet is significantly faster.

The baseline FoT [7] places local trackers on a regular grid, i.e. the local trackers cover the object uniformly. Object motion, which is assumed to be well modelled by translation and scaling, is estimated by the median of a subset of local tracker displacement estimates (translation) and the median of the relative change of distance between positions of local tracker pairs (scale).

For reliability prediction of local trackers, Kalal et al. [7] use several standard local tracker filtering methods, namely the normalised cross-correlation (or sum of squared differences) of the corresponding patches, and the consistency of the forward-backward procedure.

The original idea of exploiting a collection of trackers goes back at least to Klsch et al. [9] who proposed the Flock of Features for fast hand tracking using local trackers (Lucas-Kanade [11]) with color histograms for replenishing of failed local trackers. They also enforce ”flock behaviour” [12] to detect failing local trackers. The output of their tracker is the median position of the local trackers, which manifests the flock behaviour.

Adam et al. [1] introduced FragTrack, which represents object by multiple patches (histograms of local areas). During tracking, each patch votes for an object pose by comparing its histogram to neighbourhood patch histograms. Robust statistics is then used to combine votes from multiple patches. Nejhum et al. [13] combine global description (histogram over the whole object) and a small number of rectangular blocks (weighted histograms) to determinate the most probable object location. An approximate boundary contour is then extracted using graph-cut segmentation. Block positions and weights are then updated. Kwon et al. [10] use local patch-based appearance model and an efficient scheme for online evolution of the local patch topology. For each frame, the Maximum a Posteriori (MAP) estimate is computed from the observation and transition models of local patches in a Bayesian manner.

3 Tracker Reliability Prediction Methods

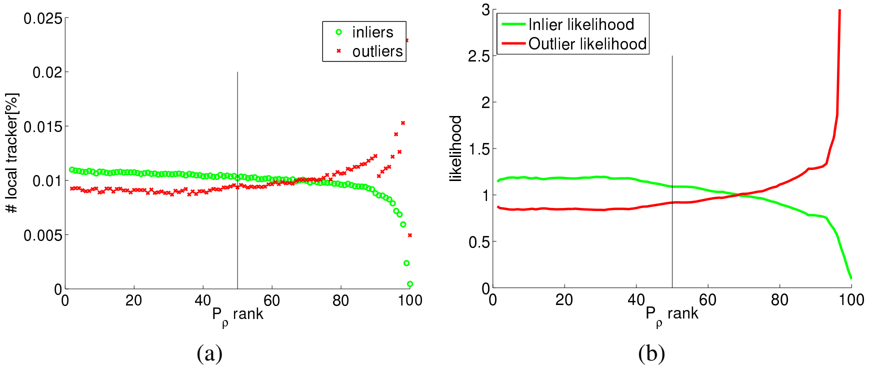
In this section, two novel methods for the local tracker reliability prediction are presented: section 3.3 describes the Neighbourhood consistency reliability predictor and section 3.4 presents the Markov predictor based on the long-term behaviour of the local tracker. Before that, two predictors used in the literature are described: the reliability predictor P_ρ based on normalised cross-correlation of the corresponding patches in consecutive frames (section 3.1) and the forward-backward predictor (section 3.2

3.1 The NCC Reliability Predictor P_ρ

The first step of the predictor is to calculate for each local tracker the normalized cross-correlation NCC, eq. 1 between the patches T_1 and T_2 at corresponding positions and size (w,h) given by the motion estimate:

$$\begin{aligned} T_1'(x,y) &= T_1(x,y) - 1/(w\cdot h)\cdot \textstyle\sum_{x',y'} T_1(x',y') \\ T_2'(x,y) &= T_2(x,y) - 1/(w\cdot h)\cdot \textstyle\sum_{x',y'} T_2(x',y') \\ \mathrm{NCC} &= \frac{\sum_{x,y}(T_1'(x,y)\cdot T_2'(x,y))}{\sqrt{\sum_{x,y} T_1'(x,y)^2 \cdot \sum_{x,y} T_2'(x,y)^2}} \end{aligned} \tag{1}$$

The P_ρ predictor, introduced in [7] works as a ranking filter. It is difficult to find a general function linking the NCC to tracker reliability, since NCC values for all local trackers may change dramatically from frame to frame due to an illumination change, shadows, small drifts, etc. The local trackers are thus only sorted by NCC and their rank is used as a predictor.



(a) (b)

Fig. 2 Properties of the P_ρ predictor averaged over a subset of the test sequences and all frames. (a) The histogram of NCC ranks ρ for local trackers with correct motion estimates (green) and incorrect motion estimates (red). (b) The correct/incorrect motion estimate ratio as a function of NCC rank ρ (green), the reciprocal value in red.

The top 50% of the local trackers are predicted to be inlier (correct motion estimate), the rest as outliers (incorrect motion estimate). The threshold was selected empirically. Figure 2(a) shows the histogram of ranks for both inliers and outliers and supports the choice to filter 40%-50% of the worst local trackers, as the probability of being an inlier in the bottom half of the ranks is smaller than the probability of being an outlier. This is illustrated in figure 2(b) in terms of the likelihood ratio of being an inlier/outlier. Another interesting fact is that probability of being an outlier slightly rises around the 1-5 rank. This is caused by local trackers that are placed on the background (due to the bounding box representation of object or tracker drift) where a zero motion is estimated. The NCC values are very high on the static background.

Experimentally we observed that the P_ρ predictor is sensitivity to local tracking precision of the model and candidate patch - small misalignment may induce arbitrary large similarity difference. This often happens for articulated or non-rigid objects.

3.2 *The Forward-Backward Reliability Predictor P_{FB}*

This underlying idea behind the forward-backward predictor is that the process of motion estimation between two images is independent of the order of the images. In an error-free situation, tracking an image region using Lucas and Kanade [11] gradient optimization from frame $1 \rightarrow 2$ and then the resulting image region from $2 \rightarrow 1$ will end up in the original position in the frame 1.

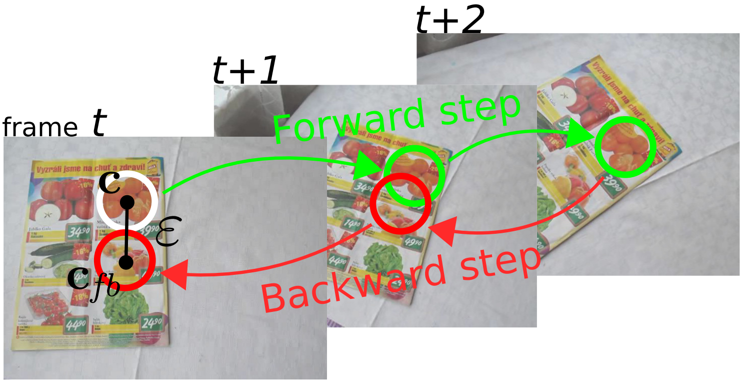
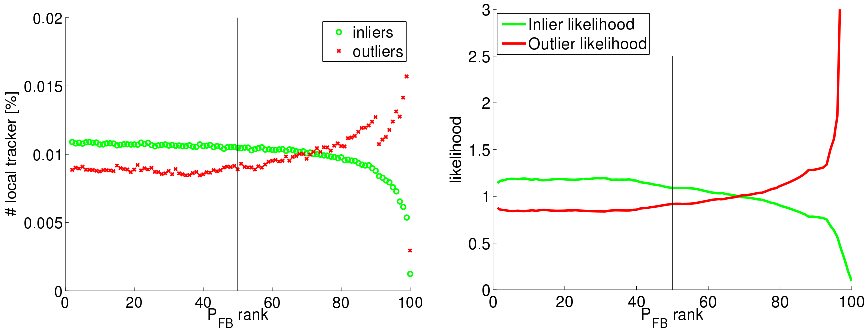


Fig. 3 A reference point of a regions of interest is tracked forward in time (from frame $t \rightarrow t+1 \rightarrow t+2$) and then backward. The positional forward-backward error $\varepsilon = \| \mathbf{c} - \mathbf{c}_{fb} \|^2$ is then used as a measure of tracker reliability.

When the deviation from the original position in frame 1 is large, then at least one of the two motion estimates is inaccurate. It is not unreasonable to assume that reliability of the motion estimate is a monotonic function of the distance of the original position and the position reached by the forward-backward procedure. The process may be generalised and the forward and backward direction tracking computed over larger number of frames. This is illustrated in Fig. 3.

Figure 4(a) shows the histogram of FB distance ranks for correct and incorrect motion estimates and supports the choice to filter $30\% - 50\%$ of the worst local trackers, as the probability of being an inlier in the bottom half of the ranks is smaller than the probability of being an outlier. Figure 4(b) depicts the ratio of being an inlier or outlier respectively as function of the rank. Similarly to P_ρ predictor, the probability of being an outlier rises around the 1-5 rank. This is also caused by local trackers that are on the background and thus are consistent with FB procedure.



(a) (b)

Fig. 4 Properties of the P_{FB} predictor averaged over a subset of the test sequences and all frames. (a) The histogram of FB ranks for local trackers with correct motion estimates (green) and incorrect motion estimates (red). (b) The correct/incorrect motion estimate ratio as a function of the FB rank (green), the reciprocal value in red.

3.3 The Neighbourhood Consistency Predictor P_N

The assumption behind the neighbourhood consistency predictor is that the motion of neighbouring local trackers is often very similar, whereas a failing local tracker returns a random displacement.

The P_N predictor is implemented as follows. For each local tracker i, a set of neighbouring local trackers N_i is defined. In all experiments, N_i included the four nearest neighbours of i. The neighbourhood consistency score S_i^N, the number of the neighbourhood local trackers that have a similar displacement. The process is visualised in Fig. 5.

We tested two definitions of the scoring functions given in eq. 2 and eq. 3. The latter has superior performance and was adopted.

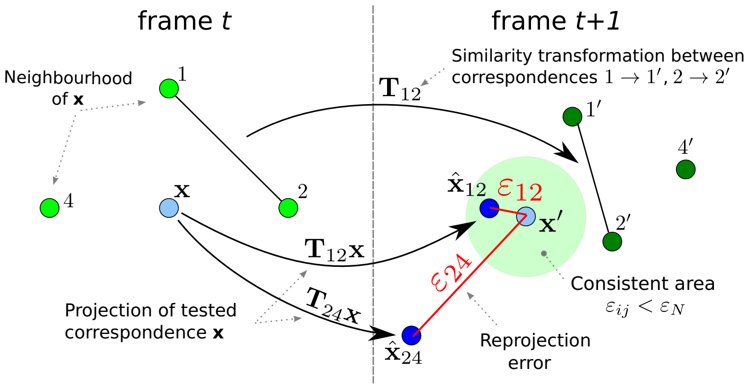


Fig. 5 Neighbourhood score computation for two pairs of correspondences. Each unique pair of correspondences (green) $i, j \in 1,2,3,4$ generate a similarity transformation $\mathbf{T}_{ij}$. The tested (blue) correspondence $\mathbf{x}$ is transform by the estimated similarities and the reprojection error $\varepsilon_{ij} = \| \hat{\mathbf{x}}_{ij} - \mathbf{x}' \|^2$ is computed. The final score is the number of $\varepsilon_{ij} < varepsilon_N$ (number of $\hat{\mathbf{x}}_{ij}$ points inside green circle around $\mathbf{x}'$).

$$\begin{aligned} &\mathrm{S'}_i^N = \frac{1}{\mathrm{Z}} \sum_{j \in N_i} \left[|\angle_{ij}| < \varepsilon_\angle \;\;\&\;\; \frac{\|\Delta_j\|}{\|\Delta_i\|} \in (\varepsilon_l, \varepsilon_h) \right] \\ &\text{where} \quad [expression] = \begin{cases} 1 & \text{if } expression \text{ is true} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

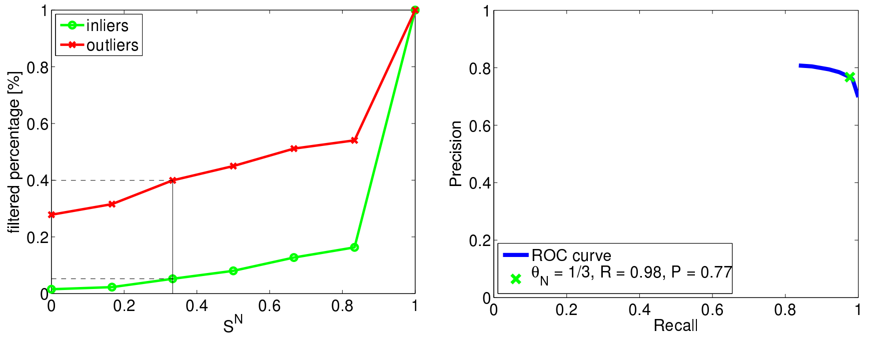
and where $\varepsilon_\angle$ is the maximum angle threshold, $(\varepsilon_l, \varepsilon_h)$ bounding range for the ratio of displacement magnitudes, Δ_i is the displacement of local tracker i and $\mathrm{Z} = \frac{4}{N_i}$ is normalization to 4-neighbourhood (to account for corners and sides of bounding box). A local tracker is defined to be consistent if $\mathrm{S}_i^N \geq \theta$, where θ is a threshold for this predictor.

$$\begin{aligned} &\mathrm{S}_i^N = \frac{1}{\mathrm{Z}} \sum_{\substack{j,k \in N_i \\ j \neq k}} \left[\| T_{jk}\mathbf{x}_i - \mathbf{x'}_i \|^2 < \varepsilon_N \right] \\ &\text{where} \quad [expression] = \begin{cases} 1 & \text{if } expression \text{ is true} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3}$$

Scoring function S_i^N counts the number of triplets of consistent local tracker. The transformation T_{jk} calculated from motion estimates of trackers j and k is applied on the reference point $\mathbf{x}$ of tracker i. If the transformed position $T_{jk}\mathbf{x}_i$ is within ε_N of its corresponding point $\mathbf{x'}_i$, one is added to the score. In experiments, ε_N was set to 2.

When used as a decision function which is required in one of the predictor combination methods described in the next section, there are finite number of possible thresholds depending on the number of neighbourhood local trackers.

Figure 6(a) shows a normalized cumulative histogram of the local tracker state for values of S^N normalized to range $< 0, 1 >$. Threshold $\theta_N = 1/6$ is chosen (i.e.



(a) (b)

Fig. 6 Properties of the P_N predictor averaged over a subset of test sequences and all frames, (a) The normalized cumulative histogram of the local tracker state for S^N, (b) The Precision-Recall curve for P_N predictor

S^N greater or equal to $1/3$ to predict an inlier state) as a good trade off between the ratio of filtered outliers and the false negative rate. Figure 6(b) shows the operating point of this threshold on the Precision-Recall curve.

3.4 The Markov Reliability Predictor P_M

The Markov reliability predictor (P_M) is based on the model of the past performance of a local tracker bound to a region specified by object coordinate frame. The model is in the form of a Markov chain with two states, $s_t \in \{0,1\}$, depicted in Fig. 7.

The predicted state (i.e. being correct - inlier or incorrect - outlier) of the local tracker depends on its state in the previous time instance and on the transition probabilities. The behaviour of each local tracker i at time t is modeled by transition matrix $\mathbf{T}_t^i$ described in Eq. 4, where s_t is the current state of the local tracker and whose columns sum to 1.

$$\mathbf{T}_t^i = \begin{bmatrix} p^i(s_{t+1} = 1 \mid s_t = 1) & p^i(s_{t+1} = 1 \mid s_t = 0) \\ p^i(s_{t+1} = 0 \mid s_t = 1) & p^i(s_{t+1} = 0 \mid s_t = 0) \end{bmatrix} \tag{4}$$

The prediction that certain local tracker would be an tentative inlier (or an outlier) is done according to equation 5.

$$\begin{bmatrix} p^i(s_{t+1} = 1) \\ p^i(s_{t+1} = 0) \end{bmatrix} = \mathbf{T}_t^i \cdot \begin{bmatrix} p^i(s_t = 1) \\ p^i(s_t = 0) \end{bmatrix} \tag{5}$$

where $p^i(s_t = 1) \in \{0,1\}$ is binary and depends on the previous state (inlier/outlier) of the i^{th} local tracker. The left side of equation 5 are then probabilities that next state would be inlier (outlier).

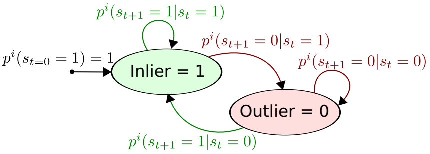

$p^i(s_{t+1} = 1|s_t = 1)$
$p^i(s_{t+1} = 0|s_t = 1)$
$p^i(s_{t=0} = 1) = 1$
Inlier = 1
$p^i(s_{t+1} = 0|s_t = 0)$
Outlier = 0
$p^i(s_{t+1} = 1|s_t = 0)$

Fig. 7 The state diagram of the Markov chain for the local tracker in the form of a two-state probabilistic automaton with transition probabilities p^i, where i identifies the local tracker and initial state $s_{t=0} = 1$.

In the update stage, transition probabilities are re-estimated as follows :

$$\begin{aligned} p^i(s_{t+1} = 1 \mid s_t = 1) &= \frac{n^i_{11}}{n^i_1} \\ p^i(s_{t+1} = 1 \mid s_t = 0) &= \frac{n^i_{01}}{n^i_0} \end{aligned} \tag{6}$$

where n_1 and n_0 are numbers for the local tracker i being inlier (outlier respectively), and n_{11} and n_{01} are relative frequency for event that the local tracker i was inlier (outlier respectively) in the time t and inlier in the time $t+1$, for $t \in (0,t\rangle$. The current state of the local tracker being inlier (outlier) is obtained by identifying local trackers that support the estimated global motion model.

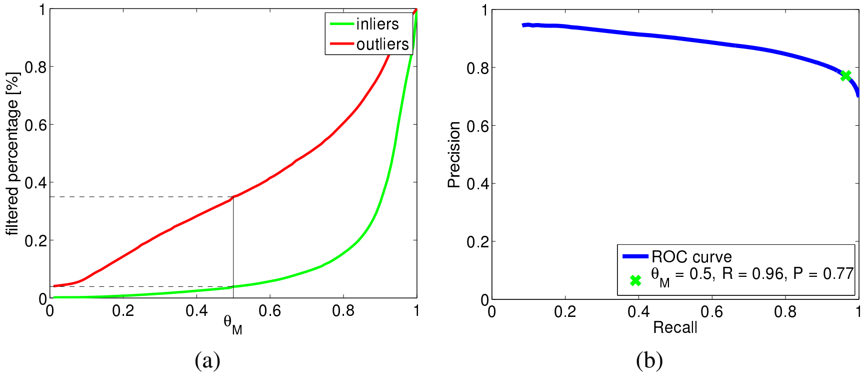


Fig. 8 Properties of the P_M predictor averaged over a subset of test sequences and all frames, (a) The normalized cumulative histograms of a the local tracker state for $p(s_{t+1} = 1)$ values quantized to 100 bins, (b) The Precision-Recall curve for the P_M predictor

When used as a decision function which is required in one of the predictor combination methods described in the next section, the observed characteristics support the natural choice of tresholding the inlier probability at 0.5. Figure 8(a) depicts the normalized cumulative histograms of a local tracker state for the Markov predictor values quantized to 100 bins. It shows how many inliers/outliers would be filtered out for different values of the θ_M threshold. The selected threshold 0.5 filtered out 4% of inliers and more than 35% of outliers. Figure 8(b) shows the operating point for threshold 0.5 on the Precision-Recall curve.

4 Methods for Combining Tracker Reliability Predictions

This section describes two predictor combination methods – $\mathscr{P}_\Theta$ and $\mathscr{P}_\wedge$ and discusses their advantages and disadvantages. The explanation of the combination methods is elaborate for the combination of three predictors P_ρ, P_N, P_M.

4.1 The $\mathscr{P}_\Theta$ Combination Method

The $\mathscr{P}_\Theta$ combination estimates the likelihood of a local tracker being an inlier. The local tracker inlier likelihood is a function of three variables (i) P_ρ rank $\in \{1,2,\ldots,100\}$ quantized equally to 25 bins, $\rho = \lceil \frac{\text{rank}}{25} \rceil$ (ii) The P_N score $\in \{0,1,2,3,4\}$ in case of four-neighbourhood (iii) P_M probability $\in (0,1)$ quantized equally to 25 bins. In the training phase a inlier/outlier likelihood ratio is estimated for all the combinations of variables using a Bayesian approach. resulting in a table with dimensions $25 \times 5 \times 25$. The combination can work in two modes (1) choose the fix threshold for local trackers inlier/outlier likelihood (2) take the *n* best local trackers, to form a local trackers subset for object pose estimation.

The advantage of this combination is a possibility to take an quasi-optimal decision (assuming independence of the individual predictors). The problem is formulated as a hypothesis test whether a local tracker is an inlier (outlier) given the likelihood ratio using a standard criterion such as NeymanPearson or min-max. The disadvantage is the need of the learning phase to the estimate local tracker inlier likelihood, which may overfit to the training data. In practice, the likelihood estimate generalized well enough to work in various scenarios.

4.2 The $\mathscr{P}_\wedge$ Combination Method

The $\mathscr{P}_\wedge$ predictor combination method computes responses of its constituent predictors and makes a binary decision for each of them (reliability below a threshold is interpreted as an outlier and visa versa). The final decision about the local tracker failure is a logical "and" function:

$$\begin{aligned} f(\mathrm{P}_\rho,\mathrm{P}_N,\mathrm{P}_M) = {} & \rho > \mathrm{median}(\rho) \\ & \wedge\ \mathrm{S}^N > \theta_N \\ & \wedge\ p(s_{t+1} = 1) > \theta_M \end{aligned} \tag{7}$$

The $\mathscr{P}_\wedge$ combination method assumes that since local tracker predictors exploit complementary information (i.e. P_ρ predictor – local appearance, P_M – temporal behaviour, P_N predictor – spatial consistency), parameters and threshold values of the inlier/outlier decision may be set independently.

5 RANSAC

The median estimator is robust and has a breakdown point 0.5. However, as shown in the experimental section, the percentage of correct local motion estimates is lower in many situations. Moreover, the median is biased if the noise is biased, which causes drifting of the tracker. This drifting happens in cases, where the background is static or locally static around the object of interest, e.g. when the bounding box is not a precise representation of the object shape and some local trackers are placed on the background.

We propose to use RANSAC for transformation estimation and show experimentally its superiority. This method has two main advantages over the median: (1) Is more robust to outliers (2) using unbiased least-square fitting to estimate transformation (up to homography).

6 Performance Evaluation

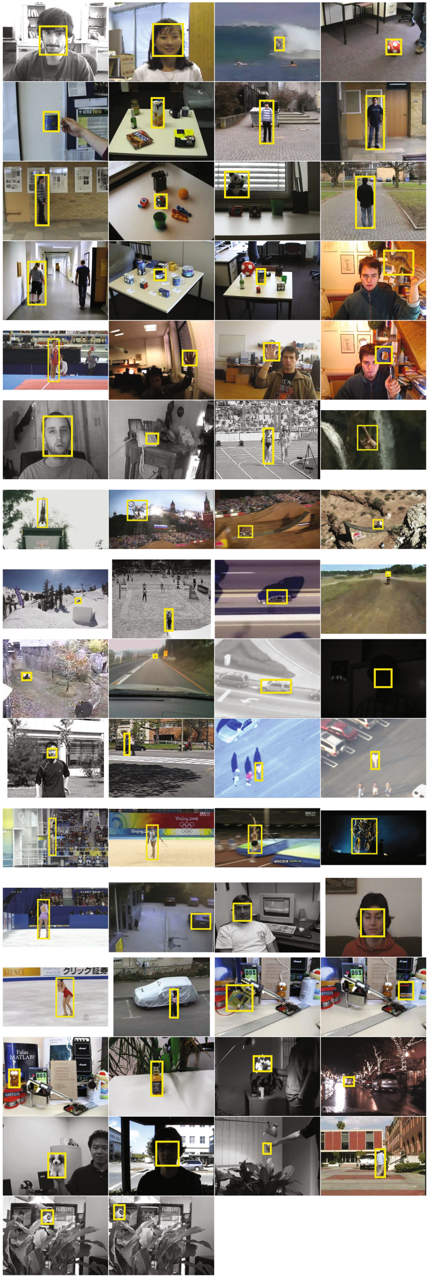
6.1 The Test Data

The performance of the FoT with combined reliability prediction of local trackers and RANSAC-based object motion estimation was tested on challenging video sequences collected from a number of recently published papers. The sequences include object occlusion (or disappearance), illumination changes, fast motion, different object sizes and object appearance variance. The videos vary in length, contain highly articulated object and background clutter; some have poor visual quality. Targets in the sequences exhibit out-of-plane and in-plane rotation and some have homogeneous surfaces almost without texture. The sequences are described in Tab. 1. For details about the sequences visit `http://cmp.felk.cvut.cz/~vojirtom/dataset`. The lists of authors who kindly provided the sequences is available on the web site.

6.2 The Experimental Set-Up

In all experiments, a frame was considered correctly tracked if the overlap with the ground truth is greater than 0.5, with the exception of experiment 6.6 where the influence of the initialization of the tracker was assessed. Since in this case the bounding boxes are randomly generated and may not fully overlap the object, the threshold was lower to 0.3, see Fig. 12. The overlap was measured as $o = \frac{area(T \cap G)}{area(T \cup G)}$, where T is object bounding box reported by the tracker and G is ground truth bounding box.

Table 1 Overview of the test sequences. Basic information (left) and sample images with the selected object of interest (right) are shown. Full information about the sequences (authors, papers reporting results on the data, etc.) and the data are available at http://cmp.felk.cvut.cz/∼vojirtom/dataset.

Seq. ID	name	#frames	#target visible	preview
1	OccludedFace2	815	815	
2	girl	501	475	
3	surfer	842	762	
4	Vid_A	602	602	
5	Vid_B	629	629	
6	Vid_C	404	404	
7	Vid_D	947	947	
8	Vid_E	305	305	
9	Vid_F	453	416	
10	Vid_G	716	716	
11	Vid_H	412	412	
12	Vid_I	1017	994	
13	Vid_J	388	383	
14	Vid_K	1020	1020	
15	Vid_L	1308	1308	
16	dinosaur	326	326	
17	gymnastics	567	567	
18	hand	244	244	
19	hand2	267	267	
20	torus	264	264	
21	head_motion	2351	2351	
22	shaking_camera	990	990	
23	track_running	503	397	
24	cliff-dive1	76	76	
25	cliff-dive2	69	61	
26	motocross1	164	164	
27	motocross2	23	23	
28	mountain-bike	228	228	
29	skiing	81	81	
30	volleyball	500	500	
31	CarChase	9928	8660	
32	Motocross	2665	1412	
33	Panda	3000	2730	
34	Volkswagen	8576	5141	
35	car	945	860	
36	david	761	761	
37	jumping	313	313	
38	pedestrian3	140	140	
39	pedestrian4	338	266	
40	pedestrian5	184	156	
41	diving	231	218	
42	gym	767	767	
43	jump	122	111	
44	trans	124	124	
45	Asada	661	661	
46	drunk2	1821	911	
47	dudek-face	1145	1145	
48	faceocc1	899	899	
49	figure_skating	624	624	
50	woman	597	597	
51	board	698	698	
52	box	1161	1129	
53	lemming	1336	1305	
54	liquor	1741	1704	
55	Sylvestr	1344	1344	
56	car11	393	393	
57	dog1	1353	1350	
58	trellis	569	569	
59	coke	292	270	
60	person	331	326	
61	tiger1	354	354	
62	tiger2	365	365	

In the experiments, the predictor of neighbourhood consistency (P_N) and the Markov predictor (P_M) were run as explained in Section 3. The normalized cross-correlation (P_ρ) and the forward-backward procedure rank local trackers and treat the top 50% as inliers. Combinations of two or more predictors use the $\mathscr{P}_\wedge$ approach. Predictors are denoted by the names of their error measure, except for the combination $P_M + P_\rho + P_N$ which is abbreviated to Σ.

6.3 Comparison of $\mathscr{P}_\wedge$ Combination vs. $\mathscr{P}_\Theta$ Combination

The $\mathscr{P}_\wedge$ predictor combination is compared with the $\mathscr{P}_\Theta$ combination in terms of inlier prediction precision. To make results comparable the measurement was done at the operating point of $\mathscr{P}_\wedge$ combination, since this method does not guarantee a number of predicted inliers and does not have any means for choosing n-best in contrast to $\mathscr{P}_\Theta$ combination.

The $\mathscr{P}_\Theta$ combination needs to learn likelihoods for the combined likelihood table of three criterion variables. A leave one out cross-validation was used to split the dataset to the training and validation sets. That means that for evaluation on sequence i the table is learned on all sequences except the sequence i. True inliers were extracted by comparing frame-to-frame tracking results with corresponding ground truth positions and criteria variables were recorded. The recorded values (P_N Score, P_M probability, P_ρ rank) were quantized (to 5, 25, 25 bins) and used to compute the inlier - outlier likelihood. Entries of the combined likelihood table are addressed by the quantized criteria values.

Table 2 The comparison of the $\mathscr{P}_\wedge$ predictor combination and the $\mathscr{P}_\Theta$ combination in terms of inlier prediction precision ± variation. Averaged performance over a subset of sequences is reported in the last row. The subset of sequences was selected such that it includes mainly rigid objects; in some sequences also articulated objects (pedestrians) are tracked.

Seq.	Θ	$\wedge$
17	0.713±0.132	0.738±0.134
20	0.875±0.022	0.919±0.021
31	0.894±0.040	0.922±0.043
32	0.857±0.060	0.895±0.058
33	0.952±0.029	0.773±0.166
34	0.943±0.007	0.965±0.005
35	0.958±0.008	0.977±0.008
36	0.945±0.006	0.966±0.004
37	0.680±0.073	0.730±0.068
38	0.623±0.053	0.684±0.060
39	0.925±0.013	0.945±0.026
40	0.967±0.002	0.986±0.001
55	0.980±0.006	0.986±0.006
59	0.924±0.008	0.967±0.006
Mean	0.874±0.033	0.890±0.043

Results in table 2 show that the two combination methods perform similarly. The $\mathscr{P}_{\wedge}$ predictor combination has an advantage that it does not require learning in advance. We choose to use the $\mathscr{P}_{\wedge}$ predictor combination to keep the tracker as independent as possible of the training data and other external variables (e.g. the precision of the ground truth used for extracting true inliers, the size of the dataset, diversity of dataset, etc.).

6.4 Comparison of the Reliability Prediction Methods

We compared performance of individual predictors and combinations $\mathrm{P}_{FB\circ\rho}$ (reference [7]), $\mathrm{P}_{N\circ M}$ and P_{Σ}. All parameters for predictors were fixed for all sequences, as described in Section 4.2.

The performance was measured by the recall and the number of reinitialization needed to track the whole sequences (reinitialization after object disappearance are not counted). The recall is defined as the ratio of the number of frame where the estimated object rectangle had an overlap with the ground truth rectangle higher then 0.5 and the number of frames where the object is visible. Approximately speaking, recall is the percentage of the frames with the tracked object visible where the object was correctly tracked.

The results are summarized in tables 3 and 4. Both tables have the same structure. Each line starting with a number presents results on one of the 62 sequence. The last two lines summarize performance. The #best line compares the median flow object motion estimator (m, left) and the RANSAC-based estimator (r, right) by counting the number of sequences when median flow outperformed RANSAC (the number before the ":"), where RANSAC dominated (the number after the ":"), the number of "draws" is given in parentheses.

According to both the recall (table 3) and reinitialization (table 4) criteria, RANSAC performs better for all reliability predictors and their combinations. Results for different predictors and combinations are presented in different columns. The final line of the table compares the mean recall and reinitialization. RANSAC performs better in terms of the mean too.

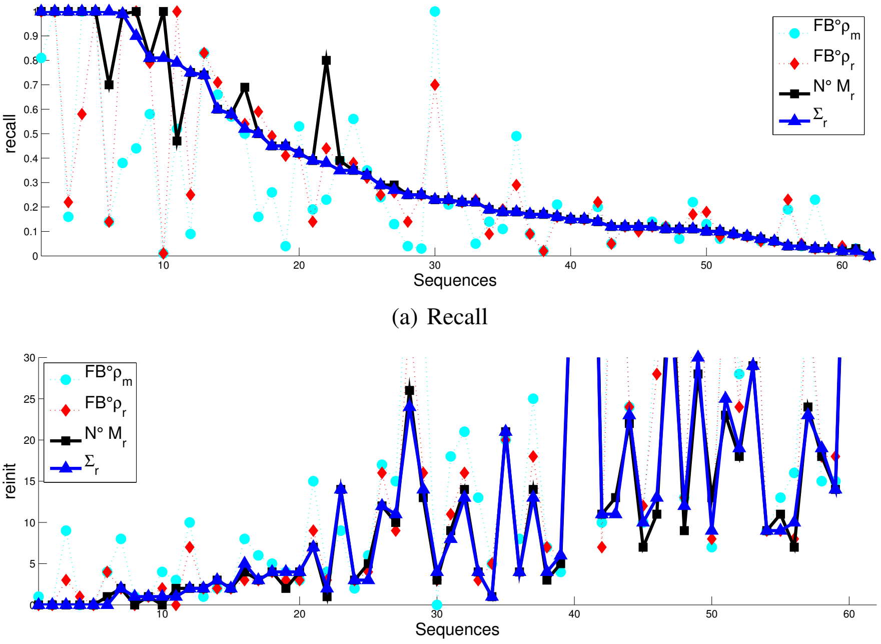
The "mean" row allows comparison of the the reliability predictors, both individually and in combination. The combinations $\mathrm{P}_{N\circ M}$ and P_{Σ} perform the best, clearly better than any individual tracker and slightly better than the forward-backward procedure combined with the NCC. Note that the P_{Σ} and even more $\mathrm{P}_{N\circ M}$ are significantly faster than the FB procedure.

Fig. 9 visualizes the performance for selected combinations of predictors in a manner facilitating comparison. Two combinations of predictors P_{Σ} and $\mathrm{P}_{N\circ M}$ are clear the most reliable methods.

Visualization of predictor performance on selected frames from two challanging sequences are shown in Figs. 10 (*motor-bike*) and 11 *woman*. Predictor score is encoded in a "heat map" (red - high score, blue - low score). Green/Red boxes below predictor score encodes false positive (red dot with red background), false negative (green dot with red background), true positive (green dot with green background) and true negative (red dot with green background). On the right side of the image, a cut out shows the outlier (red) and inlier (green) motion estimates. The green-on-black images shows the area covered by inlier local trackers.

For the *motor-bike* sequence, it is somewhat surprising that the motion estimates on the biker are small. The biker is tracked by the cameraman and the position of the bike in the image stays roughly the same, the background exhibits fast apparent motion in the oposite direction. The FoT handle are rather large change of apperance of the biker between frames #31 and #77.

The *woman* sequence is more challenging, due to occlusion and changes of appearance due to walking, the number of local trackers providing correct motion estimates is small, as low as 19 out of 90 in frame # 18.



(a) Recall

(b) Reinit count

Fig. 9 Comparison of the best performing predictor combinations and estimators in terms (a) Recall and (b) the number of reinitialization. Sequences (x-axis) are sorted by the recall measure of the P_{Σ} with RANSAC estimator.

6.5 Comparison the Speed of the Reliability Prediction Methods

The FoT tracker is intended for real-time performance and thus the speed of local tracker predictor is important. The experiment was performed on all sequences listed in Tab. 1 and then the results were averaged. Speed was measured as the average time needed for frame-to-frame tracking. For results see Tab. 5. Processing

Table 3 The recall of the FoT on 62 sequences. For details, see text.

P	$\emptyset$	ρ	N	FB	M	FB ∘ ρ	$N \circ M$	Σ
Seq.	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r
1	0.13 ◀ 0.18	0.12 ◀ 0.18	0.12 ◀ 0.18	0.11 ◀ 0.19	0.12 ◀ 0.18	0.11 ◀ 0.19	0.13 ◀ 0.18	0.13 ◀ 0.18
2	1.00 ▶ 0.40	1.00 ▶ 0.47	0.47 = 0.47	1.00 ▶ 0.70	0.47 ▶ 0.23	1.00 ▶ 0.70	0.22 ◀ 0.23	0.22 ◀ 0.23
3	0.02 ◀ 0.06	0.02 ◀ 0.06	0.07 = 0.07	0.06 = 0.06	0.07 ▶ 0.06	0.06 = 0.06	0.07 ▶ 0.06	0.07 ▶ 0.06
4	0.11 = 0.11	0.11 = 0.11	0.11 = 0.11	0.12 = 0.12	0.11 = 0.11	0.12 = 0.12	0.13 ▶ 0.11	0.12 ▶ 0.11
5	0.22 ◀ 0.38	0.24 ◀ 0.35	0.35 ◀ 0.38	0.23 ◀ 0.44	0.47 ◀ 1.00	0.23 ◀ 0.44	0.38 ◀ 0.80	0.38 = 0.38
6	0.50 ◀ 1.00	0.51 ◀ 1.00	0.47 ◀ 1.00	0.44 ◀ 1.00	0.48 ◀ 1.00	0.44 ◀ 1.00	0.47 ◀ 1.00	0.46 ◀ 0.90
7	0.57 ▶ 0.39	0.57 ▶ 0.39	0.57 ▶ 0.35	0.39 ▶ 0.38	0.58 ▶ 0.39	0.39 ▶ 0.38	0.58 ▶ 0.39	0.54 ▶ 0.35
8	0.57 ◀ 0.58	0.57 ◀ 0.58	0.57 ◀ 0.58	0.57 ◀ 0.58	0.57 = 0.57	0.57 ◀ 0.58	0.57 ◀ 0.58	0.57 ◀ 0.58
9	0.23 ◀ 0.32	0.23 ◀ 0.29	0.28 ◀ 0.29	0.24 ◀ 0.25	0.36 ▶ 0.28	0.24 ◀ 0.25	0.28 ◀ 0.29	0.36 ▶ 0.29
10	0.83 ◀ 1.00	0.83 ◀ 1.00	0.84 ◀ 1.00	0.81 ◀ 1.00	0.84 ◀ 1.00	0.81 ◀ 1.00	0.82 ◀ 1.00	0.83 ◀ 1.00
11	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00
12	0.09 ◀ 0.12	0.10 ◀ 0.11	0.09 ◀ 0.11	0.07 ◀ 0.11	0.08 ◀ 0.11	0.07 ◀ 0.11	0.08 ◀ 0.11	0.08 ◀ 0.11
13	0.20 ▶ 0.17	0.20 ▶ 0.17	0.20 ▶ 0.17	0.21 ▶ 0.16	0.27 ▶ 0.16	0.21 ▶ 0.16	0.31 ▶ 0.16	0.31 ▶ 0.16
14	0.64 ▶ 0.42	0.64 ▶ 0.54	0.52 ◀ 0.97	0.52 ◀ 1.00	0.64 ▶ 0.43	0.52 ◀ 1.00	0.52 ▶ 0.47	0.52 ◀ 0.79
15	0.16 ◀ 0.78	0.16 ◀ 0.74	0.16 ◀ 0.74	0.16 ◀ 0.59	0.16 ◀ 0.56	0.16 ◀ 0.59	0.16 ◀ 0.50	0.16 ◀ 0.50
16	0.25 ◀ 0.39	0.25 ◀ 0.39	0.25 ◀ 0.38	0.19 ▶ 0.14	0.27 ◀ 0.39	0.19 ▶ 0.14	0.39 = 0.39	0.39 = 0.39
17	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.14 ◀ 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15
18	0.09 ◀ 0.16	0.09 ◀ 0.15	0.11 ▶ 0.09	0.09 = 0.09	0.13 ▶ 0.09	0.09 = 0.09	0.16 ◀ 0.17	0.16 ◀ 0.17
19	0.09 ◀ 0.22	0.09 ◀ 0.14	0.07 ◀ 0.26	0.04 ◀ 0.14	0.05 ◀ 0.14	0.04 ◀ 0.14	0.05 ◀ 0.25	0.05 ◀ 0.25
20	0.20 ◀ 0.52	0.20 ◀ 0.56	0.21 ◀ 0.60	0.16 ◀ 0.22	0.46 ◀ 0.58	0.16 ◀ 0.22	0.54 ◀ 1.00	0.54 ◀ 1.00
21	0.77 ◀ 0.80	0.76 ▶ 0.52	0.77 ◀ 0.80	0.58 ◀ 0.79	0.77 ◀ 0.81	0.58 ◀ 0.79	0.77 ◀ 0.81	0.77 ◀ 0.81
22	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15	0.15 = 0.15
23	0.09 ◀ 0.21	0.10 ◀ 0.21	0.09 ◀ 0.21	0.20 ◀ 0.22	0.13 ◀ 0.82	0.20 ◀ 0.22	0.13 ◀ 0.14	0.13 ◀ 0.14
24	0.34 ◀ 0.42	0.34 ◀ 0.41	0.34 ◀ 0.41	0.53 ▶ 0.42	0.42 ▶ 0.41	0.53 ▶ 0.42	0.43 ▶ 0.42	0.43 ▶ 0.42
25	0.15 ▶ 0.13	0.16 ▶ 0.11	0.15 ▶ 0.11	0.13 ◀ 0.18	0.11 ◀ 0.13	0.13 ◀ 0.18	0.15 ▶ 0.10	0.15 ▶ 0.10
26	0.18 ▶ 0.04	0.18 ▶ 0.03	0.45 ▶ 0.04	0.23 ▶ 0.03	0.16 ▶ 0.03	0.23 ▶ 0.03	0.05 ▶ 0.03	0.05 ▶ 0.03
27	0.83 ▶ 0.70	0.83 ▶ 0.70	0.83 ▶ 0.70	0.83 = 0.83	0.57 ◀ 0.91	0.83 = 0.83	0.57 ◀ 0.74	0.57 ◀ 0.74
28	0.40 ◀ 0.99	0.40 ◀ 0.99	0.43 ◀ 0.99	0.38 ◀ 0.99	0.82 ◀ 0.99	0.38 ◀ 0.99	0.82 ◀ 0.99	0.82 ◀ 0.99
29	0.07 ◀ 0.10	0.07 ◀ 0.10	0.07 ◀ 0.10	0.09 = 0.09	0.06 ◀ 0.07	0.09 = 0.09	0.06 ◀ 0.09	0.06 ◀ 0.09
30	0.23 ▶ 0.22	0.23 ▶ 0.22	0.22 = 0.22	0.22 = 0.22	0.22 = 0.22	0.22 = 0.22	0.22 = 0.22	0.22 = 0.22
31	0.50 ◀ 1.00	0.48 ◀ 0.58	1.00 ▶ 0.57	1.00 ▶ 0.58	0.75 ▶ 0.50	1.00 ▶ 0.58	0.61 ◀ 1.00	0.61 ◀ 1.00
32	0.01 ◀ 0.02	0.01 ◀ 0.02	0.02 = 0.02	0.03 ◀ 0.04	0.01 ◀ 0.02	0.03 ◀ 0.04	0.02 = 0.02	0.02 = 0.02
33	0.45 ◀ 0.60	0.59 ▶ 0.32	0.59 ▶ 0.50	0.01 = 0.01	0.39 ◀ 1.00	0.01 = 0.01	0.59 ◀ 1.00	0.59 ◀ 0.81
34	0.13 ◀ 0.24	0.14 ▶ 0.11	0.11 ◀ 0.24	0.05 = 0.05	0.14 ▶ 0.12	0.05 = 0.05	0.18 ▶ 0.12	0.18 ▶ 0.12
35	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00	0.00 = 0.00
36	0.02 = 0.02	0.02 = 0.02	0.02 = 0.02	0.03 ▶ 0.02	0.02 ◀ 0.03	0.03 ▶ 0.02	0.02 ◀ 0.03	0.02 = 0.02
37	0.06 ◀ 0.19	0.06 ◀ 0.09	0.07 ◀ 0.09	0.14 ▶ 0.09	0.11 ◀ 0.32	0.14 ▶ 0.09	0.04 ◀ 0.19	0.04 ◀ 0.19
38	0.58 ▶ 0.54	0.58 ▶ 0.53	0.50 ◀ 0.70	0.66 ◀ 0.71	1.00 ▶ 0.56	0.66 ◀ 0.71	1.00 ▶ 0.60	1.00 ▶ 0.60
39	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	1.00 = 1.00	0.92 ◀ 1.00	1.00 = 1.00	0.89 ◀ 1.00	0.89 ◀ 1.00
40	0.05 ▶ 0.04	0.05 = 0.05	0.18 ◀ 0.24	0.19 ◀ 0.23	0.05 = 0.05	0.19 ◀ 0.23	0.18 ▶ 0.04	0.19 ▶ 0.04
41	0.13 ▶ 0.12	0.13 ▶ 0.12	0.13 ▶ 0.12	0.12 = 0.12	0.17 ▶ 0.12	0.12 = 0.12	0.16 ▶ 0.12	0.16 ▶ 0.12
42	0.04 ▶ 0.03	0.07 ▶ 0.03	0.03 = 0.03	0.03 = 0.03	0.03 = 0.03	0.03 = 0.03	0.03 = 0.03	0.03 = 0.03
43	0.06 ◀ 0.09	0.06 ◀ 0.10	0.10 = 0.10	0.11 ▶ 0.10	0.15 ▶ 0.14	0.11 ▶ 0.10	0.14 ▶ 0.12	0.14 ▶ 0.12
44	0.51 ▶ 0.38	0.44 ▶ 0.39	0.41 ◀ 0.50	0.56 ▶ 0.38	0.35 ◀ 0.40	0.56 ▶ 0.38	0.35 = 0.35	0.35 = 0.35
45	0.08 = 0.08	0.08 ◀ 0.15	0.15 ▶ 0.09	0.08 = 0.08	0.07 ◀ 0.09	0.08 = 0.08	0.09 ▶ 0.08	0.09 ▶ 0.08
46	0.04 ◀ 0.20	0.04 ◀ 0.17	0.03 ◀ 0.19	0.02 = 0.02	0.01 ◀ 0.61	0.02 = 0.02	0.01 ◀ 0.17	0.01 ◀ 0.17
47	0.18 = 0.18	0.18 = 0.18	0.18 ◀ 0.29	0.49 ▶ 0.29	0.18 = 0.18	0.49 ▶ 0.29	0.18 = 0.18	0.18 = 0.18
48	0.10 ◀ 0.58	0.10 ◀ 0.69	0.10 ◀ 0.58	0.09 ◀ 0.25	0.07 ◀ 0.36	0.09 ◀ 0.25	0.07 ◀ 0.75	0.07 ◀ 0.75
49	0.05 = 0.05	0.05 ▶ 0.04	0.05 ▶ 0.03	0.04 ◀ 0.05	0.04 ◀ 0.05	0.04 ◀ 0.05	0.08 ▶ 0.04	0.08 ▶ 0.04
50	0.06 ◀ 0.12	0.07 ◀ 0.11	0.06 ◀ 0.12	0.14 ▶ 0.12	0.42 ▶ 0.12	0.14 ▶ 0.12	0.05 ◀ 0.12	0.05 ◀ 0.12
51	0.06 ◀ 0.21	0.06 ◀ 0.63	0.08 ◀ 0.56	0.05 ◀ 0.23	0.22 = 0.22	0.05 ◀ 0.23	0.48 ▶ 0.22	0.48 ▶ 0.22
52	0.05 ◀ 0.26	0.08 ◀ 0.24	0.09 ◀ 0.27	0.13 ◀ 0.26	0.05 ◀ 0.26	0.13 ◀ 0.26	0.10 ◀ 0.29	0.10 ◀ 0.27
53	0.02 ◀ 0.25	0.02 ◀ 0.25	0.03 ◀ 0.25	0.03 ◀ 0.25	0.09 ◀ 0.25	0.03 ◀ 0.25	0.09 ◀ 0.25	0.09 ◀ 0.25
54	0.21 ◀ 0.23	0.21 ◀ 0.23	0.21 ◀ 0.23	0.21 ◀ 0.23	0.23 = 0.23	0.21 ◀ 0.23	0.23 = 0.23	0.23 = 0.23
55	0.26 ◀ 0.43	0.26 ◀ 0.43	0.26 ◀ 0.40	0.26 ◀ 0.49	0.26 ◀ 0.40	0.26 ◀ 0.49	0.26 ◀ 0.45	0.26 ◀ 0.45
56	0.58 ▶ 0.52	0.58 ▶ 0.53	0.65 ▶ 0.54	0.50 ◀ 0.54	0.48 ◀ 0.73	0.50 ◀ 0.54	0.53 ◀ 0.69	0.53 ▶ 0.52
57	0.31 ◀ 0.33	0.32 ◀ 0.34	0.33 ▶ 0.32	0.35 ▶ 0.32	0.34 ▶ 0.32	0.35 ▶ 0.32	0.35 ▶ 0.33	0.35 ▶ 0.33
58	0.04 ◀ 0.67	0.04 ◀ 0.45	0.04 ◀ 0.45	0.04 ◀ 0.41	0.04 ◀ 0.44	0.04 ◀ 0.41	0.04 ◀ 0.45	0.04 ◀ 0.45
59	0.14 = 0.14	0.14 = 0.14	0.14 = 0.14	0.14 = 0.14	1.00 ▶ 0.14	0.14 = 0.14	1.00 ▶ 0.70	1.00 = 1.00
60	0.05 ◀ 0.06	0.05 ◀ 0.06	0.05 ◀ 0.06	0.06 = 0.06	0.11 ▶ 0.08	0.06 = 0.06	0.10 ▶ 0.07	0.10 ▶ 0.07
61	0.07 ◀ 0.08	0.07 ◀ 0.08	0.08 = 0.08	0.07 ◀ 0.08	0.11 ▶ 0.08	0.07 ◀ 0.08	0.11 ▶ 0.10	0.11 ▶ 0.10
62	0.11 ▶ 0.09	0.11 ▶ 0.09	0.16 ▶ 0.11	0.22 ▶ 0.17	0.11 = 0.11	0.22 ▶ 0.17	0.23 ▶ 0.11	0.23 ▶ 0.11
#best	15:36 (11)	18:34 (10)	14:33 (15)	15:28 (19)	19:31 (12)	15:28 (19)	21:30 (11)	21:27 (14)
mean	0.26:0.34	0.26:0.32	0.27:0.35	0.27:0.32	0.30:0.35	0.27:0.32	0.30:**0.36**	0.30:**0.36**

Table 4 The number of reinitialisations of the FoT on 62 sequences. For details, see text.

P	$\emptyset$	ρ	N	FB	M	FB $\circ\, \rho$	$N \circ M$	Σ
Seq.	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r
1	26 ◀ 21	24 ◀ 22	23 = 23	20 = 20	25 ◀ 18	20 = 20	24 ◀ 21	24 ◀ 21
2	0 ▶ 4	0 ▶ 3	2 ▶ 3	0 ▶ 3	3 ▶ 4	0 ▶ 3	2 ▶ 3	4 = 4
3	21 ◀ 16	20 ◀ 12	15 ◀ 11	13 ◀ 9	14 = 14	13 ◀ 9	17 ◀ 11	17 ◀ 9
4	45 ▶ 50	48 ◀ 44	46 ◀ 44	40 ▶ 48	28 ▶ 39	40 ▶ 48	28 ▶ 42	25 ▶ 37
5	9 ◀ 2	7 ◀ 1	3 ◀ 2	4 ◀ 3	2 ◀ 0	4 ◀ 3	2 ◀ 1	2 = 2
6	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 = 1
7	10 ▶ 14	10 ▶ 15	10 ▶ 14	9 ▶ 14	9 ▶ 14	9 ▶ 14	9 ▶ 14	9 ▶ 14
8	2 = 2	2 = 2	2 = 2	2 = 2	2 = 2	2 = 2	2 = 2	2 = 2
9	13 ▶ 15	14 ▶ 15	18 ◀ 16	17 ◀ 16	7 ▶ 13	17 ◀ 16	9 ▶ 12	7 ▶ 12
10	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0	1 ◀ 0
11	0 = 0	0 = 0	0 = 0	0 = 0	0 = 0	0 = 0	0 = 0	0 = 0
12	23 ◀ 13	22 ◀ 15	18 ◀ 11	13 = 13	19 ◀ 10	13 = 13	13 ◀ 9	13 ◀ 12
13	5 = 5	5 = 5	6 ◀ 5	4 ▶ 6	4 ▶ 5	4 ▶ 6	4 ▶ 5	4 ▶ 6
14	2 = 2	2 ◀ 1	2 ◀ 1	3 ◀ 0	2 = 2	3 ◀ 0	2 = 2	2 ◀ 1
15	5 ◀ 1	5 ◀ 2	5 ◀ 2	6 ◀ 3	6 ◀ 3	6 ◀ 3	7 ◀ 3	7 ◀ 3
16	10 ◀ 8	9 = 9	9 ▶ 10	15 ◀ 9	5 ▶ 7	15 ◀ 9	5 ▶ 7	5 ▶ 7
17	52 ▶ 53	52 ▶ 57	51 ▶ 54	57 ◀ 56	49 ▶ 55	57 ◀ 56	49 ▶ 51	49 ▶ 51
18	21 ◀ 14	20 ◀ 12	16 ◀ 13	25 ◀ 18	17 ◀ 11	25 ◀ 18	14 = 14	14 ◀ 13
19	35 ◀ 26	33 ◀ 26	29 ◀ 26	46 ◀ 35	44 ◀ 21	46 ◀ 35	29 ◀ 26	29 ◀ 24
20	8 ◀ 2	8 ◀ 2	6 ◀ 2	9 ◀ 3	3 ◀ 2	9 ◀ 3	2 ◀ 0	2 ◀ 0
21	2 ◀ 1	2 = 2	2 ◀ 1	1 = 1	2 ◀ 1	1 = 1	2 ◀ 1	2 ◀ 1
22	95 ◀ 91	95 ◀ 91	90 ◀ 89	91 ▶ 95	93 = 93	91 ▶ 95	90 ▶ 91	90 ◀ 89
23	11 ◀ 4	9 ◀ 4	5 ◀ 2	10 ◀ 7	10 ◀ 1	10 ◀ 7	12 ◀ 11	12 ◀ 11
24	4 = 4	5 ◀ 4	5 ◀ 4	3 = 3	4 = 4	3 = 3	3 ▶ 4	3 ▶ 4
25	36 ◀ 26	35 ◀ 22	17 ▶ 19	7 ▶ 8	19 ◀ 17	7 ▶ 8	6 ▶ 13	6 ▶ 9
26	14 ▶ 17	11 ▶ 16	6 ▶ 19	15 ▶ 18	13 = 13	15 ▶ 18	15 ▶ 18	15 ▶ 19
27	2 = 2	2 = 2	1 ▶ 2	1 ▶ 2	3 ◀ 1	1 ▶ 2	1 ▶ 2	1 ▶ 2
28	6 ◀ 2	6 ◀ 2	5 ◀ 2	8 ◀ 2	3 ◀ 2	8 ◀ 2	4 ◀ 2	3 ◀ 2
29	22 ◀ 18	23 ◀ 16	19 = 19	28 ◀ 24	24 ◀ 18	28 ◀ 24	21 ◀ 18	21 ◀ 19
30	14 ▶ 15	10 ▶ 16	13 ▶ 14	21 ◀ 16	11 = 11	21 ◀ 16	6 ▶ 14	5 ▶ 13
31	1 ◀ 0	1 = 1	0 ▶ 1	0 ▶ 1	2 ◀ 1	0 ▶ 1	2 ◀ 0	2 ◀ 0
32	220 ◀ 83	210 ◀ 79	110 ◀ 76	77 ◀ 70	193 ◀ 80	77 ◀ 71	107 ◀ 65	103 ◀ 69
33	5 ◀ 1	3 ◀ 1	2 ◀ 1	4 ◀ 2	6 ◀ 0	4 ◀ 2	2 ◀ 0	2 ◀ 1
34	12 ◀ 9	12 ▶ 13	9 ▶ 10	68 ◀ 63	9 ▶ 10	68 ◀ 63	10 ▶ 13	11 = 11
35	59 ◀ 27	56 ◀ 29	45 ◀ 33	56 ◀ 50	66 ◀ 32	56 ◀ 50	55 ◀ 36	55 ◀ 34
36	67 ▶ 76	68 ▶ 77	69 ▶ 79	66 ▶ 78	68 ▶ 76	66 ▶ 78	63 ▶ 73	63 ▶ 76
37	13 ◀ 2	13 ◀ 3	8 ◀ 2	5 = 5	10 ◀ 1	5 = 5	10 ◀ 1	8 ◀ 1
38	3 ▶ 4	3 ▶ 4	4 ◀ 3	2 = 2	0 ▶ 4	2 = 2	0 ▶ 3	0 ▶ 3
39	0 = 0	0 = 0	0 = 0	0 = 0	1 ◀ 0	0 = 0	1 ◀ 0	1 ◀ 0
40	26 ◀ 10	23 ◀ 11	18 ◀ 9	16 ◀ 8	20 ◀ 10	16 ◀ 8	20 ◀ 7	15 ◀ 10
41	21 ▶ 22	21 = 21	22 = 22	24 = 24	18 ▶ 22	24 = 24	21 ▶ 22	21 ▶ 23
42	13 ▶ 17	14 = 14	13 ▶ 16	15 ▶ 18	9 ▶ 14	15 ▶ 18	12 ▶ 14	10 ▶ 14
43	10 ◀ 9	10 ▶ 11	9 ▶ 11	10 ▶ 12	8 ◀ 7	10 ▶ 12	7 = 7	7 ▶ 10
44	2 = 2	2 = 2	2 = 2	2 ▶ 3	3 ◀ 2	2 ▶ 3	3 = 3	3 = 3
45	53 ◀ 46	53 ◀ 48	42 ▶ 44	52 ◀ 50	33 ▶ 35	52 ◀ 50	32 ◀ 29	32 ◀ 29
46	7 ◀ 3	7 ◀ 3	5 ◀ 3	7 = 7	8 ◀ 3	7 = 7	6 ◀ 3	8 ◀ 4
47	7 ◀ 4	7 ◀ 3	6 ◀ 4	8 ◀ 4	8 ◀ 4	8 ◀ 4	7 ◀ 4	7 ◀ 4
48	3 ▶ 6	3 = 3	8 ◀ 7	10 ◀ 7	7 ▶ 8	10 ◀ 7	8 ◀ 2	8 ◀ 2
49	34 ▶ 37	35 ◀ 34	32 ▶ 37	37 ▶ 38	26 ◀ 22	37 ▶ 38	17 ▶ 24	17 ▶ 23
50	26 = 26	28 = 28	27 = 27	34 ◀ 28	5 ▶ 13	34 ◀ 28	8 ▶ 11	17 ◀ 13
51	8 ◀ 5	6 ◀ 5	6 ◀ 5	13 ◀ 3	10 ◀ 5	13 ◀ 3	12 ◀ 4	12 ◀ 4
52	15 ◀ 9	14 ◀ 11	10 = 10	15 ◀ 9	18 ◀ 9	15 ◀ 9	17 ◀ 10	18 ◀ 11
53	32 ◀ 9	34 ◀ 8	23 ◀ 9	37 ◀ 16	41 ◀ 11	37 ◀ 16	33 ◀ 13	33 ◀ 14
54	11 ◀ 5	11 ◀ 5	11 ◀ 5	18 ◀ 11	12 ◀ 5	18 ◀ 11	10 ◀ 9	10 ◀ 8
55	5 ◀ 4	5 ◀ 4	5 ◀ 4	5 ◀ 4	5 ◀ 4	5 ◀ 4	5 ◀ 4	5 ◀ 4
56	10 ◀ 4	10 ◀ 3	7 ◀ 3	8 ◀ 3	8 ◀ 4	8 ◀ 3	9 ◀ 4	9 ◀ 5
57	8 ◀ 6	7 ◀ 5	4 ◀ 3	6 ◀ 4	6 ◀ 3	6 ◀ 4	12 ◀ 5	10 ◀ 3
58	14 ◀ 1	13 ◀ 2	6 ◀ 3	4 ◀ 3	13 ◀ 2	4 ◀ 3	5 ◀ 2	7 ◀ 4
59	5 ◀ 4	4 = 4	3 ▶ 4	4 = 4	0 ▶ 4	4 = 4	0 ▶ 1	0 = 0
60	8 ◀ 7	8 = 8	8 ▶ 9	9 = 9	7 ▶ 8	9 = 9	6 ▶ 9	6 ▶ 9
61	34 ◀ 30	31 ▶ 32	20 ▶ 29	43 = 43	40 ◀ 21	43 = 43	31 ◀ 23	36 ◀ 25
62	37 ◀ 25	33 ◀ 24	19 ▶ 25	48 ◀ 46	32 ◀ 31	48 ◀ 46	28 = 28	28 ▶ 30
#best	13:40 (9)	11:36 (15)	19:34 (9)	14:34 (14)	17:37 (8)	14:34 (14)	22:33 (7)	19:35 (8)
mean	20.4:14.9	19.8:14.7	15.8:14.6	18.9:17.1	18.0:13.4	18.9:17.1	15.1:**13.3**	15.1:13.5

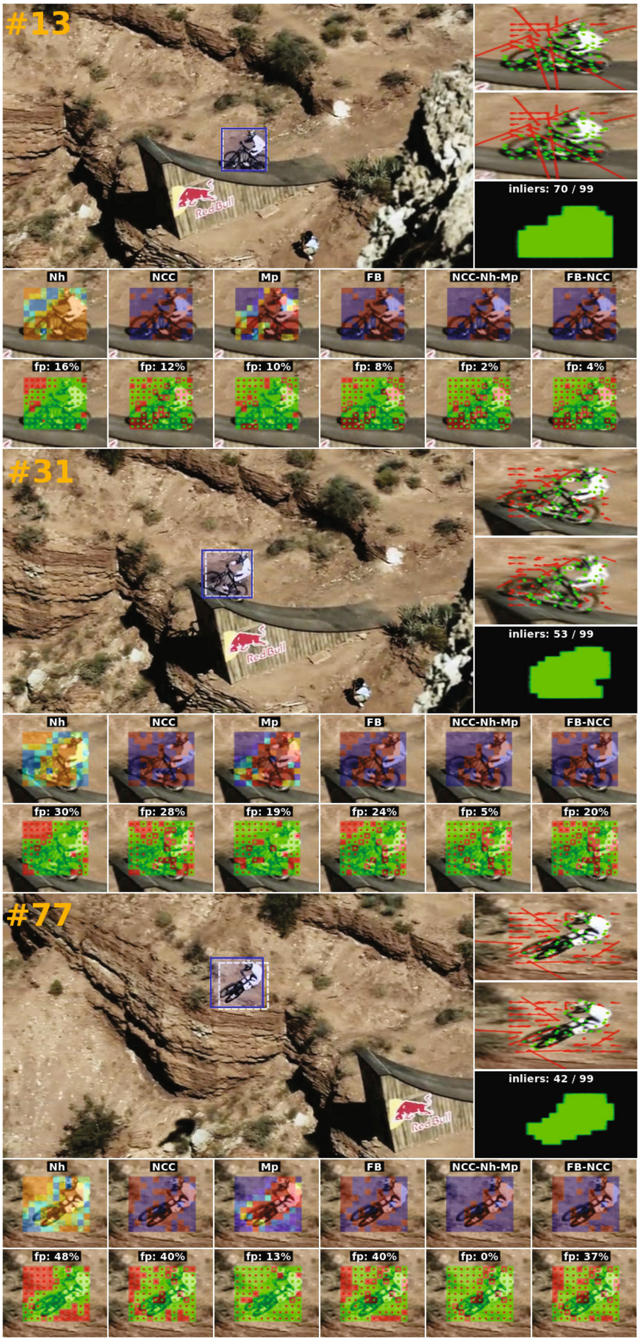


Fig. 10 Visualization of predictors performance on sequence *mountain-bike*. For details, see text.

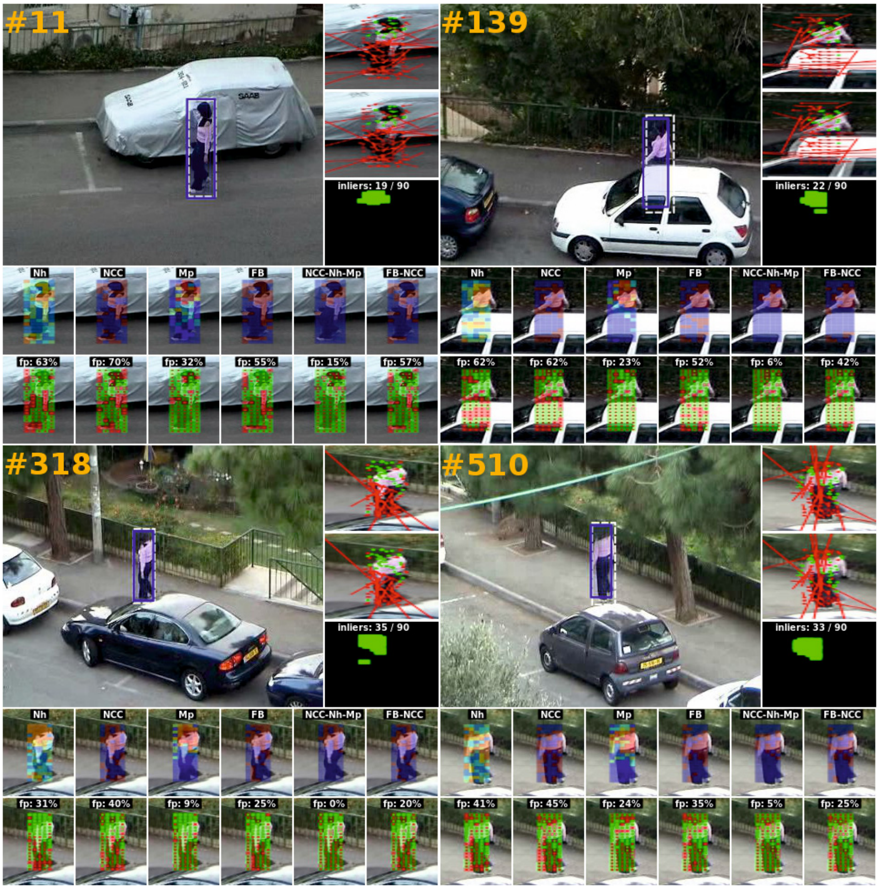


Fig. 11 Visualization of predictors performance on sequence *woman*. For details, see text.

time for I/O operations, including image loading, and other tasks not relevant to tracking were excluded. The P_{Σ} predictor performs 41% faster than $P_{FB\circ\rho}$. Most of the additional computation of P_{Σ} over the $P_{\emptyset}$ lies in computation of normalized cross-correlation. Therefore, the $P_{N\circ M}$ overhead is negligible compared to reference predictor $P_{\emptyset}$ (i.e. tracker without any predictor) and is more than two times faster then $P_{FB\circ\rho}$.

Table 5 A comparison of the speed of tracking reliability prediction methods. All times are in milliseconds. The values are averaged over all sequences.

P / Seq.	$\emptyset$	ρ	FB	$\mathrm{FB}\circ\rho$	$N\circ M$	Σ
	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r	m ◇ r
Time [ms]	1.53 ▶ 1.55	2.44 ▶ 2.87	2.52 ▶ 2.89	3.43 ▶ 3.58	1.58 ▶ 1.72	2.43 ▶ 2.52

6.6 Robustness to Bounding Box Initialization

For a tracking algorithm, it is highly desirable not to be sensitive to the initial pose specified by the object bounding box as it is often selected manually, with unknown precision.

If a part of the bounding box does not cover the object, the P_M predictor soon discover that the local trackers are consistently in the outlier set. This property can be used to define the object more precisely, e.g. as the set of object parts that are likely to be inliers according to P_M (see Figs. 10 and 11). Thus, with P_M, the global tracker may be insensitive to initialization.

This experiment tested the assumption on the challenging sequence Pedestrian 1, where an articulated object is tracked in a sequence containing background clutter and fast motions, which emphasize the need for correct initialization. We randomly generated 100 initial bounding boxes overlapping the object of interest (Fig. 12) and counted the correctly tracked frames (Tab. 6).

In the experiment, a frame was declared as correctly tracked if the overlap with the ground truth was greater than 0.3. The tracker with the P_Σ predictor performed about 90% better than the tracker with the $P_{FB \circ \rho}$ predictor and it was able to track the object correctly up to frame 84 on average.

Table 6 Evaluation of filtering methods in terms of the number of correctly tracked frames with randomly initialized bounding box (see. Fig. 12). The “score” is the total number of correctly tracked frames, the mean and the median of the same quantity are presented in the right column.

Method	Score	mean (median)
$P_{FB \circ \rho}$ [ref]	4493	45 (21)
P_Σ	8438	84.4 (99.5)

Figs. 13(a) and 13(b) show the histograms of the number of correctly tracked frames for 100 runs with different initialization and Fig. 13(c) shows the 2D histogram of the number of correctly tracked frames by $P_{FB \circ \rho}$ and P_Σ initialized

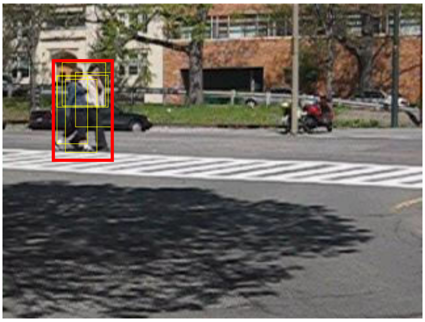
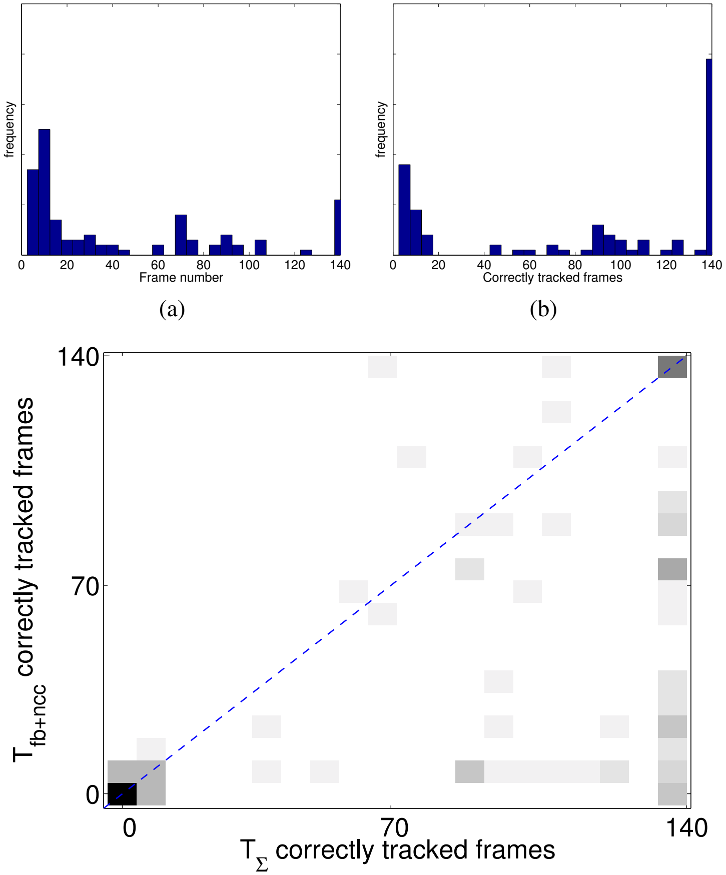


Fig. 12 Examples of randomly generated initial bounding boxes (yellow) randomly generated within the red rectangle.



(a)

(b)

(c)

Fig. 13 Histograms of the number of correctly tracked frames for tracker with (a) $P_{FB\circ\rho}$ and (b) P_Σ. (c) The 2D histogram of the number of correctly tracked frames by $P_{FB\circ\rho}$ and P_Σ initialized with the same random bounding box.

with the same random bounding box (to compare performance for individual random initialization).

7 Conclusions

We have presented a set of enhancements of the Flock of Trackers. First, new reliability prediction methods were introduced - the Neighbourhood consistency predictor and the Markov predictor.

Next, two methods for combining predictors, the ad-hoc $\mathscr{P}_\wedge$ and the likelihood thresholding $\mathscr{P}_\Theta$, were proposed and compared and similar performance was achieved. We decided to use $\mathscr{P}_\wedge$, because it is a straightforward approach without the need of learning the relevant statistics in advance.

Combined with the normalized cross-correlation predictor, the new Markov and Neighbourhood consistency predictors form a reliable compound predictor P_{Σ}. The P_{Σ} predictor was compared with the published $\mathrm{P}_{\mathrm{FB}\circ\rho}$ predictor and outperformed it in all criteria, i.e. in speed, recall, the number of reinitialization and the robustness to bounding box initialization. The simpler $\mathrm{P}_{N\circ M}$ combination performed almost identically and is faster. Finally, we have shown that the RANSAC-based global object motion estimator outperforms the published median flow algorithm.

The enhanced FoT was extensively tested on 62 sequences. Most of the sequences are standard and used in the literature. The improved FoT showed performance superior to the reference method, which competes well with the state-of-the-art [14].

For all 62 sequences, the ground truth is available at `http://cmp.felk.cvut.cz/ vojirtom/dataset`. For some of the sequences the ground truth has not been in the public domain till now.

Acknowledgements. The authors were supported by Czech Science Foundation Project GACR P103/12/G084 (JM) and by the Grant Agency of the CTU Prague project SGS13/142/OHK3/2T/13 (TV).

References

1. Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: CVPR (2006)
2. Chum, O., Matas, J., Kittler, J.: Locally optimized ransac. Pattern Recognition (2003)
3. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: CVPR (2000)
4. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM (1981)
5. Grabner, H., Matas, J., Van Gool, L., Cattin, P.: Tracking the invisible: Learning where the object might be. In: CVPR (June)
6. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. In: CVPR (2010)
7. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward-Backward Error: Automatic Detection of Tracking Failures. In: ICPR (2010)
8. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. PAMI (2012)
9. Klsch, M., Turk, M.: Fast 2d hand tracking with flocks of features and multi-cue integration. In: Workshop at CVPR (2004)
10. Kwon, J., Lee, K.: Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling. In: CVPR (2009)
11. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision (ijcai). In: IJCAI (1981)
12. Reynolds, C.W.: Flocks, herds and schools: A distributed behavioral model. In: SIGGRAPH (1987)
13. Shahed Nejhum, S., Ho, J., Yang, M.H.: Visual tracking with histograms and articulating blocks. In: CVPR (2008)
14. Vojíř, T., Matas, J.: Robustifying the flock of trackers. In: CVWW (2011)

Registration and Segmentation in Medical Imaging

Daniel Rueckert and Julia A. Schnabel

1 Introduction

The analysis of medical images plays an increasingly important role in many clinical applications. Different imaging modalities often provide complementary anatomical information about the underlying tissues such as the X-ray attenuation coefficients from X-ray computed tomography (CT), and proton density or proton relaxation times from magnetic resonance (MR) imaging. The images allow clinicians to gather information about the size, shape and spatial relationship between anatomical structures and any pathology, if present. Other imaging modalities provide functional information such as the blood flow or glucose metabolism from positron emission tomography (PET) or single-photon emission tomography (SPECT), and permit clinicians to study the relationship between anatomy and physiology. Finally, histological images provide another important source of information which depicts structures at a microscopic level of resolution.

At the same time the amount and complexity of the data generated by medical imaging modalities is also significantly increasing: Patients are often imaged with 3D imaging modalities but also are monitored over time to assess disease status or response to therapy. These longitudinal datasets produce 4D imaging data which is even more complex and costly to analyse by clinicians.

In order to fuse complementary image information, or detect structural or physiological changes occurring over time, the images need to be first brought into geometric alignment. The process of aligning an image pair, or a set of image sequences, is called image registration. Another key step in medical image analysis is the

Daniel Rueckert
Biomedical Image Analysis Group, Department of Computing, Imperial College London, UK
e-mail: d.rueckert@imperial.ac.uk

Julia A. Schnabel
Institute of Biomedical Engineering, Department of Engineering Science,
University of Oxford, UK
e-mail: julia.schnabel@eng.ox.ac.uk

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_7, © Springer-Verlag Berlin Heidelberg 2014

segmentation of the objects of interest. The result of this segmentation process is the grouping or labelling of voxels into meaningful, disjoint regions or objects. High demand on clinical expert time requires automation of both processes.

Recent advances in medical image analysis have led to the development of robust and accurate registration and segment ion algorithms. In addition, many of the state-of-art image segmentation techniques are use image registration, either implicitly or explicitly. The purpose of this chapter is to provide an introduction into some of the most commonly used image registration algorithms as well as their use in the context of image segmentation.

2 Registration

Image registration aims to find corresponding anatomical or functional locations in two or more images. Image registration can be applied to images from the same subject acquired by different imaging modalities or at different time points as well as to images acquired from different subjects. To bring images into registration it is usually necessary to estimate a geometric transformation which aligns the images. Most non-rigid registration techniques use either elastic [3, 24], fluid [12, 11, 6, 4] or other deformation models [53, 19, 33, 56, 59] to represent this geometric transformation. In this chapter we focus on registration algorithms which use a particular deformation model, namely free-form deformations based on B-splines.

In general, finding the optimal geometric transformation is achieved by minimization of a cost function which measures the degree of (mis-)alignment of the images as a function of the geometric transformation. Most registration algorithms use a cost function based on image intensity information to directly to measure the degree of (mis-)alignment of the images. These methods are called voxel-based registration techniques and are especially successful since they do not require any feature extraction or segmentation of the images. Comprehensive reviews of image registration techniques can be found in [44, 27, 65].

The goal of image registration is to relate any point in the reference or *target* image to the *source* image, i.e. to find the optimal transformation $\mathbf{T} : \mathbf{p} \mapsto \mathbf{p}'$ which maps any point in the target image $\mathscr{I}_A$ into its corresponding point in the source image $\mathscr{I}_B$. The transformation $\mathbf{T}$ can be separated into two components: A global component (e.g. a rigid or affine transformation) and a local component. Thus, the transformation $\mathbf{T}$ can be written as:

$$\mathbf{T}(\mathbf{p}) = \mathbf{T}_{global}(\mathbf{p}) + \mathbf{T}_{local}(\mathbf{p}) \tag{1}$$

The global transformation typically accounts for variations in the position, orientation and scaling between the two images. However, the global transformation cannot account for any local deformations. A common model for the local transformations is based on a free-form deformation model based on B-splines which will be reviewed in the next section.

Free-Form Deformations

Free-form deformations (FFDs) are a concept stemming from the computer graphics community, developed for modeling 3D deformable objects [55]. In image registration, they have been adopted to deform an entire image volume by manipulating an underlying mesh of regularly spaced control points, using smooth and continuous interpolation techniques in between. In particular, using FFDs in combination with cubic B-splines for medical image registration was first proposed by Rueckert et al. [52, 53], and have attracted significant further interest in the medical imaging community [48, 50, 36, 45].

To define a spline-based FFD we denote the domain of the image volume as $\Omega = \{\mathbf{p} = (x,y,z) \mid 0 \leq x < X, 0 \leq y < Y, 0 \leq z < Z\}$. Let Φ denote a $n_x \times n_y \times n_z$ mesh of control points $\phi_{i,j,k}$ with uniform control point spacing δ. Then, the FFD can be written as the 3D tensor product of the familiar 1D cubic B-splines:

$$\mathbf{T}_{local}(\mathbf{p}) = \sum_{l=0}^{3}\sum_{m=0}^{3}\sum_{n=0}^{3} B_l(u)B_m(v)B_n(w)\phi_{i+l,j+m,k+n} \tag{2}$$

where $i = \lfloor \frac{x}{\delta} \rfloor - 1, j = \lfloor \frac{y}{\delta} \rfloor - 1, k = \lfloor \frac{z}{\delta} \rfloor - 1, u = \frac{x}{\delta} - \lfloor \frac{x}{\delta} \rfloor, v = \frac{y}{\delta} - \lfloor \frac{y}{\delta} \rfloor, w = \frac{z}{\delta} - \lfloor \frac{z}{\delta} \rfloor$ and where B_l represents the l-th basis function of the B-spline [37, 38]:

$$\begin{aligned}
B_0(u) &= (1-u)^3/6 \\
B_1(u) &= (3u^3 - 6u^2 + 4)/6 \\
B_2(u) &= (-3u^3 + 3u^2 + 3u + 1)/6 \\
B_3(u) &= u^3/6
\end{aligned}$$

Other spline-based methods used in medical imaging include thin-plate splines [5] or elastic-body splines [20], which however suffer from the disadvantage of global control, making them computationally complex for modelling localised deformations. B-splines, in contrast, are locally controlled due to the finite support of the cubic basis functions, which allow for computationally very efficient local deformation modelling. More specifically, changing a control point $\phi_{i,j,k}$ only affects the local neighbourhood around this point.

One advantage of using B-splines is that analytic derivatives of the transformation can be calculated by differentiating the B-spline basis functions. The first order derivative of $\mathbf{T}_{local}(\mathbf{p})$ with respect to x thus becomes:

$$\frac{\partial \mathbf{T}_{local}(\mathbf{p})}{\partial x} = \frac{1}{\delta_x}\sum_{l=0}^{3}\sum_{m=0}^{3}\sum_{n=0}^{3} \frac{dB_l(u)}{du} B_m(v)B_n(w)\phi_{i+l,j+m,k+n} \tag{3}$$

with the other derivatives taking analogous forms. Calculating the derivatives of transformations is useful for investigating the quality of the deformation field or for explicit regularization during optimization. For example, the determinant of the Jacobian matrix of the transformation (also called the Jacobian) can give invaluable insights into local deformation properties in terms of infinitesimal local volume changes due to contraction or expansion. At any point **p** it is is defined as:

$$\mathbf{J}(\mathbf{p}) = \det \begin{pmatrix} \frac{\partial T_x(\mathbf{p})}{\partial x} & \frac{\partial T_x(\mathbf{p})}{\partial y} & \frac{\partial T_x(\mathbf{p})}{\partial z} \\ \frac{\partial T_y(\mathbf{p})}{\partial x} & \frac{\partial T_y(\mathbf{p})}{\partial y} & \frac{\partial T_y(\mathbf{p})}{\partial z} \\ \frac{\partial T_z(\mathbf{p})}{\partial x} & \frac{\partial T_z(\mathbf{p})}{\partial y} & \frac{\partial T_z(\mathbf{p})}{\partial z} \end{pmatrix} = \begin{cases} > 1 & : \text{volume expansion} \\ = 1 & : \text{no volume change} \\ < 1 & : \text{volume contraction} \end{cases} \tag{4}$$

For negative values, the transformation is *folding* and becomes no longer invertible; for very large positive values, the transformation is undergoing *tearing*.

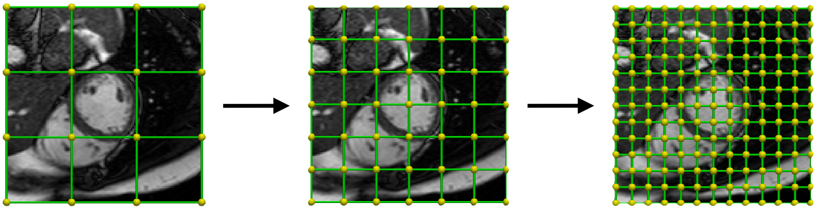


Fig. 1 Successive levels of subdivision of a B-spline FFD defined for a cardiac MR image

The resolution and computational complexity of the FFD mesh $\boldsymbol{\Phi}$ is controlled by the spacing of the control points ϕ, which act as parameters (or degrees of freedom) of the transformation. A large control point spacing allows to model global deformations, whereas a small spacing allows to model very localised, small deformations. To combine both properties efficiently, multi-level B-splines [39] can be used for recovering deformations in a coarse-to-fine fashion [54]. Furthermore, computational complexity can be reduced by using accurate B-spline subdivision between resolution levels (see Figure 1). For example in 1D, the B-spline resolution can be doubled by computing the new the control point positions ϕ' of the refined grid from the coarse control points ϕ [23] as:

$$\phi'_{2i+1} = \frac{1}{2}(\phi_i + \phi_{i+1}) \quad \text{and} \quad \phi'_{2i} = \frac{1}{8}(\phi_{i-1} + \phi_{i+1} + 6\phi_i) \tag{5}$$

Generalizing this equation to 3D is straightforward, and is achieved by applying again the 3D tensor product. FFDs can also be made non-uniform by assigning a control point status for each ϕ, allowing it to deform if *active*, or remain fixed if *passive*. In combination with multi-level splines, a strategy can be to keep the number of active control points fixed per level to have a constant computational complexity per level, by successively focussing into deformations of regions of interest [54]. An alternative to non-uniform multi-level B-splines for FFDs are non-uniform rational B-splines (NURBS), as proposed by Wang and Jiang [63] in the context for FFD based registration.

A common approach is to define FFDs in a Cartesian coordinate system. However, some clinical applications can be easier described in other coordinate systems. For example, Chandrashekara et al. [9] proposed the use of a FFD model defined in a polar coordinate system for the registration of cardiac MR images, due to the cyclic motion involved. Similarly, Lin et al. [40] proposed extended free-form deformations (EFFD) [17] for the registration of cardiac MR images. A more recent and generic approach was developed by Chandrashekara et al. [8] again, where FFDs are defined on lattices of arbitrary topology [42], which has the advantage that a control point mesh Φ can be adapted to the geometry of the anatomy under investigation, such as the epi- and endocardial surfaces of the left cardiac ventricle.

In the following, we will discuss the use of voxel-based similarity measures, which can be used in conjunction with B-spline FFDs (amongst other deformation models) in order to solve an image registration problem.

Voxel-Based Similarity Measures

To relate a point in the target image to the source image, one must define a similarity criterion (or cost function) which measures the degree of alignment between both images. A popular choice for this are voxel-based similarity measures which use the image intensities directly and do not require the extraction of any features such as a landmarks, curves or surfaces. The simplest statistical measure of image similarity is based on the squared sum of intensity differences (SSD) between $\mathscr{I}_A$ and $\mathscr{I}_B$,

$$\mathscr{S}_{SSD} = -\frac{1}{n}\sum(\mathscr{I}_A(\mathbf{p}) - \mathscr{I}_B(\mathbf{T}(\mathbf{p})))^2 \tag{6}$$

where n is the number of voxels in the region of overlap. This measure is based on the assumption that both imaging modalities have the same characteristics. If the images are correctly aligned, the difference between them should be zero except for the noise produced by the two modalities. If this noise is Gaussian distributed, it can be shown that the SSD is the optimal similarity measure (Viola [60]). Since this similarity measure assumes that the imaging modalities are identical, their application is restricted to mono-modal applications such as serial registration [29, 28].

In a number of cases, the assumption of identical imaging modalities is too restrictive. A more general assumption is that of a linear relationship between the two images. In these cases, the similarity between both images can be expressed by the normalised cross correlation (CC)

$$\mathscr{S}_{CC} = \frac{\sum(\mathscr{I}_A(\mathbf{p}) - \mu_A)(\mathscr{I}_B(\mathbf{T}(\mathbf{p})) - \mu_B)}{\sqrt{(\sum\mathscr{I}_A(\mathbf{p}) - \mu_A)^2(\sum\mathscr{I}_B(\mathbf{T}(\mathbf{p})) - \mu_B)^2}} \tag{7}$$

where μ_A, μ_B correspond to average voxel intensities in both images. Nevertheless, the application of this similarity measure is largely restricted to mono-modal registration tasks.

Note that both similarity measures make strong assumptions about the relationship of the image intensities in both images which is not suitable for multi-modality registration. Even in the case of mono-modality registration this assumption is often violated, e.g. in contrast-enhanced imaging. To deal with this challenge several similarity measures have been proposed that are based on the information content or entropy of the registered image. An important step to understanding these methods is the feature space of the image intensities which can also be interpreted as the joint probability distribution: A simple way of visualizing this feature space is by accumulating a two-dimensional histogram of the co-occurrences of intensities in the two images for each trial alignment. If this feature space is visualized for difference degrees of image alignment it can be seen that the feature space disperses as misalignment increases, and that each image pair has a distinctive feature space signature at alignment.

In an information theoretic framework the information content of images can be defined as the Shannon-Wiener entropy, $H(\mathscr{I}_A)$ and $H(\mathscr{I}_B)$ of images $\mathscr{I}_A$ and $\mathscr{I}_B$:

$$H(\mathscr{I}_A) = -\sum_{a \in \mathscr{I}_A} p(a) \log p(a) \tag{8}$$

and

$$H(\mathscr{I}_B) = -\sum_{b \in \mathscr{I}_B} p(b) \log p(b) \tag{9}$$

where $p(a)$ is the probability that a voxel in image $\mathscr{I}_A$ has intensity a and $p(b)$ is the probability that a voxel in image $\mathscr{I}_B$ has intensity b. The joint entropy $H(\mathscr{I}_A, \mathscr{I}_B)$ of the overlapping region of image $\mathscr{I}_A$ and $\mathscr{I}_B$ may be defined by

$$H(\mathscr{I}_A, \mathscr{I}_B) = -\sum_{a \in \mathscr{I}_A} \sum_{b \in \mathscr{I}_B} p(a,b) \log p(a,b) \tag{10}$$

where $p(a,b)$ is the joint probability that a voxel in the overlapping region of image $\mathscr{I}_A$ and $\mathscr{I}_B$ has values a and b, respectively.

To derive a measure of image alignment one can use concepts from information theory such as mutual information [13, 61]. Mutual information (MI) is defined in term of entropies as

$$\mathscr{S}_{MI}(\mathscr{I}_A; \mathscr{I}_B) = H(\mathscr{I}_A) + H(\mathscr{I}_B) - H(\mathscr{I}_A, \mathscr{I}_B) \tag{11}$$

and should be maximal at alignment. Mutual information is a measure of how one image "explains" the other but makes no assumption of the functional form or relationship between image intensities in the two images. It has been shown by Studholme [58] that mutual information itself is not independent of the overlap between two images. To avoid any dependency on the amount of image overlap, Studholme suggested the use of normalised mutual information (NMI) as a measure of image alignment:

$$\mathscr{S}_{NMI}(\mathscr{I}_A;\mathscr{I}_B) = \frac{H(\mathscr{I}_A) + H(\mathscr{I}_B)}{H(\mathscr{I}_A,\mathscr{I}_B)} \tag{12}$$

Similar forms of normalised mutual information have been proposed by Maes et al. [43].

Information-theoretic voxel similarity measures are based on the notion of the marginal and joint probability distributions of the two images. These probability distributions can be estimated in two different ways: The first method uses histograms whose bins count the frequency of occurrence (or co-occurrence) of intensities. Dividing these frequencies by the total number of voxels yields an estimate of the probability of that intensity. The second method is based on generating estimates of the probability distribution using Parzen windows [21] which is a non-parametric technique to estimate probability densities. The Parzen-based approach has the advantage of providing a differentiable estimate of mutual information which is not the case of the histogram-based estimate of mutual information.

A disadvantage of the voxel-similarity based measures described above is that they are global and do not take spatial context into account. Recent developments in the field include structural image representations, where the idea is to extract feature vectors for each image voxel from a spatial neighbourhood, which then can be directly compared between different images using very simple measures such as SSD. One such example is the modality independent neighborhood descriptor (MIND) developed by Heinrich et al. [31]. MIND provides a vector representation and is based on the principles of image self-similarity originally proposed for non-local means filtering by Buades et al. [7] for the purpose of image denoising. In that work, in order to estimate a noise-free intensity for a given voxel, a weighted average of all other voxels in the non-local neighborhood N in the image $\mathscr{I}$ are calculated as:

$$\mathscr{I}_{new}(\mathbf{x}_i) = \sum_{j \in N} w(\mathbf{x}_i,\mathbf{x}_j)\mathscr{I}(\mathbf{x}_j) \tag{13}$$

with the weights w between the voxel of interest, $\mathbf{x}_i$ to a neighborhood voxel $\mathbf{x}_j$ calculated as:

$$w(\mathbf{x}_i,\mathbf{x}_j) = \exp^{-\frac{\sum_{\Delta\mathbf{x}} \|\mathscr{I}(\mathbf{x}_i+\Delta\mathbf{x}) - \mathscr{I}(\mathbf{x}_j+\Delta\mathbf{x})\|^2}{\sqrt{2}\sigma^2}} \tag{14}$$

where $\Delta\mathbf{x}$ is defined over the range of voxels within a patch, and σ defines the local variance of the noise estimate from the data. MIND is then based on the SSD between the weights obtained from an image pair, by calculating the vector distance (or SSD). It was shown to be a very robust measure for noisy data due to its intrinsic smoothing properties, and applicable to the challenging task of multi-modal lung image registration, due to its relative independence from the underlying image intensities and bias fields.

Optimization

Like many other problems in computer vision, image registration can be formulated as an optimisation problem whose goal is to minimise an associated energy or cost function. A generic optimisation procedure for image registration is outlined in Figure 2. The objective or cost function used in image registration can be written as:

$$\mathscr{C} = -\mathscr{S} + \lambda \mathscr{P} \tag{15}$$

This type of cost function comprises two competing goals: The first term represents the cost associated with the voxel-based image similarity measure $\mathscr{S}$, while the second term $\mathscr{P}$ penalizes certain transformations and thus constrains the behavior of the transformation (different penalty functions will be discussed in the next section). The parameter λ is a weighting parameter which defines the trade-off between the alignment of the two images and the penalty function of the transformation. From a probabilistic point of view, the cost function in eq. (15) can be explained in a Bayesian context: The similarity measure can be viewed as a likelihood term which expresses the probability of a match between source and target image while the penalty function represents a prior which encodes a-priori knowledge about the expected transformation.

In most implementations of free-form image registration, the cost function is minimised via gradient descent optimisation. However, this can be computationally very expensive. More recently, Modat et al. [47] proposed an efficient and parallelizable version of the gradient descent algorithm for free-form image registration. They report run times of less than 1 minute using a GPU implementation of the algorithm. An alternative approach based on discrete optimisation via Markov Random Fields (MRFs) have been proposed by Glocker et al. [25] and further developed by Heinrich [32]. A comparison of different optimization strategies for FFDs can be found in [35].

Penalty Functions for Free-Form Deformations

Typically, non-rigid image registration is an ill-posed problem. Thus, it is necessary to add some constraints to render the problem well-posed. A common approach is enforce the smoothness of the deformation [53]. Free-form deformations based on B-splines are intrinsically smooth (at least relative to the control point spacing), however additional smoothness can be enforced by adding a penalty term which regularizes the transformation. The general form of such a smoothness penalty term has been described by Wahba [62]. In 3D, the penalty term takes the following form

$$\begin{aligned}\mathscr{P}_{smooth} = \int & \left(\frac{\partial^2 \mathbf{T}}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{T}}{\partial z^2}\right)^2 + \\ & 2\left(\frac{\partial^2 \mathbf{T}}{\partial xy}\right)^2 + 2\left(\frac{\partial^2 \mathbf{T}}{\partial xz}\right)^2 + 2\left(\frac{\partial^2 \mathbf{T}}{\partial yz}\right)^2 d\mathbf{p}\end{aligned} \tag{16}$$

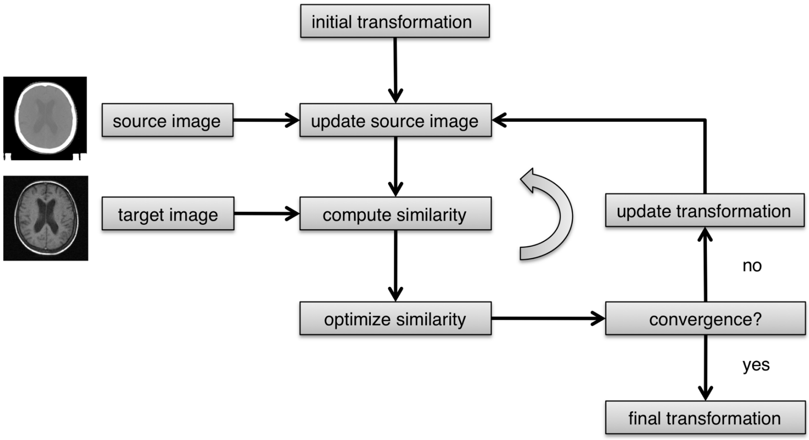


Fig. 2 Outline of a generic image registration algorithm.

This quantity is the 3D counterpart of the 2D bending energy of a thin-plate of metal and defines a cost function which is associated with the smoothness of the transformation. Note that this regularization term is zero for an affine transformation and therefore penalises only non-affine transformations [62].

FFDs using B-splines have implicit assumptions of smoothness of the deformations. However, they can still suffer from physiologically implausible, if unrealistic, local changes of volume in tissues or bone. To impose physical plausibility onto the deformations, Rohlfing et al. suggested a constraint which preserves volume [50]:

$$\mathscr{P}_{volume} = \int |\log(J(\mathbf{p}))|\,d\mathbf{p} \tag{17}$$

Here $J(\mathbf{p})$ is the determinant of the Jacobian matrix $\mathbf{J}$ of the free-form deformation. As mention previously the Jacobian measures how infinitesimal volumes change under the transformation. This function therefore penalizes the compression or expansion of tissues or organs during the registration. It should be noted that the penalty term above penalizes volume changes over the entire domain, however due to the integration there may be small regions in the image which show a large volume change while the majority of regions show no volume change. Other authors have proposed a rigidity constraint which forces the deformation in certain regions to be nearly rigid [41], e.g.

$$\mathscr{P}_{rigidity} = \int ||\mathbf{J}(\mathbf{p})\mathbf{J}(\mathbf{p})^T - \mathbf{1}||\,d\mathbf{p} \tag{18}$$

The penalty functions above do not guarantee that the resulting deformation field is diffeomorphic (smooth and invertible). In order to ensure that the FFD is diffeomorphic it is possible to add a penalty function which penalizes non-diffeomorphic transformations, e.g. transformations which introduce folding. One suitable penalty function for this has the following form:

$$\mathscr{P}_{folding} = \int \mathscr{P}(\mathbf{p})d\mathbf{p} \tag{19}$$

where

$$\mathscr{P}(\mathbf{p}) = \begin{cases} \frac{\gamma^2}{|(J(\mathbf{p})|^2} - 2 & \text{if } |J(\mathbf{p})| \leq \gamma \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

A similar penalty function was first proposed by Edwards et al. [22] and effectively penalises any transformations for which the determinant of the Jacobian falls below a threshold γ. By penalising Jacobians that approach zero, one can prevent the transformation from collapsing and ensure diffeomorphisms. Note that simply using a smoothness penalty function would not be sufficient to guarantee a diffeomorphic transformation, since it is possible for a transformation to be smooth but non-diffeomorphic.

Diffeomorphic Free-Form Deformations

In general, most registration algorithms make the assumption that similar structures are present in both images. Therefore it is desirable that the deformation field be smooth and invertible (so that every point in one image has a corresponding point in the other). Such smooth, invertible transformations are called diffeomorphisms. Choi and Lee [10] have derived sufficient conditions for the injectivity of FFDs which are represented in terms of control point displacements. These sufficient conditions can be easily tested and can be used to guarantee a diffeomorphic FFD. Without loss of generality we will assume in the following that the control points are arranged on a lattice with unit spacing. Let $\Delta\mathbf{c}_{i,j,k} = (\Delta x_{i,j,k}, \Delta y_{i,j,k}, \Delta z_{i,j,k})$ be the displacement of control point $\mathbf{c}_{i,j,k}$. Let $\delta_x = \max|\Delta x_{i,j,k}|$, $\delta_y = \max|\Delta y_{i,j,k}|$, $\delta_z = \max|\Delta z_{i,j,k}|$.

Theorem 0.1. *A FFD based on cubic B-splines is locally injective over all the domain if $\delta_x < \frac{1}{K}$, $\delta_y < \frac{1}{K}$ and $\delta_z < \frac{1}{K}$.*

Choi and Lee [10] have determined a value of $K \approx 2.48$ so that the maximum displacement of control points given by the bound $\frac{1}{K}$ is approximately 0.40. This means that the maximum displacement of control points is determined by the spacing of control points in the lattice. For example, for a lattice with 20mm control point spacing the maximum control point displacement is 8mm while for a lattice with 2.5mm control point spacing the maximum control point displacement is 1mm. In practice the bounds on the displacements are too small to model any realistic deformations. To model large deformations one can use a composition of FFDs as proposed in [26]. For each FFD in this composition, the maximum control point displacement is limited by theorem 1. This a fundamentally different to the multi-level FFDs mentioned earlier since the FFDs are concatenated,

$$\mathbf{T}(\mathbf{p}) = \mathbf{T}_n \circ \mathbf{T}_{n-1} \circ \cdots \circ \mathbf{T}_2 \circ \mathbf{T}_1(\mathbf{p}) \tag{21}$$

so that the final deformation is a composition of FFDs. Since the composition of two diffeomorphisms produces a diffeomorphism one can construct a diffeomorphic deformation by ensuring that each individual FFD is diffeomorphic.

3 Segmentation

The amount of data produced by imaging increasingly exceeds the capacity for expert visual analysis, resulting in a growing need for automated image analysis. In particular, accurate and reliable methods for segmentation (classifying image regions) are a key requirement for the extraction of information from images. In recent years many approaches to image segmentation have emerged that use image registration as a key comment. Many of these approaches are based on so-called *atlases*. An atlas can be viewed as a map or chart of the anatomy or function, either from a single individual or from an entire population. In many cases atlases the atlases are annotated to include geometric information about points, curves or surfaces, or label information about voxels (anatomical regions or function).

Atlases can be used as prior information for image segmentation. In general, an atlas $\mathscr{A}$ can be viewed as a mapping from a set of spatial coordinates (i.e. the voxels) to a set of labels $\Lambda = \{1, \cdots, L\}$. By warping the atlas to the target, one can make the atlas and its prior information *subject-specific* and obtain a segmentation $\mathscr{L}$ of image $\mathscr{I}$:

$$\mathscr{L} = \mathscr{A} \circ \mathbf{T}_{\mathscr{A} \rightarrow \mathscr{I}} \tag{22}$$

Indeed the earliest approaches to segmentation via registration have used such approaches: By registering a labelled atlas to the target images and transforming the segmentation of the atlas into the coordinate system of the subject one can obtain a segmentation of the subject's image [46, 14]. This segmentation approach is simple yet effective since the approach can segment any of the structures that are present and annotated in the atlas. However, the accuracy and robustness of the segmentation is dictated by the accuracy and robustness of the image registration. Errors in the registration process will directly affect the accuracy of the propagated segmentation.

Multi-atlas Segmentation

In the area of machine learning it is well know that the performance of pattern recognition techniques can be boosted using combining classifiers [34]. This concept can be exploited in the context of atlas-based segmentation: Assuming the availability of multiple atlases, the output of atlas-based segmentation using a particular atlas instance can be viewed the output of the classifier. Combining the output of multiple classifiers (or segmentations) into a single consensus segmentation has been show to reduce random errors in the individual atlas-to-image registration resulting in an improved segmentation [49, 30]. Using this method each atlas is registered to the the target image in question. The resulting transformation is then used to transform the segmentation from the atlas into the coordinate system of the target image. An example of this process is shown Figure 3.

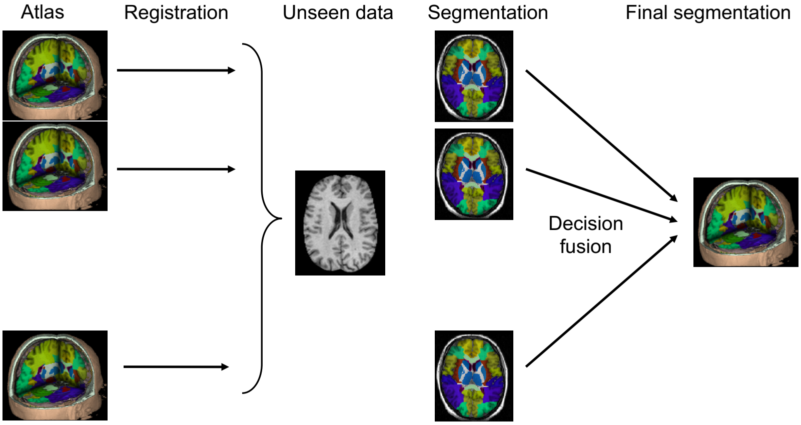


Fig. 3 Example of multi-atlas segmentation [30]: A set of annotated atlases is registered to an unseen image (target image). Each pair of atlas and registration produces a single segmentation which may contain errors. The segmentations are then merged into a single segmentation through a decision or vote fusion step into a final consensus segmentation.

By applying classifier fusion techniques at every voxel in subject space the final consensus segmentation can be applied. Several classifier fusion techniques can be used, see [34] for a detailed review and discussion of the different classifier fusion techniques. One of the most popular techniques is the majority vote rule [49]: It simply uses a *winner-takes-all* approach in which each voxel is assigned the label that gets the most votes from the individual segmentations. Assuming K classifiers (i.e. atlases) final segmentation $\mathscr{L}(\mathbf{p})$ can be expressed as

$$\mathscr{L}(\mathbf{p}) = \max[f_1(\mathbf{p}), \cdots, f_L(\mathbf{p})] \tag{23}$$

where

$$f_l(\mathbf{p}) = \sum_{k=1}^{K} w_{k,l}(\mathbf{p}) \quad \text{for } l = 1, \cdots, L \tag{24}$$

and

$$w_{k,l}(\mathbf{p}) = \begin{cases} 1, & \text{if } l = e_k(\mathbf{p}) \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

Here e_k denotes the output or label of classifier k. An extension of multi-atlas segmentation has been proposed in [1]. In their work a large number of atlases are used.

However, instead of using all atlases for for multi-atlas segmentation, only the most similar atlases are used: In the first step all atlases are registered to a common standard space using a coarse registration (e.g. affine registration). In addition, the target image is also aligned to the common standard space. After this initial alignment the similarity between each atlas and the target image can be determined using an image similarity measure $\mathscr{S}$, e.g. sums of squared differences (SSD), cross-correlation (CC), mutual information (MI) [15, 61] or normalised mutual information (NMI) [57]. This allows the ranking of all atlases with respect to the similarity to the target image. The m top-ranked atlases are then registered non-rigidly to the target image and as before a classifier fusion framework is applied to obtain a final consensus segmentation.

The use of a common standard space allows the pre-registration of all atlases to the standard common space avoiding the necessity for performing registration of each atlas to the target image for atlas selection. In principle it is also possible to rank atlases based on meta-information available from the atlases and the target image. Such meta-information can include gender, age, handedness and clinical status. In this case atlas selection can be carried out independently from the actual image data and does not require any initial registration for the atlas selections step.

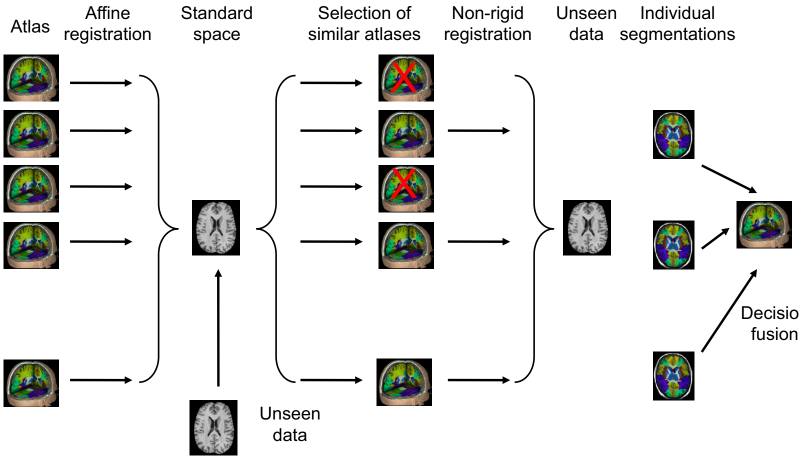


Fig. 4 Example of multi-atlas segmentation with atlas selection[1]: A set of annotated atlases and the unseen image (target image) are affinely registered to a common standard space. In this common standard space the most similar atlases are then identified using a similarity measure. The registration of the most similar atlases is then refined using non-rigid registration and resulting segmentations are fused in a similar fashion to standard multi-atlas segmentation.

Instead of ranking atlases based on their similarity to the target image and using the top m atlases for classifier fusion, it is possible to weight each atlas according to its similarity to the target image. In this case the weight w can be written as

$$w_{k,l}(\mathbf{p}) = \begin{cases} \mathscr{S}, & \text{if } l = e_k(\mathbf{p}) \\ 0, & \text{otherwise} \end{cases} \tag{26}$$

where $\mathscr{S}$ measures the similarity between atlas $\mathscr{A}_k$ and the target image. It should be noted that the atlas selection scheme can be viewed as a special case of the weighted atlas fusion scheme described above where $w = 1$ for the top-ranked atlases and $w = 0$ for all other atlases.

While weighted voting allows the incorporation of a notation of atlas similarity into the classifier fusion, it does not account for the fact that images can be dissimilar at a global level but similar at a local level and vice versa. For example, two brain MR images may have ventricles that are very different in size and shape but their hippocampi may have similar shape and size. Since the ventricle is much larger than the hippocampus, its appearance will dominate the similarity calculations. A more flexible approach is to measure image similarity locally and to adjust the weighting function accordingly:

$$w_{k,l}(\mathbf{p}) = \begin{cases} \mathscr{S}(\mathbf{p}), & \text{if } l = e_k(\mathbf{p}) \\ 0, & \text{otherwise} \end{cases} \tag{27}$$

Another approach is based on simultaneous truth and performance level estimation (STAPLE) [64]. The STAPLE framework was initially created in order to fuse several manual or automated segmentations of the same image. More specifically it computes a probabilistic estimate of the true segmentation as a measure of the performance level represented by each segmentation in an expectation-maximization (EM) framework. This framework has extended to account for spatially varying performance by extending the performance level parameters to account for a smooth, voxelwise performance level field that is unique to each atlas-based segmentation [16, 2].

4 Patch-Based Segmentation

A key challenge for multi-atlas segmentation techniques is the reliance on non-rigid registrations between the atlases and the target image. This introduces two disadvantages: First, the multi-atlas segmentation is computationally very expensive since each atlas must be registered. Secondly, each registration must be very accurate in order to guarantee good segmentation performance. However, this is difficult in the presence of large anatomical variations between the atlas database and the target image.

To overcome this problem, a patch-based label fusion strategy has been proposed in [18, 51]. In this approach the assumption of accurate correspondences between the atlas and the target image is relaxed. Instead a patch from the target image is compared to patches within a certain search region in the atlas database. In the subsequent label fusion strategy a patch-based weighting is used instead of a voxel-based weighting in order to fuse the labels. An overview of this approach is shown in Figure 5.

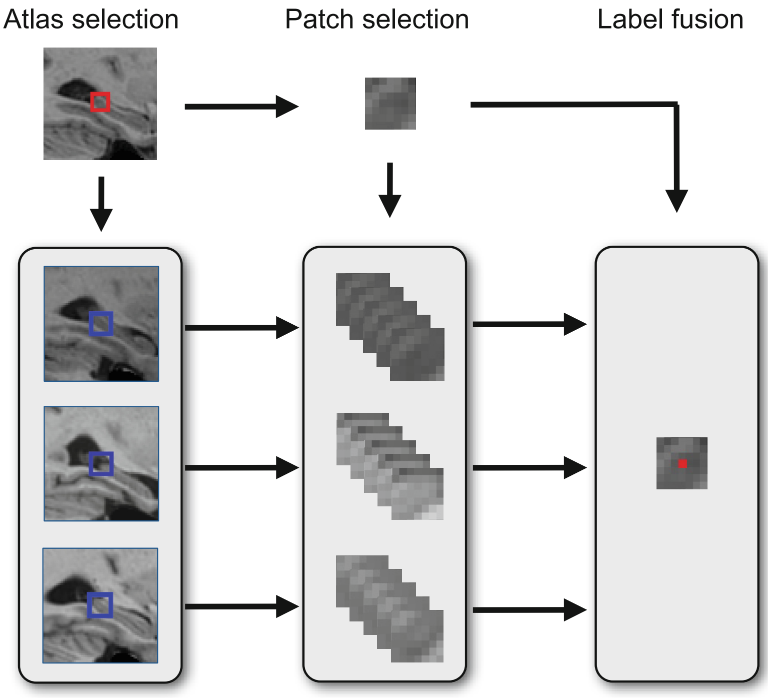


Fig. 5 Example of patch-based segmentation as proposed in [18, 51]: A set of atlas is first selected based on similarity. After this the most similar patches from the atlases are extracted and used in a weighted label fusion strategy to form the final estimate of the segmentation.

The first stage of the patch-based segmentation is very similar to multi-atlas segmentation with patch selection. The most similar atlases are selected based on a voxel similarity measure. After atlas selection, for each voxel and patch in the target image a search volume $\mathscr{V}_{\mathbf{p}}$ is defined as a cubic region centred around that voxel. The search region defines the area in which we expect to find corresponding patches in the atlas database. In addition to atlas selection, a separate patch selection is also perform to reduce the computational complexity. This is done using the following structural similarity measure:

$$\mathscr{S} = \frac{2\mu_i \mu_{j,k}}{\mu_i^2 + \mu_{j,k}^2} \frac{2\sigma_i \sigma_{j,k}}{\sigma_i^2 + \sigma_{j,k}^2} \tag{28}$$

Here μ represents the mean intensity and σ represents the standard deviation of the intensities in the patches centered on voxel $\mathbf{p}_i$ (the voxel under consideration) and voxel $\mathbf{p}_{j,k}$ at location j in atlas k. All patches for which the structural similarity measure is above a certain threshold γ are then retained for weighted label fusion. For this the weights are computed based on the patch similarity in terms of SSD:

$$w(\mathbf{p}_i, \mathbf{p}_{j,k}) = \begin{cases} \exp \frac{-||P(\mathbf{p}_i) - P(\mathbf{p}_{j,k})||_2^2}{h}, & \text{if } \mathscr{S} > \gamma \\ 0, & \text{otherwise} \end{cases} \tag{29}$$

where $P(\mathbf{p}_i)$ represents a stacked column vector of the intensities patch centered at $\mathbf{p}_i$ and h is a bandwidth parameter that controls how many samples are taken into account in the averaging. The final label is then obtained as the majority vote of f:

$$f(\mathbf{p}) = \frac{\sum_{k=1}^{K} \sum_{j \in \mathscr{V}_i} w(\mathbf{p}_i, \mathbf{p}_{j,k}) e_{j,k}}{\sum_{k=1}^{K} \sum_{j \in \mathscr{V}_i} w(\mathbf{p}_i, \mathbf{p}_{j,k})} \tag{30}$$

In this formulation $e_{j,k}$ denotes the label of atlas k and location j.

One of the most significant advantages of patch-based label fusion over atlas-based label fusion methods is that the accuracy of the registration is far less important. This is due to the fact that for each patch corresponding matches are searched for within a certain search region. In atlas-based label fusion this search region is effectively one voxel big. In contrast to this patch-based label fusion methods use search regions that are much bigger. As a result they can much better accommodate misregistration between the atlas and target image. In fact in [18, 51] only affine registrations are used to align the atlases and the target image. Despite this, the patch-based segmentation has been shown to be highly accurate and robust.

5 Summary and Conclusions

In this chapter we have presented a number of theoretical and practical aspects of medical image registration and segmentation. Medical image registration is widely used, both in clinical applications (e.g. image fusion, image-guided therapy or image-guided surgery) as well as a tool for biomedical research (e.g. to study populations in clinical trials). Despite this non-rigid registration is very much an area of on-going research and many most algorithms are still in the stage of development and evaluation. The lack of a generic gold standard for assessing and evaluating the success of non-rigid registration algorithms is one of their most significant drawbacks.

Registration algorithms also play a crucial role in state-of-the-art segmentation techniques. The advantage of atlas-based segmentation techniques is their generic nature: Different annotations present in the atlas can be propagated to the target image without the need to customise the process to different anatomical regions or objects. Moreover, multi-atlas techniques have been show to be very accurate and robust since they do not depend on individual registrations. This means that registration failures can be easily compensated for, especially if robust label fusion schemes such as majority voting is used.

References

1. Aljabar, P., Heckemann, R., Hammers, A., Hajnal, J., Rueckert, D.: Multi-atlas based segmentation of brain images: Atlas selection and its effect on accuracy. NeuroImage 46(3), 726–738 (2009)
2. Asman, A., Landman, B.A.: Formulating spatially varying performance in the statistical fusion framework. IEEE Transactions on Medical Imaging 31(6), 1326–1336 (2012)
3. Bajcsy, R., Kova**v**ci**v**c, S.: Multiresolution elastic matching. Computer Vision, Graphics and Image Processing 46, 1–21 (1989)
4. Beg, M.F., Miller, M.I., Younes, A.T.L.: Computing metrics via geodesics on flows of diffeomorphisms. International Journal of Computer Vision 61(2), 139–157 (2005)
5. Bookstein, F.L.: Principal Warps: Thin-plate splines and the decomposition of deformations. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(6), 567–585 (1989)
6. Bro-Nielsen, M., Gramkow, C.: Fast fluid registration of medical images. In: Höhne, K.H., Kikinis, R. (eds.) VBC 1996. LNCS, vol. 1131, pp. 267–276. Springer, Heidelberg (1996)
7. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale Modeling Simulation 4(2), 490–530 (2005)
8. Chandrashekara, R., Mohiaddin, R.H., Razavi, R.S., Rueckert, D.: Nonrigid image registration with subdivision lattices: Application to cardiac MR image analysis. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) MICCAI 2007, Part I. LNCS, vol. 4791, pp. 335–342. Springer, Heidelberg (2007)
9. Chandrashekara, R., Mohiaddin, R., Rueckert, D.: Analysis of myocardial motion and strain patterns using a cylindrical B-spline transformation model. In: Ayache, N., Delingette, H. (eds.) IS4TM 2003. LNCS, vol. 2673, pp. 88–99. Springer, Heidelberg (2003)
10. Choi, Y., Lee, S.: Injectivity conditions of 2D and 3D uniform cubic B-spline functions. Graphical Models 62(6), 411–427 (2000)
11. Christensen, G.E., Rabbitt, R.D., Miller, M.I.: Deformable templates using large deformation kinematics. IEEE Transactions on Image Processing 5(10), 1435–1447 (1996)
12. Christensen, G.E., Rabbitt, R.D., Miller, M.I., Joshi, S.C., Grenander, U., Coogan, T.A., van Essen, D.C.: Topological properties of smooth anatomic maps. In: Information Processing in Medical Imaging: Proc. 14th International Conference (IPMI 1995), pp. 101–112 (1995)
13. Collignon, A., Maes, F., Delaere, D., Vandermeulen, D., Seutens, P., Marchal, G.: Automated multimodality image registration using information theory. In: Information Processing in Medical Imaging: Proc. 14th International Conference (IPMI 1995), pp. 263–274 (1995)
14. Collins, D.L., Evans, A.C.: Animal: validation and applications of non-linear registration-based segmentation. International Journal of Pattern Recognition and Artificial Intelligence 11, 1271–1294 (1997)
15. Collins, D.L., Evans, A.C., Holmes, C., Peters, T.M.: Automatic 3D segmentation of neuro-anatomical structures from MRI. In: Information Processing in Medical Imaging: Proc. 14th International Conference (IPMI 1995), pp. 139–152 (1995)
16. Commowick, O., Akhondi-Asl, A., Warfield, S.K.: Estimating a reference standard segmentation with spatially varying performance parameters: local map staple. IEEE Transactions on Medical Imaging 31(8), 1593–1606 (2012)
17. Coquillart, S.: Extended free-form deformation: A sculpturing tool for 3D geometric modelling. Computer Graphics 24(4), 187–196 (1986)

18. Coupé, P., Manjón, J.V., Fonov, V., Pruessner, J., Robles, M., Collins, D.L.: Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. Neuroimage 54, 940–954 (2011)
19. Davatzikos, C.: Spatial transformation and registration of brain images using elastically deformable models. Computer Vision and Image Understanding 66(2), 207–222 (1997)
20. Davis, M.H., Khotanzad, A., Flamig, D.P., Harms, S.E.: A physics-based coordinate transformation for 3-D image matching. IEEE Transactions on Medical Imaging 16(3), 317–328 (1997)
21. Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis. Wiley (1973)
22. Edwards, P.J., Hill, D.L.G., Little, J.A., Hawkes, D.J.: A three-component deformation model for image-guided surgery. Medical Image Analysis 2(4), 355–367 (1998)
23. Forsey, D.R., Bartels, R.H.: Hierarchical B-spline refinement. ACM Transactions on Computer Graphics 22(4), 205–212 (1988)
24. Gee, J.C., Bajcsy, R.K.: Elastic matching: Continuum mechanical and probabilistic analysis. In: Toga, A.W. (ed.) Brain Warping, pp. 183–197. Academic Press (1999)
25. Glocker, B., Komodakis, N., Tziritas, G., Navab, N., Paragios, N.: Dense image registration through mrfs and efficient linear programming. Medical Image Analysis 12(6), 731–741 (2008)
26. Hagenlocker, M., Fujimura, K.: CFFD: a tool for designing flexible shapes. The Visual Computer 14(5/6), 271–287 (1998)
27. Hajnal, J.V., Hill, D.L.G., Hawkes, D.J. (eds.): Medical Image Registration. CRC Press (2001)
28. Hajnal, J.V., Saeed, N., Soar, E.J., Oatridge, A., Young, I.R., Bydder, G.M.: A registration and interpolation procedure for subvoxel matching of serially acquired MR images. Journal of Computer Assisted Tomography 19(2), 289–296 (1995)
29. Hajnal, J.V., Saeed, N., Oatridge, A., Williams, E.J., Young, I.R., Bydder, G.M.: Detection of subtle brain changes using subvoxel registration and subtraction of serial MR images. Journal of Computer Assisted Tomography 19(5), 677–691 (1995)
30. Heckemann, R.A., Hajnal, J.V., Aljabar, P., Rueckert, D., Hammers, A.: Automatic anatomical brain mri segmentation combining label propagation and decision fusion. Neuroimage 33(1), 115–126 (2006)
31. Heinrich, M.P., Jenkinson, M., Bhushan, M., Matin, T., Gleeson, F.V., Brady, S.M., Schnabel, J.A.: MIND: Modality independent neighbourhood descriptor for multi-modal deformable registration. Medical Image Analysis 16(7), 1423–1435 (2012)
32. Heinrich, M.P., Jenkinson, M., Brady, M., Schnabel, J.A.: MRF-based deformable registration and ventilation estimation of lung CT. IEEE Transcations on Medical Imaging (in press, 2013)
33. Hellier, P., Barillot, C., Mémin, É., Perex, P.: Hierarchical estimation of a dense deformation field for 3D robust registration. IEEE Transactions on Medical Imaging 20(5), 388–402 (2001)
34. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(3), 226–239 (1998)
35. Klein, S., Staring, M., Pluim, J.: Evaluation of optimization methods for nonrigid medical image registration using mutual information and B-splines. IEEE Transactions on Image Processing 16(12), 2879–2890 (2007)
36. Kybic, J., Unser, M.: Fast parametric elastic image registration. IEEE Transactions on Image Processing 12(11), 1427–1442 (2003)
37. Lee, S., Wolberg, G., Chwa, K.-Y., Shin, S.Y.: Image metamorphosis with scattered feature constraints. IEEE Transactions on Visualization and Computer Graphics 2(4), 337–354 (1996)

38. Lee, S., Wolberg, G., Shin, S.Y.: Scattered data interpolation with multilevel B-splines. IEEE Transactions on Visualization and Computer Graphics 3(3), 228–244 (1997)
39. Lee, S., Wolberg, G., Shin, S.Y.: Scattered data interpolation with multilevel B-splines. IEEE Transactions on Visualization and Computer Graphics 3(3), 228–244 (1997)
40. Lin, N., Duncan, J.S.: Generalized robust point matching using an extended free-form deformation model: Application to cardiac images. In: IEEE International Symposium on Biomedical Imaging (2004)
41. Loeckx, D., Maes, F., Vandermeulen, D., Suetens, P.: Nonrigid image registration using free-form deformations with a local rigidity constraint. In: Barillot, C., Haynor, D.R., Hellier, P. (eds.) MICCAI 2004. LNCS, vol. 3216, pp. 639–646. Springer, Heidelberg (2004)
42. MacCracken, R., Joy, K.I.: Free-form deformations with lattices of arbitrary topology. In: SIGGRAPH, pp. 181–188 (1996)
43. Maes, F., Collignon, A., Vandermeulen, D., Marechal, G., Suetens, R.: Multimodality image registration by maximization of mutual information. IEEE Transactions on Medical Imaging 16(2), 187–198 (1997)
44. Maintz, J.B.A., Viergever, M.A.: A survey of medical image registration. Medical Image Analysis 2(1), 1–36 (1998)
45. Mattes, D., Haynor, D.R., Vesselle, H., Lewellen, T.K., Eubank, W.: PET–CT image registration in the chest using free-form deformations. IEEE Transactions on Medical Imaging 22(1), 120–128 (2003)
46. Miller, M., Christensen, G.E., Amit, Y., Grenander, U.: Mathematical textbook of deformable neuroanatomies. Proc. Natl. Acad. Sci. USA 90, 11944–11948 (1993)
47. Modat, M., Ridgway, Z.A.T.G.R., Lehmann, M., Barnes, J., Hawkes, D.J., Fox, N.C., Ourselin, S.: Fast free-form deformation using graphics processing units. Computer Methods and Programs in Biomedicine 98(3), 278–284 (2010)
48. Rohlfing, T., Maurer, J.C.R.: Nonrigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees. IEEE Transactions on Information Technology in Biomedicine 7(1), 16–25 (2003)
49. Rohlfing, T., Maurer, J.C.R.: Multi-classifier framework for atlas-based image segmentation. Pattern Recognition Letters 26(13), 2070–2079 (2005)
50. Rohlfing, T., Maurer, J.C.R., Bluemke, D.A., Jacobs, M.A.: Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint. IEEE Transactions on Medical Imaging 22(6), 730–741 (2003)
51. Rousseau, F., Habas, P.A., Studholme, C.: A supervised patch-based approach for human brain labeling. IEEE Transactions on Medical Imaging 30(10), 1852–1862 (2011)
52. Rueckert, D., Hayes, C., Studholme, C., Summers, P., Leach, M.O., Hawkes, D.J.: Non-rigid registration of breast MR images using mutual information. In: Wells, W.M., Colchester, A.C.F., Delp, S.L. (eds.) MICCAI 1998. LNCS, vol. 1496, pp. 1144–1152. Springer, Heidelberg (1998)
53. Rueckert, D., Sonoda, L.I., Hayes, C., Hill, D.L.G., Leach, M.O., Hawkes, D.J.: Non-rigid registration using free-form deformations: Application to breast MR images. IEEE Transactions on Medical Imaging 18(8), 712–721 (1999)
54. Schnabel, J.A., Rueckert, D., Quist, M., Blackall, J.M., Castellano-Smith, A.D., Hartkens, T., Penney, G.P., Hall, W.A., Liu, H., Truwit, C.L., Gerritsen, F.A., Hill, D.L.G., Hawkes, D.J.: A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations. In: Niessen, W.J., Viergever, M.A. (eds.) MICCAI 2001. LNCS, vol. 2208, pp. 573–581. Springer, Heidelberg (2001)
55. Sederberg, T.W., Parry, S.R.: Free-form deformation of solid geometric models. SIGGRAPH 20(4), 151–160 (1986)

56. Shen, D., Davatzikos, C.: Hammer: hierarchical attribute matching mechanism for elastic registration. IEEE Transactions on Medical Imaging 21(11), 1421–1439 (2002)
57. Studholme, C., Constable, R.T., Duncan, J.S.: Incorporating an image distortion model in non-rigid alignment of EPI with conventional MRI. In: Kuba, A., Sámal, M., Todd-Pokropek, A. (eds.) IPMI 1999. LNCS, vol. 1613, pp. 454–459. Springer, Heidelberg (1999)
58. Studholme, C., Hill, D.L.G., Hawkes, D.J.: An overlap invariant entropy measure of 3D medical image alignment. Pattern Recognition 32(1), 71–86 (1998)
59. Thirion, J.-P.: Image matching as a diffusion process: An analogy with Maxwell's demons. Medical Image Analysis 2(3), 243–260 (1998)
60. Viola, P.: Alignment By Maximization of Mutual Information. PhD thesis, Massachusetts Institute of Technology. A.I. Technical Report No. 1548 (1995)
61. Viola, P., Wells, W.M.: Alignment by maximization of mutual information. In: Proc. 5th International Conference on Computer Vision (ICCV 1995), pp. 16–23 (1995)
62. Wahba, G.: Spline Models for Observational Data. Society for Industrial and Applied Mathematics (1990)
63. Wang, J., Jiang, T.: Nonrigid registration of brain mri using nurbs. Pattern Recognition Letters 28(2), 214–223 (2007)
64. Warfield, S.K., Zhou, K.H., Wells, W.M.: Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation. IEEE Transactions on Medical Imaging 23(7), 903–921 (2004)
65. Zitová, B., Flusser, J.: Image registration methods: a survey. Image Vision Comput. 21(11), 977–1000 (2003)

Clustering Games

Marcello Pelillo and Samuel Rota Bulò

Abstract. Clustering refers to the process of extracting maximally coherent groups from a set of objects using pairwise, or high-order, similarities. Traditional approaches to this problem are based on the idea of partitioning the input data into a predetermined number of classes, thereby obtaining the clusters as a by-product of the partitioning process. In this chapter, we provide a brief review of our recent work which offers a radically different view of the problem. In contrast to the classical approach, in fact, we attempt to provide a meaningful formalization of the very notion of a cluster and we show that game theory offers an attractive and unexplored perspective that serves well our purpose. To this end, we formulate the clustering problem in terms of a non-cooperative "clustering game" and show that a natural notion of a cluster turns out to be equivalent to a classical (evolutionary) game-theoretic equilibrium concept. We prove that the problem of finding the equilibria of our clustering game is equivalent to locally optimizing a polynomial function over the standard simplex, and we provide a discrete-time dynamics to perform this optimization, based on the Baum-Eagon inequality. Experiments on real-world data are presented which show the superiority of our approach over the state of the art.

1 Introduction

Clustering is the problem of organizing a set of objects into groups, or *clusters*, in such a way as to have similar objects grouped together and dissimilar ones assigned to different groups, according to some similarity measure (see, e.g., [40]). Recently, a wave of excitement has spread across the machine learning and computer vision communities around this problem mainly because of the important development of spectral methods [54]. At the same time, there is also growing interest around fundamental questions pertaining to the very nature of the clustering problem (see,

Marcello Pelillo · Samuel Rota Bulò
DAIS, Università Ca' Foscari Venezia, Via Torino, 155, Venezia Mestre, Italy
e-mail: {pelillo,srotabul}@dais.unive.it

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_8, © Springer-Verlag Berlin Heidelberg 2014

e.g., [44, 1, 76]). Yet, despite the tremendous progress in the field, the clustering problem remains elusive and a satisfactory answer even to the most basic questions is still to come.

Upon scrutinizing the relevant literature on the subject, it becomes apparent that the vast majority of the existing approaches deal with a very specific version of the problem, which asks for *partitioning* the input data into coherent classes. In fact, almost invariably, the problem of clustering is *defined* as a partitioning problem, and even the classical distinction between hierarchical and partitional algorithms [41] seems to suggest the idea that partitioning data is, in essence, what clustering is all about (as hierarchies are but nested partitions). This is unfortunate, because it has drawn the community's attention away from different, and more general, variants of the problem and has led people to neglect underdeveloped foundational issues. As J. Hartigan clearly put it more than a decade ago: "We pay too much attention to the details of algorithms. [...] We must begin to subordinate engineering to philosophy." [32, p. 3].

The partitional approach is attractive as it leads to elegant mathematical and algorithmic treatments and allows us to employ powerful ideas from such sophisticated fields as linear algebra, graph theory, optimization, statistics, information theory, etc. However, there are several reasons for feeling uncomfortable with this oversimplified formulation. Probably the best-known limitation of the partitional approach is the typical (algorithmic) requirement that the number of clusters be known in advance, but there is more than that.

To begin, the very idea of a partition implies that *all* the input data will have to get assigned to some class. This subsumes the old philosophical view which gives categories an *a priori* ontological status, namely that they exist independent of human experience, a view which has now been discredited by cognitive scientists, linguists, philosophers, and machine learning researchers alike (see, e.g., [46, 20, 31]). Further, there are various applications for which it makes little sense to force all data items to belong to some group, a process which might result either in poorly-coherent clusters or in the creation of extra spurious classes. As an extreme example, consider the classical figure/ground separation problem in computer vision which asks for extracting a coherent region (the figure) from a noisy background [34, 69]. It is clear that, due to their intrinsic nature, partitional algorithms have no chance of satisfactorily solving this problem, being, as they are, explicitly designed to partition all the input data, and hence the unstructured clutter items too, into coherent groups. More recently, motivated by practical applications arising in document retrieval and bioinformatics, a conceptually identical problem has attracted some attention within the machine learning community and is generally known under the name of one-class clustering [30, 19].

The second intrinsic limitation of the partitional paradigm is even more severe as it imposes that each element cannot belong to more than one cluster. There are a variety of important applications, however, where this requirement is too restrictive. Examples abound and include, e.g., clustering micro-array gene expression data (wherein a gene often participate in more than one process), clustering documents into topic categories, perceptual grouping, and segmentation of images with transparent surfaces. In fact, the importance of dealing with overlapping clusters has been recognized long ago [42] and recently, in the machine learning community, there has been a renewed interest around this problem [9, 33]. Typically, this is solved by relaxing the constraints imposed by crisp partitions in such a way as to have "soft" boundaries between clusters.

Finally, we would like to mention another restriction of current state-of-the-art approaches to clustering which, admittedly, is not caused in any direct way by the partitioning assumption but, rather, by the intrinsic nature of the technical tools used to attack the problem. Indeed, it is typically assumed that object similarities are expressed as pairwise relations, but in some applications, such as, for example, face clustering [4], perceptual grouping [29], parametric motion segmentation [29, 70], and image categorization [39], higher-order relations turn out to be more appropriate, and approximating them in terms of pairwise interactions can lead to substantial loss of information. As an illustrative example, taken from [4], consider the problem of grouping a given set of d-dimensional Euclidean points into lines. As every pair of data points trivially defines a line, there is no meaningful pairwise measure of similarity for this problem. However, it makes perfect sense to define similarity measures over *triplets* of points that indicate how close they are to being collinear. Clearly, this example can be generalized to any model fitting problem, where the deviation of a set of points from the model provides a measure of their dissimilarity. The problem of data clustering using high-order similarities is usually referred to as *hypergraph clustering*, since we can represent any instance of this problem by means of a hypergraph [15], where vertices are the objects to be clustered and the (weighted) hyperedges encode high-order similarities. Clearly, the classical pairwise (i.e., graph-based) clustering problem is but a special case of the hypergraph formulation. Recently there has been some interest around this problem in computer vision and machine learning (see, e.g., [77, 4, 70, 29]) but, again, all the approaches developed so far adhere to the partitional paradigm.[1]

In this chapter we review our recent work on clustering, which offers a radically different perspective to the problem [65, 66, 45]. Instead of insisting on the idea of determining a partition of the input data, and hence obtaining the clusters as a by-product of the partitioning process, we reverse the terms of the problem and attempt instead to derive a rigorous formulation of the very notion of a cluster. This allows one, in principle, to deal with more general problems where clusters may overlap and/or clutter points may get unassigned. Clearly, the *conceptual* question "what is a cluster?" is as hopeless, in its full generality, as is its companion "what is an *optimal* clustering?" which has dominated the literature in the past few decades, both being two sides of the same coin. An attempt to answer the former question, however, besides shedding fresh light into the nature of the clustering problem, would allow us, as a consequence, to naturally overcome the major limitations of the partitional approach alluded to above, and to deal with more general problems where, e.g., clusters may overlap and clutter elements may get unassigned, thereby hopefully reducing the gap between theory and practice.

[1] It is worth noting that many of these algorithms, though designed to deal with higher-order relations, can easily be reduced to the standard pairwise case, as shown in [3].

The starting point of our approach is the elementary observation that a "cluster" may be informally defined as a maximally coherent set of data items, i.e., as a subset of the input data C which satisfies both an *internal* criterion (all elements belonging to C should be highly similar to each other) and an *external* one (all elements outside C should be highly dissimilar to the ones inside). In our endeavor to provide a formal definition of such a notion, we found that game theory offers an elegant and general perspective that serves well our purposes. The basic idea behind our framework is that the clustering problem can be considered as a non-cooperative "clustering game." Within this context, the notion of a cluster turns out to be equivalent to a classical equilibrium concept from (evolutionary) game theory, as the latter reflects both the internal and external cluster conditions alluded to before. We also show that there exists a one-to-one correspondence between these equilibria and the local solutions of a linearly-constrained polynomial optimization problem, thereby generalizing the work described in [60]. This characterization allows us to employ a powerful class of dynamical systems to extract our clusters, based on the well-known Baum-Eagon inequality, which generalize classical (pairwise) replicator dynamics [75, 35] from evolutionary game theory to higher-order interactions. A distinguishing feature of our approach is that, unlike standard partitional techniques, we do not need to know the number of clusters is advance as we extract them sequentially. Experiments on various synthetic and real-world problems show the superiority of the proposed approach over state-of-the-art techniques.

In the sequel, to keep the discussion as general as possible, we shall focus on the hypergraph version of the problem. Interestingly, note that the graph (i.e., pairwise) version turns out to be equivalent to the dominant-set notion of a cluster introduced in [59, 60], for which no game-theoretic interpretation was originally given.

2 Notions from Evolutionary Game Theory

According to classical game theory [25], a game of strategy can be formalized as a triplet $\Gamma = (P, S, \pi)$, where $P = \{1, \ldots, k\}$ is a set of $k \geq 2$ "players" (or agents), $S = \{1, \ldots, n\}$ is a set of *pure strategies* (or actions) available to each player, and $\pi : S^k \to \mathbb{R}$ is a *payoff function*, which assigns a utility to each *strategy profile* $\mathbf{s} = (s_1, \ldots, s_k) \in S^k$, which is an (ordered) set of pure strategies played by the different players.[2] A game Γ is *super-symmetric* if its payoff function is super-symmetric, i.e., if it is invariant under permutations of the strategy profile. In the sequel we will deal only with such games and therefore we assume π to be super-symmetric.

Evolutionary game theory originated in the early 1970's as an attempt to apply the principles and tools of game theory to biological contexts, with a view to model the evolution of animal, as opposed to human, behavior (see the classical work by J. Maynard Smith [55] who pioneered the field). It considers an idealized scenario whereby individuals are repeatedly drawn at random from a large, ideally infinite,

[2] We note that although we restrict ourselves to games where all players share the same set of pure strategies and payoff function, in more general settings each agent can well be associated to its own pure strategy set and payoff function.

population to play a game $\Gamma = (P, S, \pi)$. In contrast to classical game theory, here players are not supposed to behave rationally or to have complete knowledge of the details of the game. They act instead according to an inherited behavioral pattern, or pure strategy, and it is supposed that some evolutionary selection process operates over time on the distribution of behaviors. Here, and in the sequel, an agent with preassigned strategy $j \in S$ will be called *j-strategist*. The state of the population at a given time t can be represented as a n-dimensional vector $\mathbf{x}(t)$, where $x_j(t)$ represents the fraction of j-strategists in the population at time t. Hence, the initial distribution of preassigned strategies in the population is given by $\mathbf{x}(0)$. The set of all possible states describing a population is given by

$$\Delta = \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{j \in S} x_j = 1 \text{ and } x_j \geq 0 \text{ for all } j \in S \right\}$$

which is called *standard simplex*. As time passes, the distribution of strategies in the population changes under the effect of a selection mechanism which, by analogy with Darwinian process, aims at spreading the fittest strategies in the population to the detriment of the weakest one which, in turn, will be driven to extinction (we postpone the formalization of one such selection mechanism to Section 4). For notational convenience, we drop the time reference t from a population state and we refer to $\mathbf{x} \in \Delta$ as a population rather than population state. Moreover, we denote by $\sigma(\mathbf{x})$ the *support* of $\mathbf{x} \in \Delta$:

$$\sigma(\mathbf{x}) = \{j \in S : x_j > 0\}$$

which is the set of strategies that are alive in a given population $\mathbf{x}$.

We will find it useful to define the following function $u : \Delta^k \to \mathbb{R}$:

$$\mathbf{U}\mathbf{y}^{(1)}, \dots, \mathbf{y}^{(k)} = \sum_{(s_1, \dots, s_k) \in S^k} \pi(s_1, \dots, s_k) \prod_{i=1}^{k} y_{s_i}^{(i)}, \tag{1}$$

which is invariant under any permutation of its arguments due to the super-symmetry of the payoff function π. Also, we will use the notations $\mathbf{x}^{[k]}$ as a shortcut for a sequence $(\mathbf{x}, \dots, \mathbf{x})$ of k identical states $\mathbf{x}$, and $\mathbf{e}^j$ to indicate the n-vector with $x_j = 1$ and zero elsewhere. Now, it is easy to see that the expected payoff earned by a j-strategist ($j \in S$) in a population $\mathbf{x} \in \Delta$ is given by $\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]}$, while the expected payoff over the entire population is given by $\mathbf{U}\mathbf{x}^{[k]}$.

A fundamental notion in game theory is that of an equilibrium [75]. Intuitively, an evolutionary process reaches an equilibrium $\mathbf{x} \in \Delta$ when every individual in the population obtains the same expected payoff and no strategy can thus prevail upon the other ones. Formally, $\mathbf{x} \in \Delta$ is a *Nash equilibrium* if

$$\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]} \leq \mathbf{U}\mathbf{x}^{[k]}, \qquad \text{for all } j \in S. \tag{2}$$

In other words, at a Nash equilibrium every agent in the population performs at most as well as the overall population expected payoff. Within a population-based setting, however, the notion of a Nash equilibrium turns out to be too weak as it lacks stability under small perturbations. This motivated J. Maynard Smith, in his seminal work [55], to introduce a refinement of the Nash equilibrium concept generally known as an Evolutionary Stable Strategy (ESS). His original work involved pairwise interactions (two-player games), but his notion has later been generalized to multi-player games [17]. Formally, assume that in a population $\mathbf{x} \in \Delta$, a small share ε of mutant agents appears, whose distribution of strategies is $\mathbf{y} \in \Delta$. The resulting post-entry population is then given by $\mathbf{w}_\varepsilon = (1-\varepsilon)\mathbf{x} + \varepsilon\mathbf{y}$. Biological intuition suggests that evolutionary forces select against mutant individuals if and only if the expected payoff of a mutant agent in the postentry population is lower than that of an individual from the original population, i.e.,

$$\mathbf{U}\mathbf{y}, \mathbf{w}_\varepsilon^{[k-1]} < \mathbf{U}\mathbf{x}, \mathbf{w}_\varepsilon^{[k-1]} . \tag{3}$$

Hence, a population $\mathbf{x} \in \Delta$ is said to be *evolutionary stable* if inequality (3) holds for any distribution of mutant agents $\mathbf{y} \in \Delta \setminus \{\mathbf{x}\}$, granted the population share of mutants ε is sufficiently small. It can be shown that an ESS is a refinement of the notion of a Nash equilibrium in the sense that every ESS is necessarily a Nash equilibrium (see, [75] for pairwise contests and [17] for k-wise contests).

3 Clustering as a Non-cooperative Game

For the sake of generality, we shall focus on the hypergraph version of the clustering problem, whereby high-order similarities among objects are involved, the classical pairwise (i.e., graph-based) case being but a special instance of this general formulation.

An instance of the hypergraph clustering problem can be described by an edge-weighted hypergraph [15], which is formally defined as a triplet $H = (V, E, \omega)$, where $V = \{1, \ldots, n\}$ is a finite set of *vertices*, $E \subseteq 2^V \setminus \{\emptyset\}$ is the set of (hyper-)edges (here, 2^V is the power set of V), and $\omega : E \to \mathbb{R}_+$ is a real-valued function, which assigns a positive weight to each edge. Within our clustering framework the vertices in H correspond to the objects to be clustered, the edges represent (possibly) high-order neighborhood relationships among objects, and the edge-weights reflect similarity among linked objects. Although hypergraphs may have edges of varying cardinality, in this paper we will focus on a particular class of hypergraphs, called k-graphs, whose edges have fixed cardinality $k \geq 2$ (clearly, if $k = 2$ we get back to the standard notion of a graph). Note that for simplicity, here we restrict ourselves to positive similarities, although the proposed framework can easily be generalized to deal with negative weights as well.

Given a weighted k-graph $H = (V, E, \omega)$, representing an instance of a hypergraph clustering problem, we cast it into a k-player *(hypergraph) clustering game* $\Gamma = (P, V, \pi)$ where the players' pure strategies correspond to the objects to be clustered and the payoff function π is proportional to the similarity of the objects/strategies $(v_1,\dots,v_k) \in V^k$ selected by the players:

$$\pi(v_1,\dots,v_k) = \begin{cases} \frac{1}{k!}\omega(\{v_1,\dots,v_k\}) & \text{if } \{v_1,\dots,v_k\} \in E\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{4}$$

Here, the constant of proportionality $1/k!$ has been chosen to simplify later algebraic derivations.

Our clustering game will be played within an evolutionary setting wherein the k players, each of which is assumed to play a pre-assigned strategy, are repeatedly drawn at random from a large population. Here, given a population $\mathbf{x} \in \Delta$, x_j $(j \in V)$ represents the fraction of players that is programmed to select j from the objects to be clustered. A dynamic evolutionary selection process, as the one described in the next section, will then make the population $\mathbf{x}$ evolve according to a Darwinian survival-of-the-fittest principle in such a way that, eventually, the better-than-average objects will survive and the others will get extinct. It is clear that the whole dynamical process is driven by the payoff function π which, in our case, has been defined in (4) precisely to favor the evolution of highly coherent objects. Accordingly, the support $\sigma(\mathbf{x})$ of the converged population $\mathbf{x}$ does represent a cluster, the non-null components of $\mathbf{x}$ providing a measure of the degree of membership of its elements. Indeed, the expected population payoff $\mathbf{U}\mathbf{x}^{[k]}$ can be regarded to as a measure of the cluster's internal coherency in terms of the average similarity of the objects forming the cluster, whereas the expected payoff $\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]}$ of a player selecting object $j \in V$ in $\mathbf{x}$ measures the average similarity of object j with respect to the cluster.

We claim that, within this setting, the clusters of a hypergraph clustering problem instance can be characterized in terms of the ESS's of the corresponding (evolutionary) clustering game, thereby justifying the following definition.

Definition 0.1 (ESS-cluster). Given an instance of a hypergraph clustering problem $H = (V, E, \omega)$, an *ESS-cluster* of H is an ESS of the corresponding hypergraph clustering game.

For the sake of simplicity, when it will be clear from context, the term ESS-cluster will be used henceforth to refer to either the ESS itself, namely the membership vector $\mathbf{x} \in \Delta$, or to its support $\sigma(\mathbf{x}) = C \subseteq V$.

The motivation behind the above definition resides in the observation that ESS-clusters do incorporate the two basic properties of a cluster, i.e.,

- *internal coherency*: elements belonging to the cluster should have high mutual similarities;
- *external incoherency*: the overall cluster internal coherency decreases by introducing external elements.

The rest of this section is devoted to provide support to this claim.

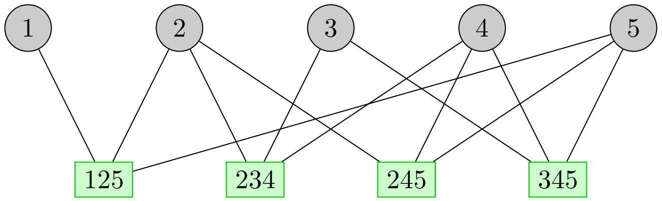


Fig. 1 Example of a 3-graph with 5 nodes (circles) and 4 edges (rectangles), represented as a bipartite graph. Each edge is connected to the vertices it contains.

3.1 Internal Coherency

The internal coherency of an ESS-cluster is a direct consequence of the Nash condition (2), which is satisfied by any ESS. Indeed, if $\mathbf{x} \in \Delta$ is an ESS of a clustering game, then from (2) it follows that every object belonging to the cluster, i.e., every object in $\sigma(\mathbf{x})$, has the same average similarity with respect to the cluster, which in turn corresponds to the cluster's overall average similarity. This is formally stated in the following theorem.

Theorem 0.1. *Let $H = (V, E, \omega)$ be an instance of a hypergraph clustering problem, and $\Gamma = (P, V, \pi)$ the corresponding clustering game. If $\mathbf{x} \in \Delta$ is an ESS-cluster of H, with support $\sigma(\mathbf{x}) = C$, then*

$$\mathbf{U}\mathbf{e}^{j}, \mathbf{x}^{[k-1]} = \mathbf{U}\mathbf{x}^{[k]}\,, \qquad \textit{for all } j \in C. \tag{5}$$

Proof. See [66].

The internal coherency of an ESS-cluster becomes clearer if we analyze it using a notion from hypergraph theory. Let $H = (V, E, \omega)$ be a (weighted) hypergraph and $C \subseteq V$. We say that C is a *two-cover* of H if for any pair of vertices $\{j, \ell\} \subseteq C$ there exists an edge $e \in E$ such that $\{j, \ell\} \subseteq e \subseteq C$. Note that if H is a graph (i.e., $k = 2$) then two-covers correspond to cliques, namely, to sets of mutually adjacent vertices. To illustrate, in the hypergraph shown in Figure 1 the sets $\{1,2,3,4,5\}$ and $\{1,2,4,5\}$ are not two-covers as there is no edge contained in them connecting vertices $\{1,3\}$ and $\{1,4\}$, respectively, while the set $\{2,3,4,5\}$ is a two-cover.

The following proposition, which is a weighted counterpart of a result by Frankl and Rödl [24] on unweighted hypergraphs, provides an interesting connection between ESS's and two-covers.

Proposition 0.1. *Let $H = (V, E, \omega)$ be an instance of a hypergraph clustering problem, and $\Gamma = (P, V, \pi)$ the corresponding clustering game. If $\mathbf{x} \in \Delta$ is an ESS-cluster of H, then its support $\sigma(\mathbf{x})$ is a two-cover of H.*

Proof. See [66].

Intuitively, the previous result shows that two objects cannot belong to (the support of) an ESS-cluster if there is no similarity relationship between them within the cluster. This is a minimal property that a cluster should satisfy in order to guarantee some form of internal coherency.

3.2 External Incoherency

In addition to the internal coherency property described above, we now show that ESS-clusters satisfy also a property of external incoherency. This follows, in the first place, from the Nash condition (2) that we already used to prove internal coherency. In fact, according to (2), every object external to an ESS-cluster C has an average similarity with respect to C that cannot exceed the cluster's overall similarity. More formally, if $\mathbf{x} \in \Delta$ is a Nash equilibrium with support $\sigma(\mathbf{x}) = C$, we have

$$\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]} \leq \mathbf{U}\mathbf{x}^{[k]}, \qquad \text{for all } j \notin C.$$

However, the Nash condition alone is not enough, as there may still be cases where the average similarity of an external object equals the cluster's overall similarity, thereby violating the external incoherency criterion. As it turns out, to some extent, this cannot be the case with an ESS, thanks to its additional stability properties.

Theorem 0.2. *Let $H = (V, E, \omega)$ be an instance of a hypergraph clustering problem, and $\Gamma = (P, V, \pi)$ the corresponding clustering game. Then, $\mathbf{x} \in \Delta$ is an ESS-cluster of H if and only if for any $\mathbf{y} \in \Delta \setminus \{\mathbf{x}\}$ and all sufficiently small positive values of ε the following inequality holds:*

$$\mathbf{U}\mathbf{w}_\varepsilon^{[k]} < \mathbf{U}\mathbf{x}^{[k]},$$

where $\mathbf{w}_\varepsilon = (1-\varepsilon)\mathbf{x} + \varepsilon\mathbf{y}$.

Proof. See [66].

The previous theorem asserts that whenever we try to deviate from an ESS-cluster $\mathbf{x} \in \Delta$, e.g., by adding an external element to its support, the cluster's overall average similarity drops, provided that deviation is not too large. This not only guarantees a form of external incoherency, but provides also support to the claim that the components of $\mathbf{x}$ reflect the degree of cluster membership.

Observe that when the number of players k equals 2, i.e., in the presence of pairwise similarities, our notion of ESS-cluster coincides with that of a dominant set [60, 73], which is a generalization of a maximal clique to the case of edge-weighted graphs. In this case a stronger notion of external incoherency holds, which asserts that no dominant set can be a subset of another. In the case of higher-order similarities, however, there is no theoretical guarantee that the support of an ESS is not contained in that of another one. Indeed, in [17] it is shown that such solution patterns might possibly appear in general games with more than two players (i.e.,

$k > 2$). However, this behaviour has never been observed in practice (e.g., on the instances used in the experiments described below).

4 Evolution Towards a Cluster

In this section, we address the issue of determining an ESS-cluster for a given instance of a hypergraph clustering problem. Unfortunately, this turns out to be a computationally hard problem [21, 57], but good heuristics do exist. Indeed, we show below that the ESS's of a clustering game are in one-to-one correspondence with (strict) local solutions of a non-linear optimization problem, thereby allowing the use of standard optimization techniques.

Theorem 0.3. *Let $H = (V, E, \omega)$ be a hypergraph clustering problem, $\Gamma = (P, V, \pi)$ the corresponding clustering game, and $f(\mathbf{x})$ a function defined as*

$$f(\mathbf{x}) = \mathbf{U}\mathbf{x}^{[k]} = \sum_{e \in E} \omega(e) \prod_{j \in e} x_j . \tag{6}$$

Nash equilibria of Γ are in one-to-one correspondence with the critical points [3] *of $f(\mathbf{x})$ over Δ, while ESS's of Γ are in one-to-one correspondence with strict local maximizers of $f(\mathbf{x})$ over Δ.*

Proof. See [66].

The problem of extracting ESS-clusters can thus be cast into the problem of finding a strict local solutions of (6) in Δ. We will address this optimization task using a well-known result due to Baum and Eagon [11], who introduced a wide class of nonlinear transformations in probability domain. Their result generalizes an earlier one by Blakley [16], who discovered similar properties for certain homogeneous quadratic transformations. The next theorem introduces what is known as the Baum-Eagon inequality.[4]

Theorem 0.4 (Baum-Eagon). *Let $Q(\mathbf{x})$ be a homogeneous polynomial in the variables x_j with nonnegative coefficients, and let $\mathbf{x} \in \Delta$. Define the mapping $\mathbf{z} = \mathscr{M}(\mathbf{x})$ from Δ to itself as follows:*

$$z_j = x_j \frac{\partial Q(\mathbf{x})}{\partial x_j} \Big/ \sum_{\ell=1}^{n} x_\ell \frac{\partial Q(\mathbf{x})}{\partial x_\ell}, \qquad j = 1, \ldots, n. \tag{7}$$

Then $Q(\mathscr{M}(\mathbf{x})) > Q(\mathbf{x})$, unless $\mathscr{M}(\mathbf{x}) = \mathbf{x}$.

[3] A point $\mathbf{x}$ is said to be a critical (or a KKT) point of an optimization problem if it satisfies the first-order necessary conditions for being a solution [53].

[4] Indeed, the original Baum-Eagon inequality is more general than presented here as it deals with a maximization problem over a product of simplices.

Although this result applies to homogeneous polynomials, in a subsequent paper Baum and Sell [13] proved that Theorem 0.4 still holds in the case of arbitrary, non-homogeneous polynomials and further extended the result by showing that $\mathscr{M}$ increases Q homotopically, which means that for all $0 \leq \eta \leq 1$, $Q(\eta\mathscr{M}(\mathbf{x}) + (1-\eta)\mathbf{x}) \geq Q(\mathbf{x})$ with equality if and only if $\mathscr{M}(\mathbf{x}) = \mathbf{x}$.

Another way of looking at Theorem 0.4 is from the standpoint of dynamical systems theory [52, 48]. The nonlinear operator $\mathscr{M}$ defines in fact a discrete dynamical system and it is therefore of particular interest to study how it behaves in the vicinity of its equilibrium points. In the theory of dynamical systems this is formalized by the concept of stability. An equilibrium point $\mathbf{x}$ is said to be *stable* if, whenever started sufficiently close to $\mathbf{x}$, the system will remain near to $\mathbf{x}$ for all future times. A stronger property, which is even more desirable, is that the equilibrium point $\mathbf{x}$ be *asymptotically stable*, meaning that $\mathbf{x}$ is stable and in addition is a *local attractor*, i.e., when initiated close to $\mathbf{x}$, the system tends towards $\mathbf{x}$ as time increases. One of the most fundamental tools for establishing the stability of a given equilibrium point is known as Lyapunov's direct method. It involves seeking a so-called *Lyapunov* function, i.e., a continuous real-valued function defined in state space which is non-decreasing along any trajectory. Of particular interest are *strict* Lyapunov functions which are, instead, strictly increasing along non-constant trajectories. Accordingly, Theorem 0.4 states essentially that the polynomial Q is a Lyapunov function for the discrete-time dynamical system defined by $\mathscr{M}$.

The Baum-Eagon inequality provides therefore an effective iterative means for maximizing polynomial functions in probability domains, and in fact it has served as the basis for various statistical estimation techniques developed within the theory of probabilistic functions of Markov chains [12]. It has also been employed for studying the dynamical properties of relaxation labelling processes [61]. Note that, even in the presence of negative coefficients, it is still possible to use the Baum-Eagon theorem, and hence the corresponding dynamical system, by applying a simple transformation to the original polynomial which does preserve the original solutions. This could be useful, for example, when the edge-weights in the hypergraph encode both similarity and dissimilarity information.

Now, let us go back to our clustering problem. Note that the function f defined in (6) is precisely a homogeneous polynomial with nonnegative coefficients and hence the Baum-Eagon theorem applies. In this case, we have

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{1}{k}\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]}, \qquad j = 1 \ldots n$$

which yields

$$\sum_{\ell=1}^{n} x_\ell \frac{\partial f(\mathbf{x})}{\partial x_\ell} = \frac{1}{k}\mathbf{U}\mathbf{x}^{[k]}$$

so that the proposed discrete-time dynamics to extract an ESS-cluster takes the following form:

$$x_j(t+1) = x_j(t)\frac{\mathbf{U}\mathbf{e}^j, \mathbf{x}(t)^{[k-1]}}{\mathbf{U}\mathbf{x}(t)^{[k]}}, \qquad j = 1 \dots n. \tag{8}$$

This dynamics can be given a natural evolutionary interpretation, and in fact generalizes a classical formalization of natural selection processes in two-player evolutionary game theory [75, 35], known as "replicator dynamics." To see this, recall that $\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]}$ represents the expected payoff of an i-strategiest in population $\mathbf{x}$, while $\mathbf{U}\mathbf{x}^{[k]}$ represents the expected payoff over the entire population. Hence, during the evolution of (8), better-than-average strategies, i.e., those satisfying $\mathbf{U}\mathbf{e}^j, \mathbf{x}^{[k-1]} > \mathbf{U}\mathbf{x}^{[k]}$, will spread in the population while the others will get extinct, giving therefore rise to a Darwinian selection process.

From Theorem 0.4 we can assert that f is a strict Lyapunov function for this dynamical system and this, in conjunction with the fact that every ESS-cluster is a strict local maximizer of f in Δ, proves the following theorem which is an obvious consequence of Lyapunov's theorem of asymptotically stability [52, 48].

Theorem 0.5. *A point* $\mathbf{x} \in \Delta$ *is an ESS-cluster of an instance of a hypergraph clustering problem if and only if it is an asymptotically stable equilibrium point (and, hence, a local attractor) for the nonlinear dynamics* (8).

In practical applications, without heuristic information about the optimal solution, it is customary to start out the dynamics from the barycenter of the simplex, i.e., from the vector $\mathbf{x}(0) = (\frac{1}{n}, \dots, \frac{1}{n})^\top \in \Delta$, which is the uniform distribution over the set of vertices V. This choice ensures that no particular solution is favored. Moreover, the dynamics (13) satisfies the invariant property $\sigma(\mathbf{x}(t)) \subseteq \sigma(\mathbf{x}(0))$ for any time $t > 0$. Hence, in order to allow any vertex $i \in V$ to potentially take part of a solution, we need to select an initial state $\mathbf{x}(0)$ with full support, i.e., $\sigma(\mathbf{x}(0)) = V$. In particular, if the numerator of (13) is positive for all $j \in \sigma(\mathbf{x}(0))$ then $\sigma(\mathbf{x}(t)) = \sigma(\mathbf{x}(0))$ for all *finite* values of $t \geq 0$ and only asymptotically we might possibly have $\sigma(\mathbf{x}^*) \subsetneq \sigma(\mathbf{x}(0))$, $\mathbf{x}^*$ being the limit point of the trajectory, namely $\mathbf{x}^* = \lim_{t\to\infty} \mathbf{x}(t)$. This fact suggests that given a solution $\mathbf{x}(T)$ obtained after $T < \infty$ steps of (13), we need to threshold its components in order to get the support of the corresponding ESS-cluster. Observe also that the components of an ESS-cluster $\mathbf{x}$ provide information about the degree of membership of each element to the cluster (which could be useful, e.g., to extract a representative of the cluster found).

Unlike standard partitional techniques, our approach involves extracting one cluster at a time, much in the same spirit as [60, 63, 68]. Depending on the application at hand, one might want to obtain either overlapping or non-overlapping clusters. In the latter case, a simple, yet effective "peel-off" strategy, which has also been used in the experiments reported below, can be as follows: 1) Find an ESS-cluster with dynamics (8); 2) remove its vertices from the hypergraph; 3) reiterate on the remaining vertices. Alternatively, in order to extract overlapping groups one needs to enumerate the ESS-clusters. In this paper we do not address this issue, but we mention that a possible strategy to accomplish this has been proposed in [74], although restricted to the standard pairwise case.

Finally, as pointed out in [13], note that our dynamics (8) contrasts sharply with gradient methods, for which an increase in the objective function is guaranteed only when infinitesimal steps are taken, and determining the optimal step size entails computing higher-order derivatives. We add that performing gradient ascent in Δ requires some projection operator to ensure that the constraints not be violated, and this might cause some problems for points lying on the boundary [23, 56]. In (8), instead, a computationally simple normalization is required. Overall, the complexity of finding an ESS-cluster with our algorithm turns out to be $O(\rho|E|)$, where $|E|$ is the number of edges of the hypergraph and ρ is the average number of iteration needed to converge. In the experiments reported below ρ never exceeded 100. More efficient algorithms to extract ESS-clusters can well be developed, e.g., along the lines suggested in [67, 62] for quadratic optimization.

5 Experiments

To test the effectiveness of the proposed approach, we conducted experiments on synthetic data as well as real-world applications. As for the synthetic experiments, we address the problem of line clustering in high-dimensional space in Section 5.1 and the problem of model-based 3D point-pattern matching in Section 5.2. Real-world experiments have been conducted in Section 5.3 on the problem of object detection in images (in particular, car and pedestrian detection), and on illuminant-invariant face clustering in Section 5.4. All the experiments, except the one on object-detection exploit similarities of order $k > 2$, whereas the the latter relies on 2-graphs. The experimental setting used for the object detection task is described separately in Section 5.3. Here, we describe the experimental setting used in all other cases.

We compared our approach against two of the most powerful hypergraph clustering algorithms available in the literature, namely the Clique Averaging algorithm (CAVERAGE) of Agarwal et al. [4], and the Super-symmetric Non-negative Tensor Factorization (SNTF) of Shashua et al. [70]. Note that in [4] CAVERAGE was shown to outperform consistently several existing hypergraph clustering techniques such as Clique Expansion combined with Normalized cuts [38], Gibson's Algorithm under sum and product model [28], the two-phase multi-level algorithm kHMeTiS [43], and therefore we decided not to include them in our experimental comparisons. Note also that, unlike CAVERAGE, which resorts to a pairwise approximation of the high-order similarity function, SNTF works directly on the hypergraph as we do.

Since both CAVERAGE and SNTF, in contrast to our method, require as a parameter the number of clusters K, we run them with values of $K \in \{K^* - 1, K^*, K^* + 1\}$, where K^* denotes the correct number of clusters. As in practical application the optimal number of clusters is not known in advance, this allowed us to assess the robustness of the approaches in the presence of under- and over- estimation of the correct number of clusters.[5] As concerns the other free parameters of all competing algorithms, they were optimally tuned using a small validation set, which consisted of a set of labeled observations sampled from the same distribution as that used in the testing phase. As for our algorithm, we used the peel-off strategy described in the previous section. The quality of the clusterings found by the different algorithms was evaluated in terms of classification error with minimum-cost bipartite matching except for the experiment in Section 5.3, where a different evaluation protocol has been adopted (see the description in the section).

We run the experiments on an AMD Sempron 3Ghz computer equipped with a 4Gb RAM. In the case of CAVERAGE and SNTF we used the original codes as provided by the authors (Matlab and C++ implementations, respectively). For our algorithm, we used a non-optimized Matlab implementation. As for running time, we report that our algorithm typically took a hundred seconds or so to converge, CAVERAGE was an order of magnitude faster, while SNTF was an order of magnitude slower. This is indeed to be expected as CAVERAGE, unlike our algorithm and SNTF, transforms at the outset the original hypergraph into a graph, thereby greatly reducing the complexity of the problem. On the other hand, like our algorithm, SNTF does not resort to any graph approximation but, by optimizing a single variable at a time, it has a substantially larger computational complexity.

5.1 Line Clustering

Here, we consider the problem of clustering lines in spaces of dimension greater than two, i.e., given a set of points in $\mathbb{R}^d$, the task is to extract subsets of collinear points. This is a typical example where classical pairwise approaches cannot work because any pair of points defines a straight line, and hence higher-order similarity relations are needed (see, e.g., [4]). An obvious ternary similarity measure for this clustering problem can be defined as follows. Given a triplet of points $\{i,j,k\}$ and its best fitting line ℓ, we calculate the mean distance $d(i,j,k)$ between each point and ℓ, and then we obtain a similarity function using a standard Gaussian kernel: $\omega(\{i,j,k\}) = \exp(-\beta d(i,j,k)^2)$, where β is a properly tuned precision parameter.

In order to assess the robustness of the competing approaches to both local and global perturbations, we conducted two kinds of experiments. In the first set of experiments we generated a few lines (from 3 to 5) in a 5-dimensional space $[-2,2]^5$. Each line consisted of 20 points, which were locally perturbed using a varying amount of Gaussian noise, namely from $\sigma = 0$ to $\sigma = 0.08$ (see Figure 2(a) for a specific example). Figure 2(b–d) shows the results obtained by the competing algorithms in terms of classification error with 3,4 and 5 lines, respectively, as a function of the noise level. Each plot shows the average performance obtained over 30 randomly generated instances together with the corresponding standard deviations.

[5] Note that running any clustering algorithm with $K < K^*$ prevents it from achieving perfect results. However, we think that the experiments presented with $K = K^* - 1$ do indeed provide some interesting information concerning the algorithms' behavior.

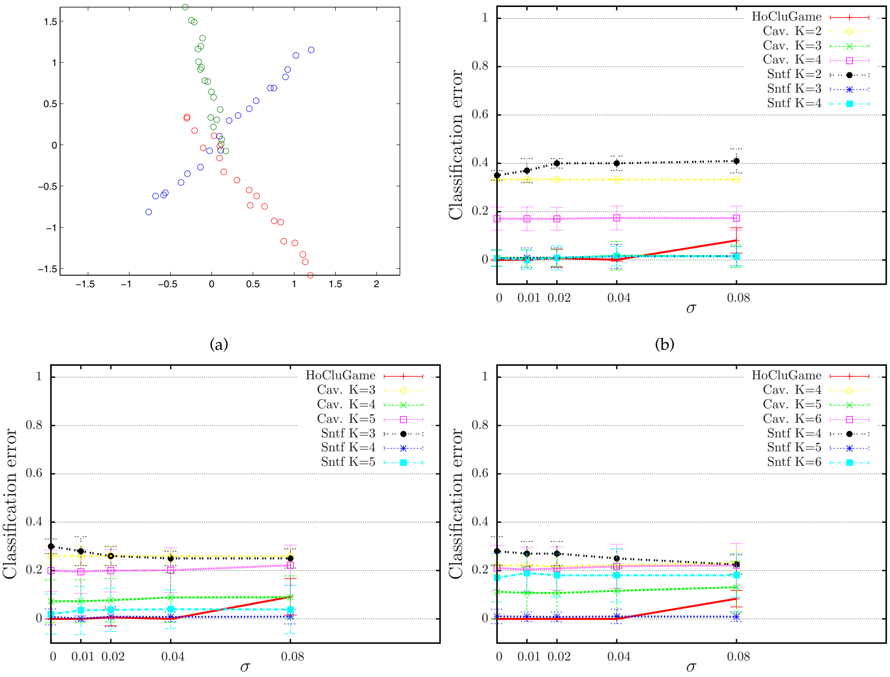
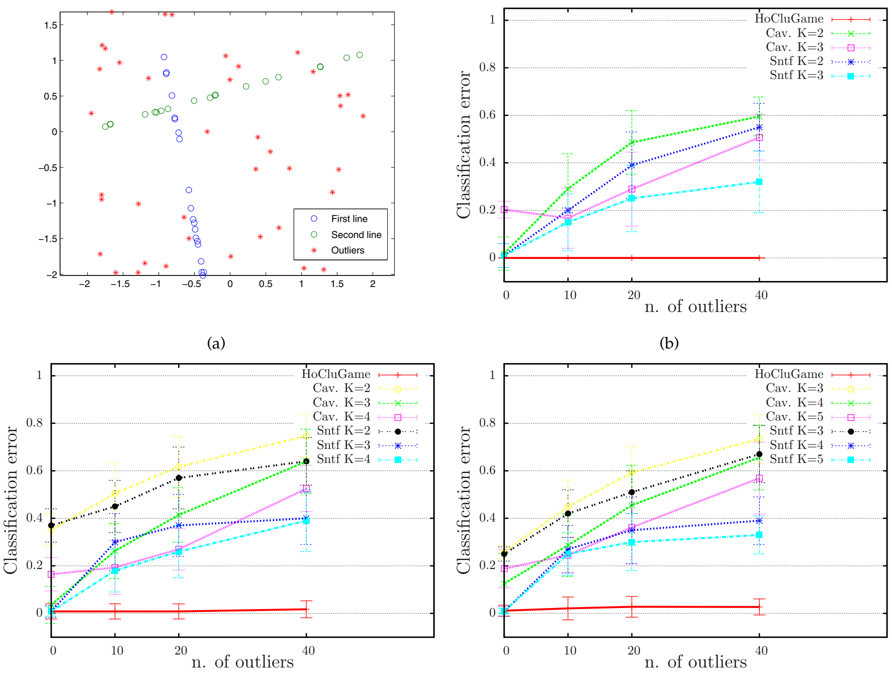


Fig. 2 Results of clustering 3, 4 and 5 lines perturbed locally with increasing levels of Gaussian local noise ($\sigma = 0, 0.01, 0.02, 0.04, 0.08$). (a) Example of three 5D lines (projected in 2D), perturbed with $\sigma = 0.04$. (b) Three lines. (c) Four lines. (d) Five lines.

In the first place, note that our algorithm performs essentially as well as as the best performing parametrization of SNTF on all instances with a level of noise not exceeding 0.04. As for CAVERAGE, note that even using the correct number of clusters $K = K^*$, its performances gradually deteriorate as the number of lines is increased. In all cases, both SNTF and CAVERAGE are systematically outperformed by our algorithm when they are run with a non-optimal value of K. We also observe that when $K = K^* - 1$, the error of CAVERAGE and SNTF is expected to decrease significantly as we increase K^*, e.g., when we use five instead of four lines, while this does not happen thereby suggesting that they do not achieve the best possible result here. Further, as expected, the influence of local noise on their performance is typically negligible. Indeed, this makes intuitively sense as, once they stick to a partition of the original input data, it is unlikely that the result will change drastically under moderate local perturbations. On the other hand, our approach appears to be slightly more vulnerable to local perturbations as points deviating too much from a cluster's average collinearity will get excluded, by construction, as they undermine internal coherency.



(a) (b)

(c) (d)

Fig. 3 Results of clustering $2,3$ and 4 lines with an increasing number of clutter points $(0,10,20,40)$. (a) Example of two 5D lines (projected in 2D) with 40 clutter points. (b) Two lines. (c) Three lines. (d) Four lines.

The second series of experiments aimed at assessing robustness to clutter (global noise). To this end, we randomly generated a few lines (in our experiments, from 2 to 4) in the 5-dimensional hypercube $[-2,2]^5$, and then added from 0 to 40 clutter points uniformly drawn from the hypercube (see Figure 3(a) for a specific example). In order to make the setting more realistic, we also slightly perturbed the original lines using a local Gaussian noise with standard deviation 0.01 . As in the previous set of experiments, each generated line consisted of 20 points.

Figure 3(b–d) shows the results obtained by all algorithms as a function of clutter. As can be clearly seen, our algorithm substantially outperformed both CAVERAGE and SNTF even when they were fed by the correct number of clusters K^*, and it worked almost perfectly irrespective of the clutter level. Note also that both competitors achieved better performances when $K > K^*$, and this is intuitively clear as the only way to get rid of clutter points is to group them into additional (garbage) clusters. Nevertheless, due to the intrinsic unstructured nature of clutter points, they typically did not get assigned to the garbage class but, instead, were associated to the original groups, thereby making the performance of CAVERAGE and SNTF poorer and poorer as clutter increases.

5.2 Model-Based 3D Point-Pattern Matching

We present here a different type of experiment, which highlights the advantages of our approach over the existing partition-based ones. We consider the problem of finding in a scene (possibly multiple) copies of a reference 3D model subject to a similarity transformation (i.e., rescaling + rotation + translation). Here, both the model and the scene are represented as clouds of 3D points. Motivated by the approach described in [7] (which deals with pairwise relations only), here we tackle this problem from a hypergraph clustering perspective.

Let $\mathscr{M}$ be a set of 3D points representing the model to be found and let $\mathscr{S}$ be a set of 3D points representing the scene. We denote by $\mathscr{A}$ the set of all possible pointwise correspondences between model and scene points, i.e., $\mathscr{A} = \mathscr{M} \times \mathscr{S}$. Given a set of three correspondences $e = \{(\mathbf{m}_1, \mathbf{s}_1), (\mathbf{m}_2, \mathbf{s}_2), (\mathbf{m}_3, \mathbf{s}_3)\} \subseteq \mathscr{A}$, we compute the similarity transformation T which minimizes the least-square error $d(e) = \sum_{i=1}^{|e|} \|T(\mathbf{m}_i) - \mathbf{s}_i\|^2$ using the Horn method [36]. Consider now the hypergraph $H = (\mathscr{A}, \mathscr{E}, \phi)$ where the set of vertices is given by the set of correspondences $\mathscr{A}$, the set of hyperedges $\mathscr{E}$ consists of subsets of $\mathscr{A}$ of cardinality 3, and $\phi(e)$ is the edge-weight function defined as $\phi(e) = \exp(-\beta d(e))$, where $\beta > 0$ is a precision parameter. Intuitively, the function $\phi(e)$ can be regarded as a *compatibility function* encoding the likelihood of the correspondences in e to be related by the same similarity transformation. According to our framework, an ESS-cluster C of H is a subset of correspondences in $\mathscr{A}$ exhibiting both internal coherence and external incoherence. Therefore, all correspondences in C are mutually highly compatible. This, by definition of ϕ, implies that all correspondences in C are related by the same similarity transformation between the model and the scene. Hence, C is a good candidate for being a potential match (i.e., a set of correspondences) providing a detection of the model in the scene, which is invariant to a similarity transformation. This motivates the use of our game-theoretic approach in order to address this matching problem from a clustering perspective. Note that this problem is particularly challenging, for only a small fraction of the correspondences in $\mathscr{A}$ will be part of a solution, the rest being outliers. Indeed, for example, if we consider a scene $\mathscr{S}$ containing any number of distinct instances of a model $\mathscr{M}$, then only a small share of *at most* $|\mathscr{M}|^{-1}$ correspondences appearing in $\mathscr{A}$ do belong to the solution.

We tested our approach on different artificial datasets. Each dataset is characterized by a reference model consisting of 30 random 3D points and a scene which contains up to 3 instances of the model. Each model instance in the scene is equivalent to the original one modulo a random similarity transformation. Moreover, a random subset of points of each copy of the model has been dropped (0-20% of points) in order to introduce structural noise. Consistently with the previous experiments, we considered two types of settings to assess the robustness of the algorithms to local and global noise. In the first set, we employed a Gaussian perturbation of the points in the scene, whereas in the second one 3D points (clutter points) were randomly added to the scene. For each different combination of number of model instances, noise type and noise level, we generated 20 random datasets. We refer to

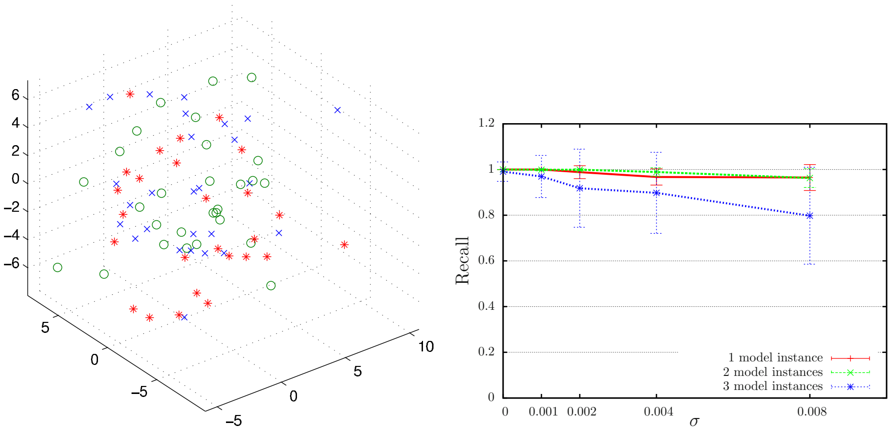
Figures 4(a) and 5(a) for examples of datasets with 3 model instances affected by local and global noise, respectively.

The hypergraphs we got in the various experimental settings were considerably large. Indeed, the number of vertices (i.e., potential correspondences) varied between 1000 and 5000 and the number of edges (i.e., triplets of correspondences) ranged approximately between 10^8 and 10^{10}. In order to reduce the size of the edge set, we adopted a sampling strategy aimed at efficiently excluding triplets that cannot belong to a good match. This allowed us to limit the number of edges to a maximum number of 25000 edges.

The evaluation protocol used to assess the quality of the results is given as follows. First, we clustered the hypergraphs thus obtaining a set of potential matches. Then, by means of the Horn method, we estimated a similarity transformation from the pointwise correspondences in each cluster. This yielded a set of m transformations $\{T_t\}_{t=1}^m$, which were used to determine the correspondences between the scene points and the model points according to the projection error. Specifically, let $d_{ijt} = \|T_t(\mathbf{m}_i) - \mathbf{s}_j\|$ be the distance between scene point $\mathbf{s}_j$ and model point $\mathbf{m}_i$ mapped according to transformation T_t and consider $(j^*, t^*) \in \arg\min_{(j,t)} d_{ijt}$. Then we decided to leave scene point $\mathbf{s}_i$ unassigned if $d_{ij^*t^*} > \tau$ (i.e., the point did not belong to the model) for some fixed threshold $\tau > 0$, while it was assigned to point $\mathbf{m}_{j^*}$, otherwise. Let $\mathscr{R} \subseteq \mathscr{A}$ be the set of assignments obtained according to this procedure and let $\mathscr{G} \subseteq \mathscr{A}$ be the set of ground-truth assignments of scene points to model points. We evaluated the quality of the obtained result in terms of the share of ground-truth assignments that have been correctly recovered (recall), i.e., $|\mathscr{G} \cap \mathscr{R}|/|\mathscr{G}|$.

In the following, we report only the results obtained by our approach, because the competitors CAVERAGE and SNTF were unable to provide a meaningful solution (recall below 10%). In fact, this is not surprising because the structure of the clustering problem arising from this application is characterized by an amount of wrong correspondences in $\mathscr{A}$ (outliers) that is considerably larger than the number of correct correspondences and, as demonstrated also by the previous series of experiments, partition-based approaches like CAVERAGE and SNTF are highly sensitive to outliers. As a consequence, the similarity transformations computed from the noisy clusters found by these approaches did not correspond to any mapping between the model and the scene. Our approach on the other hand was very robust to this kind of global noise and was thus able to perform considerably well also in challenging situations like the ones addressed in this experiment.

In Figure 4 we report the average recall and standard deviations obtained by our approach on the experiments with increasing level of local Gaussian noise ($\sigma = 0, 0.001, 0.002, 0.004, 0.008$). As experienced in previous sections, our approach achieves good scores, which slightly drop at increasing levels of noise. Indeed, larger perturbations of the points in the scene prevent the clustering approach from finding a correct similarity transformation and, therefore, some points in the scene are erroneously considered as clutter points. We also note that the drop in the performance is sharper in case of datasets with 3 model instances. This is due to the fact that 3 models in the scene lead to a higher density of points and, hence,



(a)

(b)

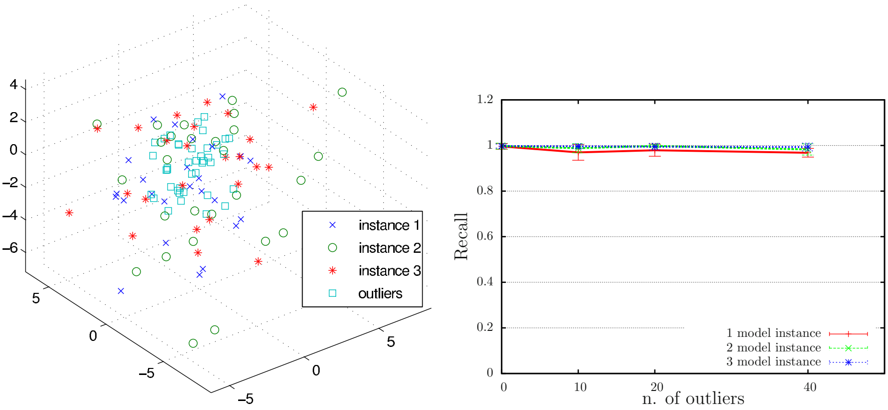
Fig. 4 Results of the experiments on model-based 3D point-pattern matching with $1,2$ and 3 model instances perturbed with increasing levels of Gaussian local noise ($\sigma = 0, 0.001, 0.002, 0.004, 0.008$). (a) Example of a model-based 3D point-pattern matching problem instance with 3 model instances perturbed with $\sigma = 0.004$. (b) Results obtained by our approach. Note that CAVERAGE and SNTF, which do not appear in the plots, obtained recall below 10%.

wrong assignments enforced by the local noise are more likely to happen. Additionally, the edge sampling procedure mentioned above lead to less accurate hypergraph representations in case of datasets with a large number of points.

In Figure 5 we report the results obtained by our approach with a fixed level of $\sigma = 0.001$ local Gaussian noise and with an increasing level of global noise, expressed in terms of $0, 10, 20, 40$ clutter points. The obtained results confirm the robustness of our approach to clutter points. Indeed, independently from the noise level and the number of model instances, we achieve an almost constant performance between 97%-100%.

5.3 Object Detection

In this section we show an application of our game-theoretic clustering approach to the problem of finding multiple instances of an object category within an image. Unlike the experiments that we have seen so far, we rely here on pairwise similarities. By exploiting Hough-voting based detection frameworks, like the Hough Forest [26] or the Implicit Shape Model [50], we cast the object detection problem into the problem of grouping hypothesis of an object's presence. These hypothesis consist in votes that have been generated by the Hough-based detection algorithms and have been collected within a generalized Hough space. As opposed to the standard approaches, which rely on *non-maxima suppression* (NMS) techniques to individuate an object from the set of noisy votes, we undertake a similarity-based approach.



(a)

(b)

Fig. 5 Results of the experiments on model-based 3D point-pattern matching with $1,2$ and 3 object instances with a level of Gaussian local noise of $\sigma = 0.001$ and increasing number of clutter points $(0,10,20,40)$. (a) Example of a model-based 3D point-pattern matching problem instance with 40 clutter points. (b) Results obtained by our approach. Note that CAVERAGE and SNTF, which do not appear in the plots, obtained recall below 10%.

We first organize the set of voting element derived from Hough-voting based detection frameworks into a pairwise compatibility matrix representation, where we take into account geometric information like orientation and object center agreement between voting elements. In a second step, we analyse this compatibility matrix with our game-theoretic clustering framework in order to detect groups of votes that are mutually highly compatible and, by construction, geometrically coherent. These groups represent our final object hypothesis in contrast to bounding-box delimited object detections, typically obtained as a result of NMS methods.

We denote with S the set of N observations (voting elements) from an image. Each observation $i \in S$ has a spatial origin $\mathbf{y}_i$ in the image space, stemming from voting elements and their respective descriptors I_i. We furthermore assume that we are given a classification function $h(i)$ for the class label assignment and a probability score $p(h(i)|I_i)$ for each voting element $i \in S$. In addition, each voting element $i \in S$ obtains a voting vector $\mathbf{d}_i$ after being classified, pointing towards its associated object center. All of the above parameters can be obtained from previous works like the ISM [50] or the Hough Forest [26].

We organize the collected voting elements into a pairwise representation by means of a compatibility function, which is composed by different terms. One term consists in the object center certainty which is modelled as a function weighting the distance between the hypothesized centers of voting elements $i, j \in S$ according to

$$p_{c_{ij}} = \exp\left(-\frac{||(\mathbf{y}_i + \mathbf{d}_i) - (\mathbf{y}_j + \mathbf{d}_j)||^2}{\sigma_h^2}\right), \tag{9}$$

where σ_h is a parameter to control the allowed deviation. This term may also be considered as pairwise breakdown of the original Hough center projection. A second component of our pairwise representation models the orientation similarities between the considered pair of votes and the actual relative orientation between the spatial origins in the image domain. Hence, we define

$$p_{\varphi_{ij}} = \exp\left(-\frac{\measuredangle(\hat{\mathbf{y}}_{ij}, \hat{\mathbf{d}}_{ij})^2}{\sigma_\varphi^2}\right), \tag{10}$$

where $\measuredangle(\cdot,\cdot)$ returns the enclosed angle between the normalized vectors $\hat{\mathbf{y}}_{ij} = \frac{\mathbf{y}_i - \mathbf{y}_j}{||\mathbf{y}_i - \mathbf{y}_j||}$ and $\hat{\mathbf{d}}_{ij} = \frac{\mathbf{d}_i + \mathbf{d}_j}{||\mathbf{d}_i + \mathbf{d}_j||}$, mapped on the interval $[0, \pi]$. This orientation feature penalizes differences between the observed geometric configuration in the image and the provided voting information. σ_φ allows to control the influence of the orientation feature. By combining the terms in Equ. (9) and (10), we construct a *compatibility function* $C : S \times S \rightarrow [0, 1]$ defined as follows:

$$C(i, j) = p(h(i)|I_i)p(h(j)|I_j)p_{c_{ij}}p_{\varphi_{ij}}\,. \tag{11}$$

Please note that a voting pair (i, j) has to satisfy not only the geometrical constraints formulated in Equ. (9) and (10) but also needs to be classified as part of the object in order to receive a non-zero compatibility value.

The compatibility function C can be regarded as the payoff π of a 2-graph clustering game, where the voting elements in S represent the data objects to be clustered. By construction, we have that an ESS-cluster of this game, represents a set of voting elements that are geometrically compatible in the sense that they provide strong votes on the same object hypothesis and can thus be regarded as an object detection.

We use the Hough forest [26] in order to provide the required data to construct compatibility matrix and set $\sigma_h = \sigma_\varphi = 9$. In every Hough tree $t \in \mathscr{T}$, we reduce the set of voting vectors in every leaf node to the median vote vector $\mathbf{d}$. In order to keep the resulting payoff matrices at reasonable size, we constrain the number of considered voting elements to patches with foreground probability ≥ 0.5 and to locations with a gradient magnitude ≥ 25 for pedestrians and ≥ 10 for cars. Additionally, we consider only pixels lying on a regular lattice with a stride of 2 which massively reduces the amount of data to be processed. Unless otherwise stated, we always grow 15 trees with a maximum depth of 12 on 25 000 positive and negative training samples from the referenced data sets. The considered patch size is 16×16 and all training samples are resized to a similar scale.

We apply our method for localization of cars on the UIUC cars dataset [2] and pedestrians on the extended TUD crossing dataset [10]. In order to demonstrate the broad applicability, we also show qualitative mouth detection results on images of the BioID Face Database.[6]

UIUC Car Dataset. In our first experiment we evaluate the proposed method on the single scale UIUC car dataset [2]. The training dataset contains 550 positive

[6] BioID Technology Research. `http://www.bioid.de/`

and 450 negative images while the test dataset consists of 170 test images showing 210 cars. Although the silhouettes of the cars are mostly rigid, some cars are partially occluded or have low contrast while being located in cluttered background. We achieve a score of 98.5% in terms of equal error rate (EER), hence we are on par with state-of-the-art methods [47, 26] while identifying the set of votes that are corresponding to the individual objects. In Figure 6 we show some sample detections and the groups of votes producing the respective detections. Please note how our method is able to deal with partial occlusions and successfully groups coherent object votes.

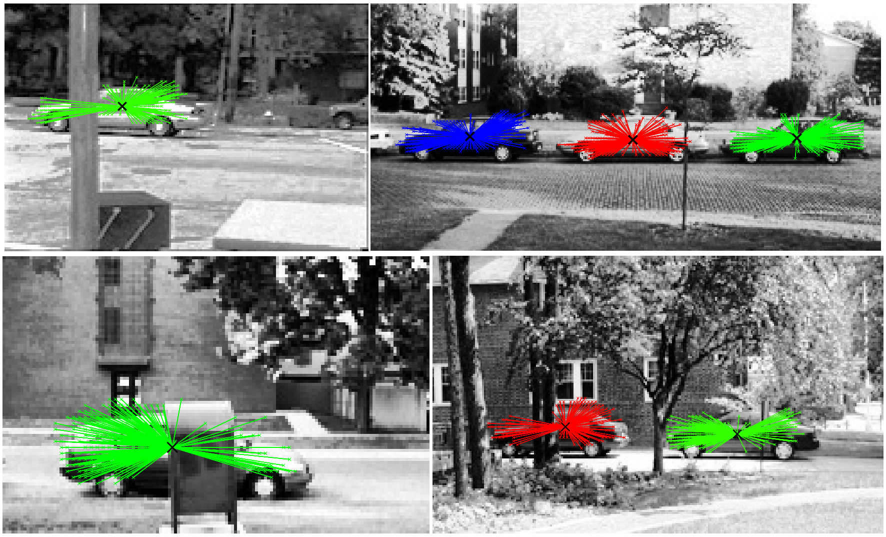


Fig. 6 Car detections and their contributing votes on selected images of UIUC car databset

Extended TUD Crossing Scene. Next, we evaluated on the extended version of the TUD crossing database [10], showing several strongly occluded pedestrians walking on a cross-walk. The extended version includes also overlapping pedestrians where head and at least one leg are visible. This results in a very challenging data set consisting of 201 images with 1018 bounding boxes. We used the same training protocol as described in [8]. Since we are not obtaining bounding boxes but rather the sets of contributing voting elements for each person, we decided to evaluate the detection results with the strict criterion introduced in [72]. This criterion accepts detections as correct when the hypothesized center is within 25 pixels of the bounding box centroid on the original scale. In our case, we determined the centroid by taking the median of the reprojected center votes for all detected voting elements. For evaluation, we rescaled the images and the acceptance criterion by a factor of 0.55, such that true positives were counted only within a radius of 13.75 pixels. After constructing the payoff matrices, we found the ESS-clusters. To provide a comparison, we handed the same matrices to the widespread normalized cut (nCut) algorithm [71] and illustrate the results (F-measure per test image) in the top row in Figure 7. Since nCut requires the number of clusters to be given, we evaluated it by providing our number of detections as well as the ground truth number of persons. As can be seen, our method outperforms the nCut algorithm, even when the true number of objects is provided. We obtain a mean F-measure score of 79.88% compared to 66.56% and 65.23% for nCut provided with ground truth or our detected number of persons, respectively. Since nCut aims at partitioning the whole input into clusters, we tried another setup where we give an additional cluster to the ground truth number and the detected number of persons from our method, respectively. This should allow nCut for partitioning the non-person objects. Before computing the F-measure, we removed the detection associated to the lowest eigenvalue. This resulted in F-measure scores of 61.94% and 58.79%, hence considerably lower than before and suggesting that nCut does not group noise in an individual cluster but rather incorporates it in the individual detections. The bottom row in Figure 7 shows color-coded, qualitative results of individual detections of our method. Please note the plausible assemblies of votes from strongly overlapping persons to individual pedestrians, even in the rightmost image, where a person is missed due to assignment of votes to another person in the back. Moreover, it is possible to hypothesize for the person's center by detecting coherent votes of the feet alone (green detection in first image, yellow detection in forth image).

Mouth Localization. With this experiment we demonstrate yet another application of our method. The Hough forest software package provides readily trained trees for mouth detection, presumably those used in [22]. However, we used these trees for evaluating on some selected images of the BioID Face Database. The obtained solutions (*i. e.* the support forming the coherent vote sets) live in the standard simplex and therefore each voting element is associated with a probability, describing its individual importance for the set. In Figure 8 we illustrate the importance of the individual votes on some sample images. It can be seen that elements truly belonging to the mouth regions are associated with higher probabilities.

5.4 Illuminant-Invariant Face Clustering

In [14] it has been shown that images of a Lambertian object illuminated by a point light source lie in a three-dimensional subspace. According to this result, if we assume that four images of a face form the columns of a matrix, then $d = s_4^2/(s_1^2 + \cdots + s_4^2)$ provides us with a measure of dissimilarity, s_i being the ith singular value of this matrix. Following [4], we used this dissimilarity measure for clustering faces in high-dimensional space. We tested our algorithm and its competitors over the Yale Face Database B and its extended version [27, 49], which contained faces of 38 individuals under 64 different illumination conditions. Specifically, we considered subsets of faces from 4 and 5 randomly drawn individuals (10 faces per individual), with and without outlier faces. The case with outliers consisted in 10 additional faces taken from as many random individuals. For each such

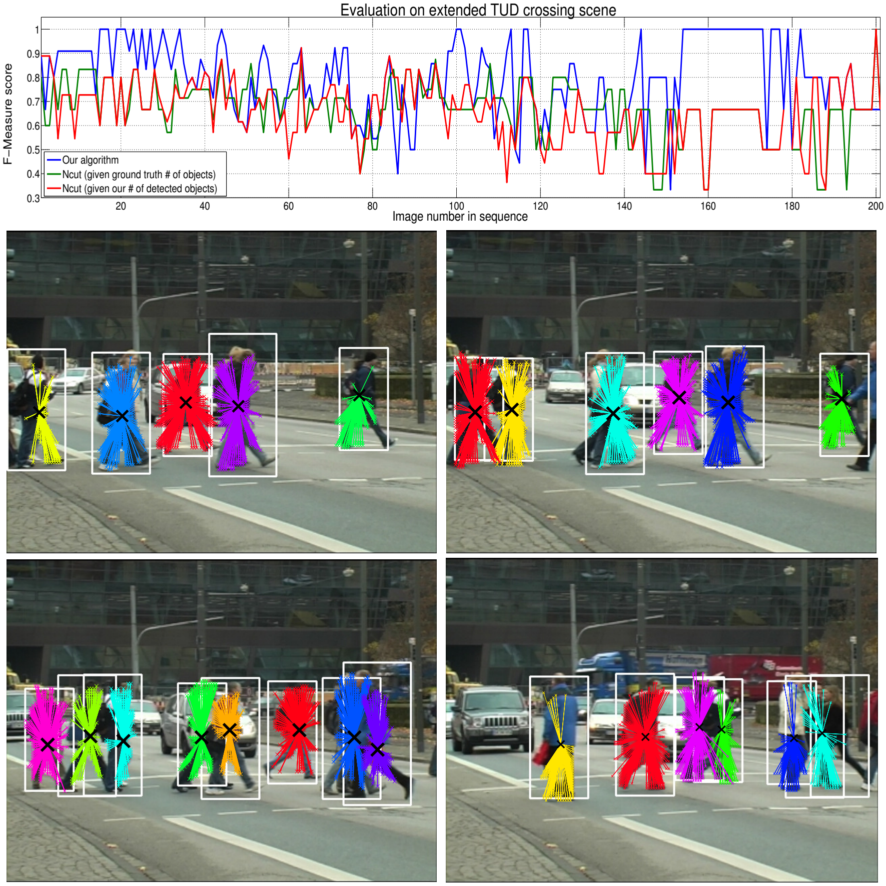


Fig. 7 Top row: Classification results on extended TUD crossing sequence per image using single scale evaluation. We obtain a mean F-Measure score of 79.88% in comparison to 66.56% and 65.23% for nCut [71] (provided with ground truth # or our detected # of objects, respectively). Second and third rows: Successive and missing (last image) detections of proposed method. White bounding boxes correspond to ground truth annotations. Best viewed in color.

combination, we created 10 different subsets (see Figure 9 for an example with 4 individuals and outlier faces). Similarly to the case of line clustering, we run both CAVERAGE and SNTF with values of $K \in \{K^* - 1, K^*, K^* + 1\}$, where K^* is the correct number of individuals.

Table 1 reports the results obtained by the three approaches in terms of classification error (mean and standard deviation). The results are consistent with those obtained in the case of line clustering with the exception of SNTF, which performed worse than the other approaches. On the other hand, our algorithm and (the optimal-tuned) CAVERAGE performed comparably well within the no-outlier setting, while

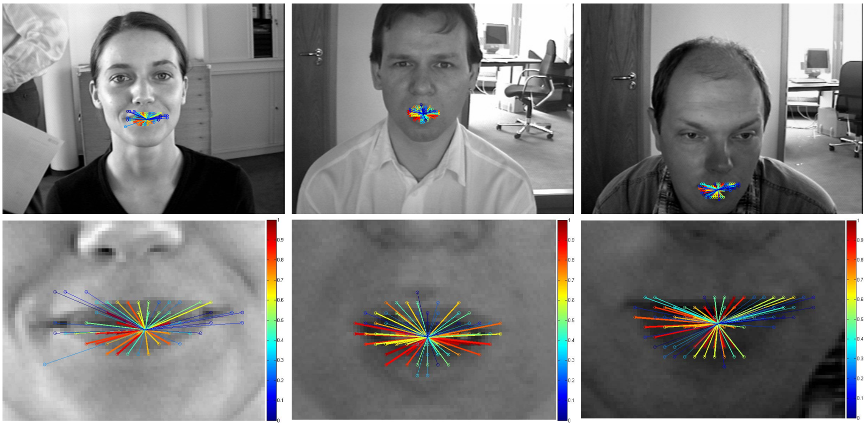


Fig. 8 Selected mouth detections on BioID Face Database. Color-coded voting elements are associated to their individual importance for the extracted coherent sets.

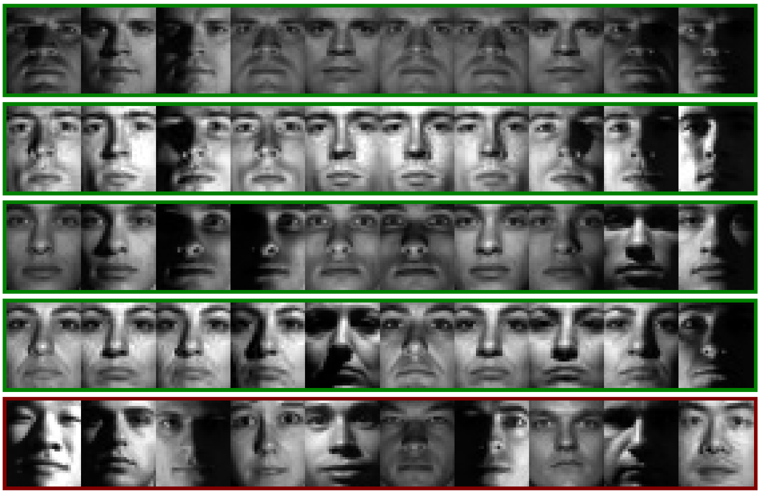


Fig. 9 Example of dataset for illuminant-invariant face clustering with 4 individuals (first four rows) and 10 outlier faces (last row).

our approach dramatically outperformed the other algorithms in the cases comprising outliers.

6 Conclusions

In this chapter, we have reviewed our recent game-theoretic formulation of the clustering problem (more details can be found in [65, 66, 45]). Within our framework

Table 1 Experiments on illuminant-invariant face clustering. We report the average classification error and the corresponding standard deviation.

n. of classes:	4		5	
n. of outliers:	0	10	0	10
CAVERAGE K=3	0.26±0.09	0.40±0.10	-	-
CAVERAGE K=4	**0.03±0.04**	0.24±0.07	0.21±0.11	0.65±0.12
CAVERAGE K=5	0.13±0.05	0.12±0.05	0.07±0.07	0.41±0.09
CAVERAGE K=6	-	-	0.13±0.08	0.37±0.11
SNTF K=3	0.29±0.10	0.39±0.09	-	-
SNTF K=4	0.14±0.06	0.26±0.09	0.28±0.11	0.51±0.12
SNTF K=5	0.19±0.09	0.25±0.13	0.11±0.09	0.43±0.11
SNTF K=6	-	-	0.14±0.09	0.39±0.13
HoCluGame	0.06±0.03	**0.07±0.02**	**0.06±0.02**	**0.07±0.03**

the problem is viewed as a non-cooperative game, and classical equilibrium notions from evolutionary game theory turn out to provide a natural formalization of the notion of a cluster. We showed that the problem of finding these equilibria (clusters) is equivalent to solving a polynomial optimization problem with linear constraints, which we solve using (high-order) replicator dynamics based on the Baum-Eagon inequality.

In a nutshell, our game-theoretic perspective has the following attractive features:

1. it makes no assumption on the underlying (individual) data representation: like, e.g., spectral clustering, it does not require that the elements to be clustered be represented as points in vector space;
2. it does not require *a priori* knowledge on the number of clusters (since it extracts them sequentially);
3. it leaves clutter elements unassigned (useful, e.g., in figure/ground separation or one-class clustering problems)
4. it allows extracting overlapping clusters (see, e.g., [74])
5. it can naturally handle high-order similarities.

Notice that, in the pairwise case, the approach allows also using asymmetric affinity matrices, which might be useful in several circumstances [73]. The experimental results presented here on various problems show the superiority of our approach over the state of the art in terms of quality of solution. Further computer vision applications of our framework can be found, e.g., in [7, 5, 64, 6, 37].

We are currently studying alternatives to the Baum-Eagon dynamics in order to improve efficiency (e.g., [67]). We finally note that, inspired by our work in [65], in a recent paper a parametrized version of our framework has been introduced, which allows one to control the minimum cluster size [51].

The approach outlined above is but one example of using purely game-theoretic concepts to model *generic* machine learning problems (see [18] for another such example in a totally different context), and the potential of game theory to machine learning is yet to be fully explored. Other areas where game theory could potentially offer a fresh and powerful perspective include, e.g., semi-supervised learning, multi-similarity learning, multi-task learning, learning with incomplete information, learning with context-dependent similarities. The concomitant increasing interest around the algorithmic aspects of game theory [58] is certainly beneficial in this respect, as it will allow useful cross-fertilization of ideas.

Acknowledgements. We would like to thank Peter Kontschieder, Michael Donoser and Horst Bischof for allowing us to use material from our joint paper [45].

References

1. Ackerman, M., Ben-David, S.: Measures of clustering quality: A working set of axioms for clustering. In: Advances in Neural Inform. Process. Syst. (2008)
2. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. IEEE Trans. Pattern. Anal. Machine Intell. 26, 1475–1490 (2004)
3. Agarwal, S., Branson, K., Belongie, S.: Higher order learning with graphs. Int. Conf. on Mach. Learning 148, 17–24 (2006)
4. Agarwal, S., Lim, J., Zelnik-Manor, L., Perona, P., Kriegman, D., Belongie, S.: Beyond pairwise clustering. In: IEEE Conf. Computer Vision and Patt. Recogn., vol. 2, pp. 838–845 (2005)
5. Albarelli, A., Rodolà, E., Torsello, A.: A game-theoretic approach to fine surface registration without initial motion estimation. In: IEEE Conf. Computer Vision and Patt. Recogn. (2010)
6. Albarelli, A., Rodolà, E., Torsello, A.: Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: A game-theoretic perspective. Int. J. Comput. Vision 97(1), 36–53 (2012)
7. Albarelli, A., Torsello, A., Rota Bulò, S., Pelillo, M.: Matching as a non-cooperative game. In: Int. Conf. Comp. Vision (2009)
8. Andriluka, M., Roth, S., Schiele, B.: People-tracking-by-detection and people-detection-by-tracking. In: IEEE Conf. Computer Vision and Patt. Recogn. (2008)
9. Banerjee, A., Krumpelman, C., Basu, S., Mooney, R.J., Ghosh, J.: Model-based overlapping clustering. In: Int. Conf. on Knowledge Discovery and Data Mining, pp. 532 – 537 (2005)
10. Barinova, O., Lempitsky, V., Kohli, P.: On detection of multiple object instances using hough transforms. In: IEEE Conf. Computer Vision and Patt. Recogn., pp. 2233–2240 (2010)
11. Baum, L.E., Eagon, J.A.: An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. Bull. Amer. Math. Soc. 73, 360–363 (1967)
12. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann. Math. Statistics 41, 164–171 (1970)
13. Baum, L.E., Sell, G.R.: Growth transformations for functions on manifolds. Pacific J. Math. 27, 221–227 (1968)
14. Belhumeur, P., Kriegman, D.: What is the set of images of an object under all possible lighting conditions. Int. J. Comput. Vision 28(3), 245–260 (1998)
15. Berge, C.: Hypergraphs: Combinatorics of Finite Sets. North-Holland, Amsterdam (1989)

16. Blakley, G.R.: Homogeneous nonnegative symmetric quadratic transformations. Bull. Amer. Math. Soc. 70, 712–715 (1964)
17. Broom, M., Cannings, C., Vickers, G.T.: Multi-player matrix games. Bull. Math. Biology 59(5), 931–952 (1997)
18. Cesa Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press (2006)
19. Crammer, K., Talukdar, P.P., Pereira, F.: A rate-distortion one-class model and its applications to clustering. In: Int. Conf. on Mach. Learning (2008)
20. Eco, U.: Kant and the Platypus: Essays on Language and Cognition. Harvest Books (2000)
21. Etessami, K., Lochbihler, A.: The computational complexity of evolutionarily stable strategies. Int. J. Game Theory 37(1), 93–113 (2007)
22. Fanelli, G., Gall, J., van Gool, L.: Hough transform-based mouth localization for audio-visual speech recognition. In: British Machine Vision Conf. (2009)
23. Faugeras, O.D., Berthod, M.: Improving consistency and reducing ambiguity in stochastic labeling: an optimization approach. IEEE Trans. Pattern. Anal. Machine Intell. 3, 412–424 (1981)
24. Frankl, P., Rödl, V.: Hypergraphs do not jump. Combinatorica 4, 149–159 (1984)
25. Fudenberg, D., Tirole, J.: Game Theory. MIT Press, Cambridge (1991)
26. Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: IEEE Conf. Computer Vision and Patt. Recogn., pp. 1022–1029 (2009)
27. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.J.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern. Anal. Machine Intell. 23(6), 643–660 (2001)
28. Gibson, D., Kleinberg, J.M., Raghavan, P.: Clustering categorical data: An approach based on dynamical systems. In: Proc. 24th Int. Conf. Very Large Data bases (VLDB), pp. 311–322 (1998)
29. Govindu, V.M.: A tensor decomposition for geometric grouping and segmentation. In: IEEE Conf. Computer Vision and Patt. Recogn., pp. 1150–1157 (2005)
30. Gupta, G., Ghosh, J.: Robust one-class clustering using hybrid global and local search. In: Int. Conf. on Mach. Learning (2005)
31. Guyon, I., von Luxburg, U., Williamson, R.C.: Clustering: Science or art? In: JMLR: Workshop and Conference Proceedings, vol. 27, pp. 65–79 (2012)
32. Hartigan, J.: Introduction. In: Arabie, P., Hubert, L.J., de Soete, G. (eds.) Clustering and Classification. World Scientific, River Edge (1996)
33. Heller, K., Ghahramani, Z.: A nonparametric bayesian approach to modeling overlapping clusters. In: Int. Conf. AI and Statistics (2007)
34. Herault, L., Horaud, R.: Figure-ground discrimination: a combinatorial optimization approach. IEEE Trans. Pattern. Anal. Machine Intell. 15(9), 899–914 (1993)
35. Hofbauer, J., Sigmund, K.: Evolutionary games and population dynamics. Cambridge University Press, Cambridge (1998)
36. Horn, K.P.: Closed-form solution of absolute orientation using unit quaternions. J. Optical Soc. of America A 4, 629 (1987)
37. Hsiao, P.C., Chang, L.W.: Image denoising with dominant sets by a coalitional game approach. IEEE Trans. Image Process. 22(2), 724–738 (2013)
38. Hu, T., Moerder, K.: Multiterminal flows in hypergraphs. In: Hu, T., Kuh, E.S. (eds.) VLSI Circuit Layout: Theory and Design, pp. 87–93 (1985)
39. Huang, Y., Liu, Q., Lv, F., Gong, Y., Metaxas, D.N.: Unsupervised image categorization by hypergraph partition. IEEE Trans. Pattern. Anal. Machine Intell. 33, 1266–1273 (2011)

40. Jain, A.K.: Data clustering: 50 years beyond K-means. Pattern Recogn. Letters 31(8), 651–666 (2010)
41. Jain, A.K., Dubes, R.C.: Algorithms for data clustering. Prentice-Hall (1988)
42. Jardine, N., Sibson, R.: The construction of hierarchic and non-hierarchic classifications. Computer J. 11, 177–184 (1968)
43. Karypis, G., Kumar, V.: Multilevel k-way hypergraph partitioning. VLSI Design 11(3), 285–300 (2000)
44. Kleinberg, J.M.: An impossibility theorem for clustering. In: Advances in Neural Inform. Process. Syst. (2002)
45. Kontschieder, P., Rota Bulò, S., Donoser, M., Pelillo, M., Bischof, H.: Evolutionary hough games for coherent object detection. Comp. Vis. and Image Understanding 116, 1149–1158 (2012)
46. Lakoff, G.: Women, Fire, and Dangerous Things: What Categories Reveal about the Mind. The University of Chicago Press (1987)
47. Lampert, C., Blaschko, M., Hofmann, T.: Beyond sliding windows: Object localization by efficient subwindow search. In: IEEE Conf. Computer Vision and Patt. Recogn. (2008)
48. LaSalle, J.P.: The Stability and Control of Discrete Processes. Springer, New York (1986)
49. Lee, K.C., Ho, J., Kriegman, D.: Acquiring linear subspaces for face recognition under variable lighting. IEEE Trans. Pattern. Anal. Machine Intell. 27(5), 684–698 (2005)
50. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. Int. J. Comput. Vision 77(1-3), 259–289 (2008)
51. Liu, H., Latecki, L.J., Yan, S.: Robust clustering as ensembles of affinity relations. In: Advances in Neural Inform. Process. Syst., vol. 23, pp. 1414–1422 (2010)
52. Luenberger, D.G.: Introduction to Dynamic Systems. Wiley, New York (1979)
53. Luenberger, D.G.: Linear and nonlinear programming. Addison Wesley, Reading (1984)
54. von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)
55. Maynard Smith, J.: Evolution and the Theory of Games. Cambridge University Press, Cambridge (1982)
56. Mohammed, J.L., Hummel, R.A., Zucker, S.W.: A gradient projection algorithm for relaxation labeling methods. IEEE Trans. Pattern. Anal. Machine Intell. 5, 330–332 (1983)
57. Nisan, N.: A note on the computational hardness of evolutionary stable strategies. In: Electr. Colloquium on Comp. Complexity (2006)
58. Nisan, N., Roughgarden, T., Tardos, É., Vazirani, V. (eds.): Algorithmic Game Theory. Cambridge University Press (2007)
59. Pavan, M., Pelillo, M.: A new graph-theoretic approach to clustering and segmentation. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 145–152 (2003)
60. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern. Anal. Machine Intell. 29(1), 167–172 (2007)
61. Pelillo, M.: The dynamics of nonlinear relaxation labeling processes. J. Math. Imag. and Vision 7(4), 309–323 (1997)
62. Pelillo, M., Torsello, A.: Payoff-monotonic game dynamics and the maximum clique problem. Neural Computation 18(5), 1215–1258 (2006)
63. Perona, P., Freeman, W.T.: A factorization approach to grouping. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 655–670. Springer, Heidelberg (1998)
64. Rodolà, E., Bronstein, A.M., Albarelli, A., Bergamasco, F., Torsello, A.: A game-theoretic approach to deformable shape matching. In: IEEE Conf. Computer Vision and Patt. Recogn. (2012)

65. Rota Bulò, S., Pelillo, M.: A game-theoretic approach to hypergraph clustering. In: Advances in Neural Inform. Process. Syst., vol. 22, pp. 1571–1579 (2009)
66. Rota Bulò, S., Pelillo, M.: A game-theoretic approach to hypergraph clustering. IEEE Trans. Pattern. Anal. Machine Intell. 35(6), 1312–1327 (2013)
67. Rota Bulò, S., Pelillo, M., Bomze, I.M.: Graph-based quadratic optimization: A fast evolutionary approach. Comp. Vis. and Image Understanding 115, 984–995 (2011)
68. Sarkar, S., Boyer, K.L.: Quantitative measures of change based on feature organization: eigenvalues and eigenvectors. Comp. Vis. and Image Understanding 71(1), 110–136 (1998)
69. Shashua, A., Ullman, S.: Structural saliency: The detection of globally salient features using a locally connected network. In: Int. Conf. Comp. Vision (1988)
70. Shashua, A., Zass, R., Hazan, T.: Multi-way clustering using super-symmetric nonnegative tensor factorization. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 595–608. Springer, Heidelberg (2006)
71. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern. Anal. Machine Intell. 22, 888–905 (2000)
72. Shotton, J., Blake, A., Cipolla, R.: Contour-based learning for object detection. In: Int. Conf. Comp. Vision (2005)
73. Torsello, A., Rota Bulò, S., Pelillo, M.: Grouping with asymmetric affinities: A game-theoretic perspective. In: IEEE Conf. Computer Vision and Patt. Recogn., pp. 292–299 (2006)
74. Torsello, A., Rota Bulò, S., Pelillo, M.: Beyond partitions: Allowing overlapping groups in pairwise clustering. In: Int. Conf. Patt. Recogn. (2008)
75. Weibull, J.W.: Evolutionary game theory. Cambridge University Press, Cambridge (1995)
76. Zadeh, R.B., Ben-David, S.: A uniqueness theorem for clustering. In: Uncertainty in Artif. Intell. (2009)
77. Zhou, D., Huang, J., Schölkopf, B.: Learning with hypergraphs: Clustering, classification, embedding. In: Advances in Neural Inform. Process. Syst., vol. 19, pp. 1601–1608 (2006)

About 3D Faces

Stefano Berretti, Alberto Del Bimbo and Pietro Pala

Abstract. Identity recognition using 3D scans of the face has been recently proposed as an alternative or complementary solution to conventional 2D face recognition approaches based on still images or videos. In fact, face representations based on 3D data are expected to be much more robust to pose changes and illumination variations than 2D images, thus allowing accurate face recognition also in real-world applications with unconstrained acquisition. Based on these premises, in this Chapter we will first introduce the general and main methodologies for 3D face data acquisition and preprocessing, also presenting some 3D benchmark databases and performance indicators used for evaluation and comparison. Then, we will discuss some of the results recently achieved on this subject, also presenting current trends and challenges of the research.

1 Introduction

Human target recognition has been an active research area in recent years, with several biometric techniques developed for measuring unique physical and behavioral characteristics of human subjects for the purpose of recognizing their identity. In particular, two different modalities are considered to recognize the identity of a person: *verification* (authentication) and *identification* (recognition). Verification ("Am I who I claim I am?") involves confirming or denying a person's claimed identity. Instead, identification ("Who am I?") requires the system to recognize a person from a list of users in the template database. Due to this, identification is a more challenging problem because it involves *one-to-many* matching compared to the *one-to-one* matching required for verification. The idea of automatically recognizing or authenticating users' identity is based on the possibility to extract unique physical features

Stefano Berretti · Alberto Del Bimbo · Pietro Pala
Dipartimento di Ingegneria dell'Informazione, University of Firenze, via S.Marta 3, 50139 Firenze, Italy
e-mail: {stefano.berretti,alberto.delbimbo,pietro.pala}@unifi.it

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_9, © Springer-Verlag Berlin Heidelberg 2014

from the anatomical traits that univocally characterize each individual. The features that are most used for this goal can be summarized as follows:

- *Fingerprints and hand geometry* – The most common biometric authentication. Provides high accuracy, it is easy to implement (though contact with the sensor is required), showing a low cost. Can be also performed via the Internet (BioWeb);
- *Voice recognition* – Relies on the voice pattern to authenticate individuals. Very user friendly. However, changing the voice due to sinus congestion, cold or anxiety can produce false negatives results;
- *Eye scans* – Retinal and iris scans are used for authentication. They provide accuracy where physical contact to the scanner is required. The user must focus in particular point in the scanner and hold this position. Low-intensity light might affect the results;
- *Facial recognition* – Looks for the different parts of the face such as the location, and shape of the eyes and the nose, cheekbones and the side of the mouth;
- *Signature dynamics and typing patterns* – Looks for patterns in writing pressures at different points in the signature, and the writing speed;
- *Heartbeat biometric authentication* – Identifies the individually unique information of the subject heartbeats;
- *Infrared hand vein pattern biometric* – Uses the shape of the finger vein and infrared is used to make the skin tissue transparent, and highly visible to recognize the veins in the finger.

Table 1 Comparison between biometric technologies

Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H[a]	L	M	H	L	H	L
Fingerprint	M	H	H	M	H	M	H
Hand geometry	M	M	M	H	M	M	M
Keystroke dynamics	L	L	L	M	L	M	M
Hand vein	M	M	M	M	M	M	H
Iris	H	H	H	M	H	L	H
Retina	H	H	M	L	H	L	H
Signature	L	L	L	H	L	H	L
Voice	M	L	L	M	L	H	L
Facial Thermogram	H	H	L	H	M	H	H
DNA	H	H	H	L	H	L	L

H = High, M = Medium, L = Low

[a]

Table 1 summarizes and rates the main characteristics of the biometric features used for human identity recognition with respect to: *universality* (how common is found in each person); *uniqueness* (how well separates persons); *permanence* (how well resists aging); *collectability* (how easy it is to acquire); *performance* (the achievable accuracy); *acceptability* (the degree of acceptance by the public); *circumvention* (the level of difficulty to circumvent).

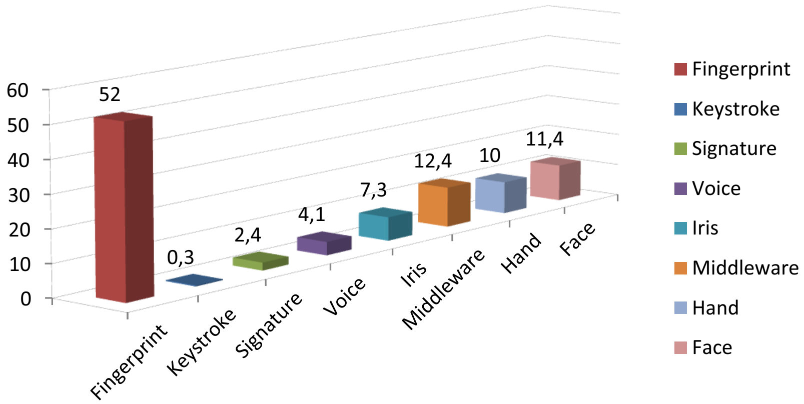


Fig. 1 The impact of different biometric techniques (in percentage)

Depending on the particular application, one or a combination of the diverse biometric modalities listed in Tab. 1 can be more appropriate. This is evidenced by the different diffusion and impact that different biometric technologies have on the global market, as reported in Fig. 1 (data are referred to the year 2010). It can be observed, that fingerprints is the most largely used biometric technique, mainly because of its very high accuracy and simplicity, with face recognition following in third position after middleware solutions.

While biometric technologies are being widely used in *forensics* for criminal identification, recent advancements in biometric sensors and matching algorithms have led to the deployment of biometric authentication in a large number of civilian and government applications, such as *physical access control*, *computer log-in*, *welfare disbursement*, *international border crossing and national ID cards*, etc. Before implementing such technologies within any businesses as a method of authentication it is imperative to identify risks and cost justifications:

- Biological uniqueness attributes can change over time: injures may change fingerprint features, making it hard to match, the same with retinal changes;
- High frequency of false positives and false negatives, incorrect calibration, and inaccurate initial reading can cause identity problems;
- Spoofing attack by artificial features is a security problem, especially on a single biometric feature. A combined solution of authentication might be effective;
- Stolen biometrics presents a problem. While stolen smart cards or passwords can be reissued or changed, biometric data is there to stay forever or to be excluded from the authentication system, and users no longer can be authenticated by such technique. Such incidents can raise security risks and cost;
- The deployment of biometric technology involves collecting biometric data which can be a big task to take for any organization. The cost of IT resources to deploy and maintain biometric readers is a huge challenge within any organization.

Among the biometric techniques listed above, identity recognition based on facial traits is widely used for its social acceptance, applicability in a range of different contexts and the good balance between risks and benefits associated to its implementation. In fact, face recognition has its main prerogative in not requiring contact or closeness between the acquisition sensor and the captured subject, thus permitting its deployment in a variety of different situations which span from indoor application with constrained pose and illumination conditions (for example in capturing face images used for personal identification documents) to the surveillance of vast outdoor areas with unconstrained conditions (as can be the case of a sporting event) using pan-tilt-zoom (PTZ) active cameras. In fact, automatic human target identification by detecting and matching human faces in 2D still images and videos has been an active research area in pattern recognition since 90s [67]. Performance of 2D face matching systems depends on their capability of being insensitive to critical factors such as facial expressions, makeup, and aging, but mainly hinges upon extrinsic factors such as illumination differences, camera viewpoint and scene geometry. The Face Recognition Vendor Test (FRVT, `http://face.nist.gov/`) is an independent evaluation contest of face recognition algorithms carried out every two/three years by the National Institute of Standards and Technologies (NIST). The FRVT 2002 [50] showed that performance in the presence of illumination variations decreases up to 46 percent and similar and higher decreases occur for rotations of the face with respect to the frontal case. Great progress was documented in the 2006 FRVT [51]. The best performer showed a False Rejection Rate (FRR) interquartile range between 0.6 and 1.5 percent at 0.001 False Acceptance Rate (FAR) under controlled illumination, and between 10.3 and 13 percent at 0.001 FAR across illumination changes (see Sect. 3.2 for a definition of FRR and FAR). A performance decrease of about one order of magnitude was observed at lower resolution.

The inherent limitations of 2D face matching have supported the belief that effective recognition of identity should be obtained through multi-biometric technologies. In particular, the exploitation of the geometry of the anatomical structure of the face rather than its appearance with definition of algorithms and systems for 3D face matching has been a growing field of research in very recent years. Based on these premises, there are several topics related to 3D faces which are attracting an increasing interest:

- 3D face *acquisition* and *preprocessing*;
- 3D face *datasets* with challenging scans for recognition and facial expression recognition;
- In *cooperative* contexts
  - 3D to 3D face recognition on (*large*) *databases*;
  - 3D to 3D face recognition with *aging*;
  - 3D to 3D *face ethnicity* / *gender* recognition;
  - 3D to 3D *static facial expressions* recognition;
- In *semi-cooperative* or *non-cooperative* contexts
  - 3D to 3D face recognition in presence of *pose variations* and *occlusions*;

- 4D (3D + time) *dynamic facial expression* recognition;
- 3D *low-resolution* to 3D *high-resolution* face recognition;
- 2D to 3D face recognition *hybrid / multimodal*.

In particular, 3D face acquisition and preprocessing are the first operations that precede any 3D face analysis task performed in real contexts. The outcomes of these operations are of paramount importance in that, depending on their quality, the accuracy achievable by any subsequent 3D face analysis can largely vary. Then, a substantial different exists between approaches that operate in *cooperative* or *non-cooperative* contexts. In the former case, subjects are aware and collaborate to the acquisition process. For example, it can be the case of an access control performed via 3D face acquisition and recognition: in this case the user is asked to assume a predefined position in front of the scanner device and the acquisition environment is also set up (for example in terms of background and lighting conditions) in a way that can maximize the quality of the acquisition. Differently, in the latter case, authors cooperate to the acquisition only partially or not at all, in that they can change their pose or even move. In the most challenging cases, the environment can be constituted by outdoor areas with changing illumination conditions, varying background and crowding, thus making extremely challenging the acquisition of face scans of sufficient quality for recognition purposes. In summary, the conversion of 3D scans to efficient and meaningful descriptors of the face structure is therefore crucial to performing fast processing and particularly to permitting indexing over large data sets for identification. On the other hand, the effectiveness of 3D face recognition is principally concerned with the capability of achieving invariance to face expressions, missing parts and occlusions. In fact, while 3D face models are almost insensitive to lighting conditions, they are affected by pose changes and occlusions, and are even more sensitive than 2D images to face expressions.

The above considerations evidence the richness and potential impact of face recognition applications based on 3D scans. In the remaining of this Chapter, we will focus on some of the above trends of investigation giving insights on the state of the art solutions and on the most promising directions of research. In particular, the content is organized in four Sections as follows:

- In Sect. 2, we provide some basic information about 3D face acquisition techniques, motivating and discussing the issues related to the preprocessing of the captured 3D face scans;
- The 3D face datasets that are most widely used as reference for comparing different solutions are reported in Sect. 3. In the same Section, we also report the most commonly used performance indicators for evaluating and comparing 3D face recognition approaches;
- Recent 3D face recognition techniques are reported in Sect. 4, where we distinguish between methods devised for cooperative scenarios, that are mainly targeted to be robust to expression variations, and solutions that are also capable to address missing parts and occlusions as can occur in non-cooperative scenarios;
- Finally, in Sect. 5 we draw conclusions and indicate some of the current and future research directions.

2 3D Face Acquisition and Preprocessing

In the last few years, technologies for 3D acquisition have rapidly advanced with many new 3D scanner devices released for the purpose of acquiring 3D scans of real objects. The specific characteristics of the acquisition device in terms of scans resolution and capability to acquire static or dynamic scenes directly influence any application targeting the analysis of 3D faces. We also observe that 3D sensors produce point clouds as output of the scanning process. Points have xyz coordinates in the 3D reference system which, typically, has the sensor at the origin, the z axis directed outward from the sensor, the y axis directed in the vertical direction, and the x axis obtained by the cross-product between xy. These 3D points can then be triangulated to produce a mesh, or projected to the plane orthogonal to the z axis so as to derive a depth map of the scanned scene.

Independently from the specific device used for the acquisition of 3D face scans, the output of the scanning process is affected by noise and clutter due to spurious acquisition of parts of the scene that have no interest for face analysis (for example, hair, shoulder, neck, ear, etc.). In addition, the acquired faces can also show an arbitrary pose that needs for some normalization procedure.

In the following, we first present some technological solutions for acquiring 3D scans of the face, then we give a summary of the preprocessing solutions that are typically applied to acquired 3D face scans before any subsequent analysis.

2.1 *Laser Scanners*

Laser scanners are capable to acquire high-resolution 3D models by sweeping a plane of laser light across the field of view. The movement of this light stripe is obtained by rotating a mirror which is controlled with high precision by a galvanometer. This laser light is reflected from the face surface so that it can be observed by a single frame captured by the CCD camera. This idea is summarized in Fig. 2. Full 3D face models can be constructed by merging multiple scans of the same subject.

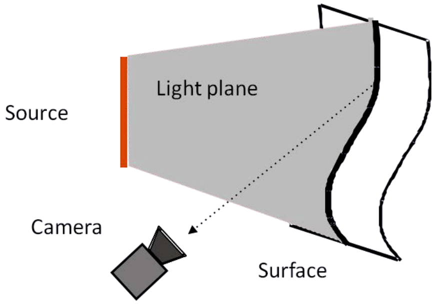


Fig. 2 Basic idea of the laser scanner acquisition. A light stripe (the red line in the figure) is swept across the field of view. The laser light is reflected from the face surface and is observed by the CCD camera

As an example, the main characteristics of the Konica Minolta 3D Vivid 910 are listed below [35]:

- The capture speed varies between 2.5s (at full resolution) to 0.3s in fast mode (lower resolution);
- The device measures a map of 640×480 individual points per scan thus producing a lattice of about 300,000 vertices (76,000 in fast mode). A color image is captured with a very short time lapse by the same CCD (at the resolution of $640 \times 480 \times 24$);
- The geometry accuracy captures the spatial location of 3D points with high precision ($x \pm 0.22mm$, $y \pm 0.16mm$, $z \pm 0.10mm$). However, the device is sensible to regions that do not reflect the laser light (black regions like the eyebrows);
- The operating specification indicates best performance when the scanned surface is at a distance in a range from 0.6 to 2.5m (optimal depth of field is from 0.6 to 1.2m).

2.2 *Structured Light Scanning*

Structured light scanners project a spatio-temporal pattern of light (which includes points or lines) on a surface. The projected pattern is structured in a way such that two cameras viewing the patterns from a slightly different position can triangulate the positions of the same points of the pattern in the two images to extract 3D scene properties. This constitutes a very popular method in computer vision and industrial applications since it also avoids problems of 3D estimation in scenes with complex texture. The technology for structured light scanners has rapidly evolved resulting into devices for static acquisition in high-resolution, and devices capable to perform dynamic real-time acquisition in 3D, though at a lower-resolution. In the following, we illustrate the main characteristics of two of such devices.

2.2.1 Static Acquisition

The 3dMD structured light scanner [1], is a high resolution scanner specifically devised for 3D face acquisition in tasks where very high accuracy and short acquisition time are required, such as in medical, dental, biometrics, engineering, and research applications. The main characteristics of the 3dMD scanner are as follows:

- The coverage angle of the device for face capture is of 180-degree (ear-to-ear);
- The capture speed is of about 1.5ms at the highest resolution;
- The geometry generated by the device is constituted by one continuous point cloud produced from the two stereo camera viewpoints, which eliminates the data errors associated with merging/stitching data sets together;
- The geometry accuracy is lower than 0.2mm RMS or better;
- The produced face scans have a 3D mesh with about 50,000 vertices and 100,000 facets, and a texture stereo image with a resolution of 3341×2027 pixels;
- The operating specifications indicate best performance when the scanned surface is at a distance in a range between 0.6 to 1.2m.

The acquisition process of the 3dMD structured light scanner is summarized in Fig. 3. In (a) and (b) the images acquired by the left / right infrared and RGB cameras are shown, respectively. The spatial pattern projected by the structured light can be observed in the infrared images. In (c) the reconstructed 3D face model obtained from the images in (a) and (b) is reported. A detail of the 3D mesh is shown in (d).

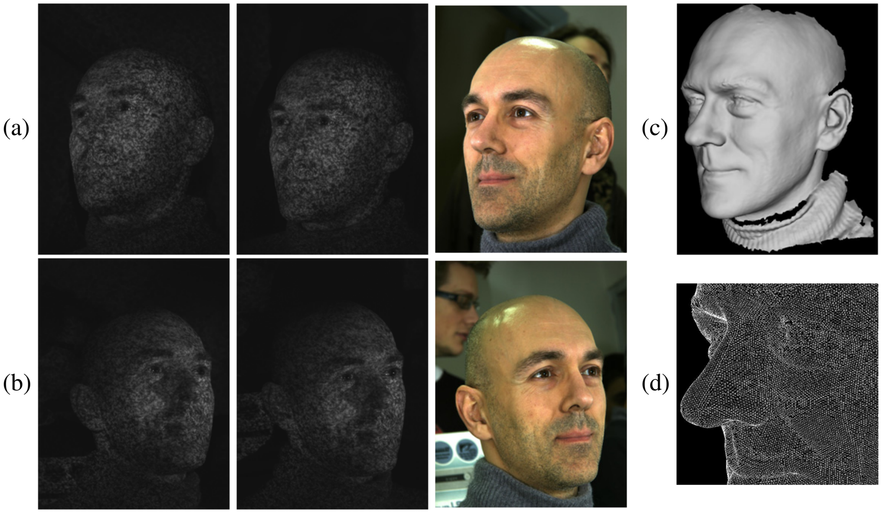


Fig. 3 3dMD acquisition: (a)-(b) Face images with the projected structured light pattern acquired by the infrared camera and by the RGB camera, left and right respectively; (c) 3D reconstructed face model; (d) Particular of the 3D mesh

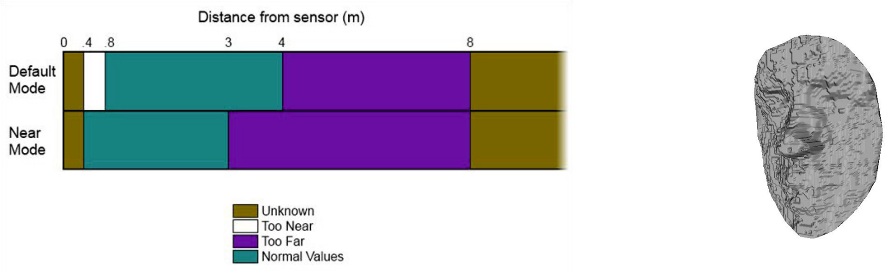
2.2.2 Dynamic Acquisition

The MS Kinect [34] is a dynamic structured light scanner which is capable to acquire a stream (video) of depth images thus opening the way to dynamic analysis of temporal sequences in 3D. This input device is commercialized by Microsoft for the Xbox 360 video game console and developed by PrimeSense with both proprietary and open source drivers. In particular, the Kinect depth sensor is a system-on-chip that provides real-time depth images of a scene through a calibrated stereo pair exploiting a near-infrared light emitter and near-infrared light CMOS. With respect to other 3D scanning devices, Kinect is characterized by a low cost and simplicity of use which make its potential market of increasing importance, though its resolution is still lower than that exhibited by static scanners or by much costly dynamic depth scanners. The acquisition specification of the Kinect are listed below:

- The nominal geometry accuracy is of $1cm$ depth at $2m$ of distance;
- Depth images at a resolution of 640×480 and 16-bits are captured at a speed of around 25-30 frames per second (fps);

- RGB color images are synchronized with depth images and captured at a resolution of 640 × 480 and 24-bit (RGB at 1280 × 960 is also possible, but at 12fps);
- The operative range is between 0.8 to 4m (see Fig. 4(a)).

The Kinect for Windows sensor expands the possibilities with the so called "Near Mode," which enables the depth camera to see objects as close as 40cm in front of the sensor. Fig. 4(a) summarizes the operative range of the device both for the default and the near mode. This increases the possibility to use the dynamic scanners for 3D face analysis in real-time. This is shown in Fig. 4(b), where the low resolution face scan of an individual depth frame of the dynamic stream acquired by the sensor is reported. As can be observed, the resolution of these scans is still too low for permitting accurate identity recognition and can be more useful for macro-expressions recognition. However, methods that permit the increment of the resolution by fusing together depth information from consecutive frames can open the way also to face recognition applications (see [11] for further reading on this point).



(a) (b)

Fig. 4 Kinect sensor: (a) Operative range when the sensor is used in the default and near mode, respectively; (b) The 3D face scan obtained by rendering a depth frame in a dynamic sequence

2.3 Preprocessing

Different sources of noise affect acquired 3D face scans, such as holes, spikes, clutter, etc. In particular, some sources of noise, like holes and spikes are more accentuated in acquisitions performed with laser scanners due to the reflective nature of their acquisition process which is sensible to dark regions. Instead, clutter and unwanted parts are common to any acquisition technique. In general, preprocessing is required to remove holes, spikes, and to fill missing parts. Then, detection of some face landmarks is usually required in order to separate the face region from the unwanted parts of the acquired scene. Finally, for many techniques, pose normalization is also required. A typical preprocessing chain is summarized in Fig. 5. In the following, some guidelines to clean the scans from noise effect, and to detect landmarks and perform pose normalization are discussed.

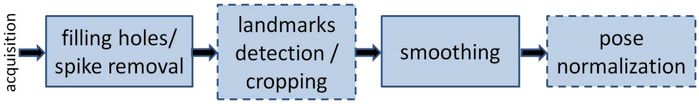


Fig. 5 Typical preprocessing operations applied to the acquired 3D data

2.3.1 Noise Removal

The noise removal step is discussed in several works on 3D face recognition as a preliminary and necessary step. A complete processing chain with solutions for holes filling and spike removal is presented in [41]. In this approach, holes filling is performed by using cubic interpolation of horizontal slices of the face; Spike removal is obtained by median filtering, where the 3D coordinates of each point are replaced with the median of the 3D coordinates of neighboring points. In many approaches, a final step of smoothing step is performed using Laplacian filtering or the convolution with a discrete Gaussian kernel [61].

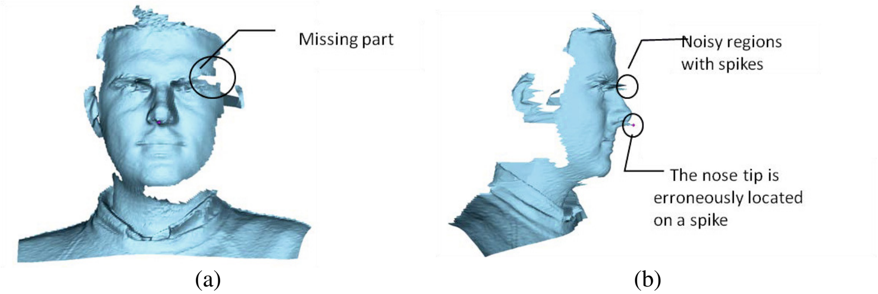

Missing part
(a)
Noisy regions with spikes
The nose tip is erroneously located on a spike
(b)

Fig. 6 Example of noisy acquisition: (a) Missing parts in the eyebrows region; (b) Some spikes are evidenced in the eyes region; A spike in the nose region can determine a wrong localization of the nose tip

As an example, Fig. 6(a) shows the effect of holes and missing parts due to dark regions in a 3D face scan acquired with a laser scanner; In Fig. 6(b) the effect of spikes are evidenced for the same scan shown in (a). It can be noticed, as spikes are concentrated in regions of the face, like the eyes (due to the eyelid), or the region jut under the nose. As shown in (b), the effect of spikes can alter the detection of facial landmarks (in the example, the nose tip is wrongly detected on a spike).

As discussed in the introduction of Sect. 2, the output of the scanning devices is typically in the form of a point cloud where points have no predefined distances among them. In some cases, it is useful to resample the scanned points in order to define a uniform square grid [41]. This can be easily performed by looking to the depth value of each point (i.e., the z-coordinate) as the value of a function in the *xy* variable (that is, $z = f(x,y)$), and interpolating the values of this function in the points of a regular grid in the *xy* plane.

2.3.2 Landmarks Detection

Anthropometric studies have given evidence that Euclidean and geodesic distances computed between 47 landmarks (fiducial points) of the face suffice to discriminate between different subjects [28]. However, only few fiducial points can be reliably detected automatically: *g-glabella*, *n-nasion*, *en-endocanthion*, *ex-exocanthion*, *or-orbital*, *prn-pronasal*, *sn-subnasal*, *al-alare*, *ch-cheilion*, *pg-pogonion*, *gn-gnathion*, *go-gonion*, *me-menton*.

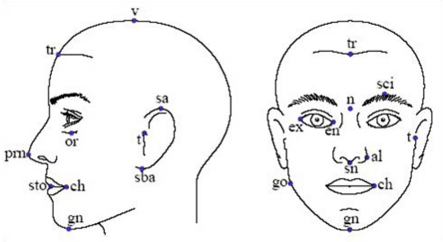


Fig. 7 Some of the landmarks that most characterize the human face. Euclidean or geodesic distances computed between pairs of such landmarks can be used to discriminate the identify of different subjects

The nose tip (*prn-pronasal* in Fig. 7) is considered as the most easily detectable and stable facial landmark. The majority of 3D face recognition approach require at least this point in order to perform face cropping aiming to restrict the 3D scan to the face area or to perform some form of normalization, alignment or preprocessing of 3D faces. For example, in [41] a coarse to fine geometric approach is proposed for nose tip detection that iteratively performs horizontal slicing of the face and detects the triangle corresponding to the nose section. The same coarse to fine strategy is used to identify the inflection points at the base of the nose (*alare* points): The two intersection points of the circle centered on the horizontal slice of the face that includes the nose tip are used to approximate the position of the alare points. In [29], 10 facial landmarks are automatically detected with sufficient accuracy. Given a depth image in the form $(x,y,z(x,y))$, the Gaussian surface curvature K, the mean surface curvature H, and the two principal curvatures k_1, and k_2 are computed from the first and second partial derivatives:

$$K = \frac{z_{xx}z_{yy} - z_{xy}^2}{1 + z_x^2 + z_y^2} \;, \quad H = \frac{z_{xx}(1+z_y^2) + z_{yy}(1+z_x^2) - 2z_xz_yz_{xy}}{(1+z_x^2+z_y^2)^{3/2}}$$

$$k_1, k_2 = H \pm \sqrt{H^2 - K}\,. \qquad (1)$$

The signs of the Gaussian and the mean curvature values help to identify differently shaped regions of a surface. The regions with $K > 0$ are *elliptic*, those with $K < 0$ are *hyperbolic*, and those with $K = 0$ are either *planar* or *cylindrical*. Regions of the surface with $H > 0$ are *concave*, while those with $H < 0$ are *convex*.

The region surrounding the tip of the nose has the highest elliptic Gaussian curvature $K > 0$, and the highest convex elliptic Gaussian curvature $H < 0$. For detecting the nose tip the algorithm searches for the point with the maximum elliptic Gaussian curvature within a $96mm \times 96mm$ central region of each face, which surrounds the initial estimate of the nose tip (obtained using ICP alignment with a template or maximum z value). More elaborated operations permit the detection of the following points (see also Fig. 7 for the name of facial landmarks): *al-alare* (i.e., corners of the nose), *ch-cheilion* (i.e., mouth corners), *en-endocanthion* and *ex-exocanthion* (i.e., inner and outer eyes corners), and the *n-nasion* (i.e., the point at the base of the nose and between the two eyes).

In [5], an automatic approach is presented which permits automatic detection of 9 facial landmarks with sufficient accuracy. Using depth images of the face, the nose tip and the alare points are identified with an approach similar to that described in [29]. Following the anthropometric proportions of the face proposed in [28], these points are used to define the regions of the face that include the inner and outer eyes corners and the mouth corners. In practice, a separate region is defined for each landmark, and each of these regions is used as search windows for *en*, *ex* and *ch* points, respectively. In particular, in each search window the Scale Invariant Feature Transform (SIFT) detector algorithm [38] is run, and the SIFT point detected at the highest scale is retained as landmark of the search window. An example is shown in Fig. 8 for the right mouth. The leftmost image shows the 3D surface of the search window for the right mouth corner (*ch*). The rightmost image shows, with red circles, the SIFT keypoints detected on the depth map of the search window. The *ch* landmark corresponds to the keypoint detected at the highest scale (i.e., the keypoint represented in the Figure with the longest radial segment colored in blue).

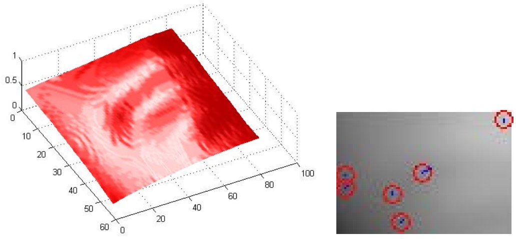


Fig. 8 The 3D search window used for the detection of the right ch, and the SIFT keypoints detected on the depth map of the search window are shown on the left and right, respectively. The ch landmark is the keypoints detected at the highest scale (represented with the longest radial segment in blue)

2.3.3 Pose Normalization

Pose normalization is the most difficult and time consuming preprocessing operation. It is needed in the case descriptors extracted from 3D face scans and used for face analysis are not rotation invariant. The problem is stated as follows: Given the coordinates of a set of points measured in two Cartesian coordinate systems (left, right) find the rigid transformation T, between the two systems so that for corresponding points P_r and P_l we have $P_r = T(P_l)$. The cardinality of both data sets is usually not equal.

Since pairing between points is unknown, iterative solutions are used. These solutions require initialization and only guarantee convergence to local optimum. Initialization is often performed by using facial landmarks. Iterations alternate between a matching step and a transformation computation step based on an analytic solution. A solution to this problem is given by the Iterative Closest Point (ICP) algorithm that was independently developed by several authors [13, 20, 66]. In each iteration step, ICP selects the closest points as correspondences and calculates the rotation and translation (R,t) to find alignment by minimizing the equation:

$$E(R,t) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} w_{ij} \cdot \parallel m_i - R \cdot d_j + t \parallel , \qquad (2)$$

where M is the model set $\{m_i\}_{i=1}^{N_m}$, and D is the data set $\{d_j\}_{j=1}^{N_d}$, with N_m and N_d points, respectively; w_{ij} coefficients permit different weights for each points pair. The above equation converges if the starting positions of the two sets of points are close enough; differently it can converge to a minima. Due to this, in some cases the scans undergo to a preliminary rough alignment based on the correspondence of a few landmarks of the face [12]. In summary, the ICP algorithm is as follows:

Initialization:

1. Set cumulative transformation, and apply to points.
2. Pair corresponding points and compute similarity (e.g., root mean square distance).

Iterate:

1. Compute incremental transformation using the current correspondences (i.e., analytic least squares solution).
2. Update cumulative transformation, and apply to points.
3. Pair corresponding points and compute similarity.
4. If improvement in similarity is less than threshold ε or number of iterations has reached the maximum number L terminate.

The worst case complexity of the algorithm is $O(N_m N_d)$ for two data sets of size N_m and N_d, respectively. The complexity can be reduced to $O(N_m \log N_d)$ using a kD-tree to speed up the search of nearest neighbor.

3 3D Face Datasets and Performance Evaluation

The recent achievements of 3D face analysis techniques are also due to the availability of large and challenging 3D face datasets with a large variability in terms of gender, ethnicity and age of the acquired subjects, acquisition devices, conditions of acquisition (including subjects with varying expressions, missing parts and occlusions). These challenging datasets have permitted the development of advanced solutions an their comparison on common benchmark datasets. More recently, a growing interest is captured also by *dynamic 3D acquisition* (3D plus time, or 4D), which allows temporal sequences of 3D scans to be acquired (the MS Kinect is an example of such devices). In the following, we restrict our discussion to static 3D face datasets that are used for the purpose of face recognition or facial expression recognition. In the last part of the Section, we present the performance indicators that are more commonly used in order to present and compare the verification and identification accuracy of 3D face recognition approaches.

3.1 3D Face Datasets

Several 3D face databases have been made publicly available for testing algorithms that use 3D data to perform face modeling, analysis and recognition. These databases have progressively included face scans of subjects that exhibit non-neutral facial expressions and non-frontal poses. Some of these datasets have been originally promoted within face recognition competitions. In particular, the Face Recognition Grand Challenge (FRGC) initiative directed by NIST provided common data sets to be used as a reference for training (FRGC v1.0) and evaluation (FRGC v2.0); a 3D face matching contest was launched in 2005, with the final results published in 2006 [49]. The SHape REtrieval Contest (SHREC) initiative developed in the framework of the Aim@Shape project at the European Commission organized a special track for 3D face retrieval in 2008, providing a data set for evaluation (SHREC08 data set) that is smaller, but includes stronger face expressions; final results were published in late March 2008 [24]. A second SHREC initiative for evaluating 3D face recognition approaches in the presence of 3D scans with missing parts was launched in 2011, with the results available in March 2011 [59]. Table 2, reports the more general datasets that are currently available for the task of 3D face recognition. For each dataset, information about the sensor used during acquisition, the total number of subjects and acquired scans are reported. Notes about the presence of scans with missing parts or occlusions and the availability of 2D texture images associated to the 3D data are also given.

Table 2 Publicly available datasets for 3D face recognition

Database	Sensor	n° subjects	n° scans	Missing data	Occlusions	Texture
Bosphorus	structured light	105	4,666	Yes	Yes	Yes
BU-3D	structured light	100	2,500	No	No	Yes
FRGC v1.0	laser	275	943	No	No	Yes
FRGC v2.0	laser	466	4,007	Yes	No	Yes
FU	structured light	53	212	Yes	No	Yes
Gavab	laser	61	549	Yes	No	No

Table 3 Main characteristics of the most used 3D face databases that include non-neutral facial acquisitions

Dataset	Expressions	Pose
BU-3DFE	anger, disgust, fear, happiness, sadness, surprise	frontal
Bosphorus	action units, anger, disgust, fear, happiness, sadness, surprise	13 yaw and pitch rotations, hand, eyeglasses
FRGC v2.0	not categorized: disgust, happiness, sadness, surprise, puffy	small changes
Gavab	smile, laugh, random	up, down, left, right

Table 3 summarizes the characteristics of some of the most known and used 3D face databases that include subjects with non-neutral facial expressions. These datasets can be used both to test the robustness of face recognition algorithms with respect to the presence of expressions and to investigate solutions for classifying facial expressions from 3D face scans. In the following sections, some more details on these datasets are given.

3.1.1 The Face Recognition Grand Challenge Database (FRGC)

The FRGC data set [48] includes 3D face scans partitioned into three subsets, namely, the *Spring2003* subset, also known as FRGC v1.0 (943 scans of 275 individuals), and the *Fall2003* and *Spring2004* subsets (4,007 scans of 466 subjects in total) that are commonly identified as the FRGC v2.0 dataset. Face scans are acquired with a Konica-Minolta Vivid 910 laser scanner and given as matrices of 3D points of size 480 × 640, with a binary mask indicating the valid points of the face. Due to different distances of the subjects from the sensor during acquisition, the actual number of points representing a face can vary. Individuals have been acquired with frontal view from the shoulder level, with very small pose variations. Considering the FRGC v2.0, about 59% of the faces have neutral expression, and the others show moderate non-neutral expressions of disgust, *happiness*, *sadness*, and *surprise*. Some scans include small occlusions due to hair. FRGC guidelines suggest using the *Spring2003* for training and the remaining two sets for validation. This dataset has been extended into the *University of Notre Dame biometric database* (UND) [58] with the aim to include scans of side views of the subjects (from 45 to about 90 degrees) that can be also used for identity recognition based on the ear shape.

3.1.2 The Bosphorus 3D Face Database

The Bosphorus database has been collected at the Bovgazicci University and made available during 2008 [55]. It consists of the 3D facial scans and images of 105 subjects acquired under different expressions and various poses and occlusion conditions. Occlusions are given by hair, eyeglasses or predefined hand gestures covering one eye or the mouth. Many of the male subjects have also beard and moustache.

The majority of the subjects are Caucasian aged between 25 and 35, with a total of 60 males and 45 females. The database includes a total of 4,666 face scans, with the subjects categorized into two different classes:

- 34 subjects with up to 31 scans per subject (including 10 expressions, 13 poses, four occlusions and four neutral faces);
- 71 subjects with up to 54 different face scans. Each scan is intended to cover one pose and/or one expression type, and most of the subjects have only one neutral face, though some of them have two. Totally, there are 34 expressions, 13 poses, four occlusions and one or two neutral faces per subject. In this set, 29 subjects are professional actors/actresses, which provide more realistic and pronounced expressions.

Each scan has been also manually labeled with 24 facial landmarks such as *nose tip*, *inner eye corners*, etc., provided that they are visible in the given scan.

3.1.3 The Gavab Database

The Gavab database [43] is characterized by facial scans with very large pose and expression variations and noisy acquisition[1]. It includes 3D face scans of 61 adult Caucasian individuals (45 males and 16 females). For each individual, nine scans are taken that differ in the acquisition viewpoint and facial expressions, resulting in a total of 549 facial scans. In particular, for each individual, there are two frontal face scans with *neutral* expression, two face scans where the subject is acquired with a *rotated* posture of the face (around ±35° looking up or looking down) and neutral facial expression, and three frontal scans in which the person *laughs*, *smiles*, or shows a *random* expression. Finally, there are also a right side and a left side scans nominally acquired with a rotation of ±90° left and right. This results in about 67% of the scans having a neutral expression, but just 22% having neutral expression and frontal pose. Modified scans of this database have been used as data for the SHREC 2008 *Shape Retrieval Contest of 3D Face Scans* [24], and to test face recognition accuracy in several other papers as well as to test recognition performance in the case parts of the face scans are missing [12, 25, 30].

3.1.4 The Binghamton 3D Facial Expression Database (BU-3DFE)

The BU-3DFE database has been recently constructed at the *Binghamton University* [65]. It was designed to provide 3D facial scans of a population of different subjects each showing a set of prototypical emotional states at various levels of intensities. There are a total of 100 subjects in the database, divided between female (56 subjects) and male (44 subjects). The subjects are well distributed across different ethnic groups or racial ancestries, including *White*, *Black*, *East-Asian*,

[1] The database is publicly available at the following address:
http://gavab.escet.urjc.es/index_en.html

Middle-East Asian, *Latino-Americans*, and others. During the acquisition, each subject was asked to perform the six basic facial expressions defined by Ekman, namely, *anger* (AN), *disgust* (DI), *fear* (FE), *happiness* (HA), *sadness* (SA), and *surprise* (SU), plus the *neutral* (NE) one. Each facial expression has four levels of intensity, respectively, *low*, *middle*, *high* and *highest*, except the neutral facial expression. Thus, there are 25 3D facial expression scans for each subject, resulting in 2500 3D facial expression scans in the database. Each of the 3D facial expression scan is also associated with a raw 3D face mesh, a cropped 3D face mesh, a set of 83 *manually annotated* facial landmarks, and a facial pose vector. These data give a complete 3D description of a face under a specific facial expression. The landmarks are distributed in correspondence to the most distinguishing traits of the face, that is, *eyes*, *eyebrows*, *nose* and *mouth* (plus some landmarks on the face boundary). Finally, since these 3D data are built from a stereo-camera system that reconstructs the 3D shape of the face from two different left/right views, two 2D color images of the left/right view of the face are also acquired and can be used for multi-modal 2D/3D face analysis.

3.1.5 The Florence 2D/3D Face Dataset

The 2D/3D Florence face dataset (UF-2D/3D) has been constructed at the Media Integration and Communication Center of the University of Florence [4][2]. The dataset consists of high-resolution 3D scans of human faces along with several video sequences of varying resolution and zoom level. Each subject is recorded under various scenarios, settings and conditions. This dataset is being constructed specifically to support research on techniques that bridge the gap between 2D, appearance-based recognition techniques, and fully 3D approaches. It is designed to simulate, in a controlled fashion, realistic surveillance conditions and to test the efficacy of exploiting 3D models in real scenarios. The 3D part of the dataset (UF-3D), currently includes 53 subjects (14 females and 39 males, numbered from *subject001* to *subject053*) of Caucasian ethnicity. The age of the subjects ranges from 20 to 60, with the majority of the subjects (28) being student at the School of Engineering of the University of Florence, aged between 20-30 years. The 3D scans of each subject are acquired in the same session and include two frontal scans with neutral expression (named as *frontal1* and *frontal2*), and two scans where the subject is rotated of 90° on the left and right side (named *left* and *right*, respectively). In all the acquisitions, the subjects are required to assume a neutral expression, though some scans exhibit moderate, involuntary, facial expressions. The *3dMD face system* [1] scanner has been used in the acquisition, which produces one continuous point cloud from two stereo cameras with a capture speed of about $1.5ms$ at the highest resolution, and a geometry accuracy lower than $0.2mm$ RMS.

[2] The database is publicly available and can be accessed upon request from the following address: http://www.micc.unifi.it/masi/research/ffd/

3.2 Performance Indicators

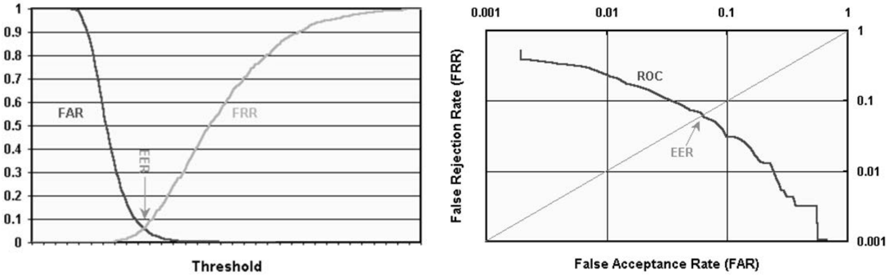
Different performance indicators are used to evaluate the performance of face recognition methods in *verification* or *identification* scenarios.

In general, face verification is solved by measuring the similarity between a probe scan and a reference scan and comparing it against a similarity threshold: values of similarity that are greater than the threshold correspond to two matching scans and permit subjects authentication; conversely, similarity values lower than the threshold are interpreted as corresponding to non-matching scans, thus denying subjects authentication. Performance indicators for such face verification methods measure the capability to correctly authenticate qualified subjects, while rejecting fraudulent attempts of authentications. To this end, the two following quantities are computed:

- *False Rejection Rate* (FRR) – Average of: Number of rejected verification attempts for qualified person / Number of all verification attempts for qualified person;
- *False Acceptance Rate* (FAR) – Average of: Number of successful independent fraudulent attempts against a person / Number of all independent fraudulent attempts against a person.

Both these quantities vary in the range [0,1]. As shown in Fig. 9(a), FRR and FAR are strictly correlated. As the similarity threshold increases FRR grows while FAR lowers, so that by adjusting the FAR/FRR ratio the sensitivity of the system can be adapted. An high similarity threshold defines a conservative approach to recognition; as the threshold is lower a loose verification is obtained. *Equal Error Rate* (EER) represents the system intrinsic error at which FRR = FAR. In the practice of face verification evaluation, the *Receiver Operating Characteristic* (ROC) curve (also referred to as *Detection Error Tradeoff* curve) is used to represent the correlation between FRR and FAR at a given threshold. The Operating Point (OP) is defined in terms of FRR achieved for a fixed FAR. As an example, a ROC curve is reported in Fig. 9(b). In many studies of face recognition, the *True Acceptance Rate* (TAR), computed as $1 - FRR$, is used instead of the FRR, and the TAR at 0.001 FAR is reported as synthetic performance indicator.

Face identification is performed by measuring the similarity between a probe scan and a set of scans included in the gallery: The identity of the probe scan is associated to the subject whose gallery scans has the greatest similarity with the probe. In general, it happens that the gallery scan whose identity corresponds to the probe scan has a similarity score that ranks the scan in a position *k* of the overall list of sorted similarity scores (with *k* varying from 1 to N, being N the number of scans in the gallery). According to this, the performance of face identification approaches are typically summarized by the *Cumulative Matching Characteristic* (CMC) curve. The curve plots the rate of correct recognition (i.e., probability of identification) at different ranks. The curve is cumulative, so it results monotonically not decreasing and it reaches 1 at one of the rank *k* (at the highest rank in the worst case).



(a) (b)

Fig. 9 Performance indicators: (a) Example of FRR and FAR at varying threshold; (b) Example of ROC curve

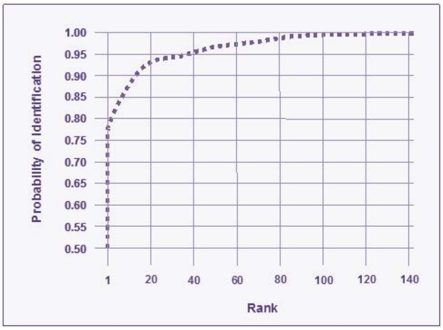


Fig. 10 Performance indicators: Example of CMC curve. The probability of recognition (i.e., recognition rate) is reported on the vertical axis; the rank *k* on the horizontal axis

4 3D Face Recognition

So far, the most promising use of 3D face scans is in performing facial recognition. In doing so, a preliminary distinction of existing solutions is between methods that are devised to operate in *cooperative* scenarios, and methods that can also provide accurate recognition in *semi-cooperative* or fully *non-cooperative* scenarios. In the following, we review some of the main approaches in both these categories.

4.1 Cooperative 3D Face Recognition

3D face recognition has first been introduced in cooperative scenarios, where both gallery scans (i.e., scans to be included in a reference set and acquired from subjects whose identity is known) and probe scans (i.e., scans to be compared with the gallery set in order to perform recognition, and acquired from subjects whose identity is unknown) are acquired using a cooperative protocol. This defines the acquisition conditions giving some constraints (e.g., subjects stay fix in front of the scan, showing a neutral facial expression and without wearing any cap, scarf or glasses) and corresponds to the cases that are encountered in face verification contexts (i.e., subjects cooperate to be recognized so as to pass some security control or gain access to some service). In this scenario, facial scans exhibit a frontal pose, with very small occlusions or missing parts, with facial expressions representing the main source of variability in the acquired scans. And though facial expressions shown by the subjects are typically small or moderate, they demand for methods capable to perform accurate recognition by smoothing the effects due to facial variations induced by expression changes. A list of the solutions proposed can be derived from the survey of Bowyer et al. [15], and the literature reviews of [8, 33, 41]. In the following, we give some more details on some methods.

4.1.1 Existing Approaches

According to an agreed classification [67], the approaches for 3D face recognition can be distinguished as: *holistic* (or *global*), which performs face matching based on the whole face; *region-based* (or *local*), that partition the face surface into regions and extract appropriate descriptors for each of them; *hybrid* and *multi-modal*, that combine different approaches such as holistic and region-based, or perform both 2D and 3D matching separately and fuse the two matchings together to achieve better recognition accuracy. Generally speaking, holistic methods are sensitive to face alignment. Moreover, since they take global face measures, they tend to treat face differences that are due to different facial traits and non-neutral expressions in the same way. The performance with these methods can also be very much impaired if the 3D face includes elements like hair, ears, and neck. Region-based approaches promise much higher effectiveness in that, at least in principle, they can apply different processing to distinct face regions and therefore filter out those regions that are mostly affected by expression changes or spurious elements. Nevertheless, they are also sensitive to face alignment and useful face regions are hard to detect automatically. Their performance depends on local features and differences in resolution. Hybrid and multi-modal approaches provide the highest accuracy, but at the expense of a greater architectural complexity. They are especially suited for verification, less for identification in that do not permit easy indexing.

Holistic Methods

Among the holistic methods, several authors have attempted to find the main distinguishing elements of the faces from the direct analysis of face depth images, after realignment to a reference face model. Principal Component Analysis (PCA) was applied to depth images and to both depth and color image channels. Conformal transformations have been used in [44, 60] among others. Since conformal mapping is a one-to-one angle preserving transformation, 3D surface matching is reduced to a simpler 2D image-matching problem. In particular, in [44], a region of interest of the face, defined as the intersection between a sphere centered on the nose tip and the 3D face, was mapped onto an isomorphic planar circle and eigenface analysis was used to compare faces. With this method, the authors reported 95 percent rank-1 recognition rate on FRGC v1.0 data set, with manual alignment of 3D face models. In [53], 3D faces were instead represented through iso-depth lines projected onto the base plane. Then, shapes of the iso-depth lines were compared, exploiting differential geometry for 2D planar-closed curves. Since face expressions may induce strong alterations of the iso-depth lines, this approach is likely to be very sensitive to expression changes. A similar approach was followed in [56]. There, the authors used sample points taken at the intersection between contour curves and radial profiles originated from the nose tip and calculated the Euclidean distances between corresponding points of different faces.

As a different approach, some authors have proposed representing 3D faces as points in low-dimensional feature spaces. The 3D coordinates of face points were therefore encoded through transformations, and dimensionality reduction was applied in the feature space. In [60], 3D spherical Gabor filters were used to extract a view invariant representation of 3D facial models. The authors used a modified version of the Hausdorff distance in order to improve the robustness of matching in the presence of self-occlusions. However, tests were performed on a too small test set to assess the effectiveness of the approach. In [16], face models were represented with the geometric moments up to the fifth order computed for the 3D face canonical surface. Canonical surfaces were obtained from face surfaces by warping according to a topology preserving transformation so that the Euclidean distance between two canonical surface points is equivalent to the geodesic distance between the corresponding points of the face surface. However, while the effect of expressions is attenuated, a similar attenuation also occurs for discriminating features such as eye sockets and nose. Some limitations of the method were indeed removed in [17]. Other authors have proposed exploiting the full 3D geometrical information of the face model and performed matching according to pointwise registration, avoiding calculation of features and the consequent loss of information. In [39], rules of transformation from neutral to generic expressions were learned from a training set so as to create synthetic 3D models for any expression. The ICP algorithm was then used to align the synthesized models to an input model, handling adaptation to both pose and expression simultaneously. Elastic registration with morphable models was used in [3, 33, 45]. In particular, in [33] and [45], the points of an annotated 3D face reference model were shifted according to elastic constraints so as to match the corresponding points of 3D target models in a gallery. Similar morphing was performed for each query face. Then, face matching was performed by comparing the wavelet coefficients of the deformation images obtained from morphing. In [3], a 3D morphable model was learned from face models with neutral expression and adapted to gallery and query faces using a variant of the nonrigid ICP algorithm. Distances between the deformation coefficients were used to assess matching. Although registration-based methods support accurate face matching, they perform matching iteratively and are extremely expensive from the computational viewpoint. Attempts to reduce the computational complexity have been proposed in [39, 64].

Region-Based Methods

Holistic approaches reveal some limitations in performing accurate recognition in the presence of facial expressions. Local approaches can be distinguished by the way in which face regions are detected and segmented. These methods also try to smooth the effects of facial expressions by processing differently the regions of the face that are most affected by expression changes. In [23], Log-Gabor templates were used to break a single-range image into a predefined number of overlapping spatial regions at three different frequency scales. These observations were each classified individually and then combined at the score level. PCA was applied to the responses of the Log-Gabor filters in each subregion used to reduce the dimensionality. Instead, regions in the proximity of face landmarks were used in [32]. Features were extracted at the landmark regions and face matching was performed according to Hierarchical Graph Matching, with graph nodes positioned at the landmarks. In [63], multiple face regions are originated by intersecting 3D face scans with spheres of increasing radius centered on the middle point between the nose tip and the nasion. In [19], multiple overlapping regions around the nose are segmented and the scores of ICP matching on these regions are combined together. This idea is extended in [26] by using a set of 38 regions that densely cover the face and selecting the best-performing subset of 28 regions to perform matching using the ICP algorithm. A further improvement of the approach is proposed in [27] by considering a multi-instance enrollment of gallery scans with multiple expressions (experiments are provided using up to five scans per individual). Accordingly, up to 140 ICP region matches are required to compute the similarity between a probe scan and the scans representing an enrolled individual. Robustness to non-neutral facial expressions is improved at the cost of a greater computational complexity (matching two scans is reported to take more than 2 seconds in [26] and five times longer in [27]), thus making these approaches more suited to face verification than identification. The idea of using facial regions of the face is also exploited in [52], where the circular and elliptical areas around the nose were used together with forehead and the entire face region for authentication. The Surface Interpenetration Measure (SIM) were used for the matching. Taking advantage of invariant face regions, a Simulated Annealing approach was used to handle expressions. Other methods have performed segmentation of the 3D face into distinct regions according to the values of the curvature function calculated from the face surface [36]. A crucial limitation of curvature-based approaches is the extreme sensibility of curvature values to perturbations of surface points that may occur due to noise, fallacious acquisition, or changes of expressions. In most of the cases, face comparison has been restricted to the comparison of only a few regions, where the effects of expressions are small or null. In particular, in [36] Extended Gaussian Images – that provide a one-to-one mapping between curvature normals and the unit sphere – were created for each convex region and compared by graph matching with relational constraints. The approach in [61], is one of the best performing solutions on the FRGC v2.0. It used a Signed Shape Difference Map (SSDM) computed between two aligned 3D faces as an intermediate representation for the shape comparison. Based on the SSDMs, three kinds of features were used to encode both the local similarity and the change characteristics between facial shapes, namely, Haar-like, Gabor, and Local Binary Pattern (LBP). The most discriminative local features were selected optimally by boosting, and trained as weak classifiers for assembling three collective strong classifiers.

Hybrid and Multimodal Methods

These approaches have shown the best recognition results so far, trying to combine multiple processing paths into a coherent architecture, so as to solve critical drawbacks of individual methods. Among the multimodal methods, in [18], the authors proposed applying PCA to face depth images and 2D face images separately and then fusing the results together. In [39], ICP registration of the 3D face models was combined with Linear Discriminant Analysis applied to 2D face images, to improve the robustness of 2D face matching in the presence of pose and illumination variations. In [14], central and lateral profiles of the face were extracted and compared in both 3D and 2D. In [32], landmark positions used to define the face regions were also detected on 2D texture images obtained with the 3D face scan. One of the best performance on the FRGC v2.0 dataset was obtained with the solution reported in [41] which is both hybrid and multimodal. The authors assembled a fully automated system performing the following steps:

1. Pose correction (of both the 3D model and the corresponding 2D color image provided by the scanner);
2. Automatic region segmentation to account for local variations of the face geometry (by detecting the inflection points around the nose tip);
3. Quick filtering of distant faces using SIFTs and 3D Spherical Face Representation (a quantization of the face point cloud into spherical bins centered at the nose tip);
4. Matching of the remaining faces applying a modified ICP to a few regions of the face (eyes, forehead, and nose) that are less sensitive to face expressions. The similarity scores provided by the two matching engines were fused into a single similarity measure. Performance of the method in terms of TAR at FAR 0.001 are 99.7% and 98.3% for neutral vs. neutral and neutral vs. non-neutral expressions, respectively.

There are nevertheless several considerations to be made about these performance figures. In the FRGC data set, the nose tip of 3D face models is estimated from the 2D images provided by the scanner. To have more precise measures, additional preprocessing must be performed on the 3D models. Due to this, performance measures with this data set account for the capability of the system in both identifying the reference point precisely and performing face recognition effectively. Besides, given the nature of the FRGC data set, performance measures are markedly representative of face recognition in the presence of moderate facial expressions [33]. Data sets with stronger facial expressions to verify invariance to facial expressions in more challenging conditions are nevertheless available, such as the extended data set of [33] (obtained as the integration of FRGC v2.0 with the University of Huston (UH) proprietary database) and the SHREC08 data set. With specific reference to the method of [41], it must also be observed that, although it performs segmentation into face regions, invariance to facial expressions is simply obtained by discarding those face regions that are more affected by expressions. Consequently, many facial details are missed at important parts, such as mouth, chin, and cheeks. This can be very critical in many cases where these details are key distinguishing elements. Moreover, the method requires prefiltering and iterative point-wise registration to match individual models. According to this, descriptors cannot be organized using traditional indexing structures (such as point access methods and metric access methods) to support efficient identification in very large data sets.

4.1.2 Face Recognition in the Presence of Strong Facial Expressions

The SHape REtrieval Contest (SHREC) initiative developed in the framework of the Aim@Shape project at the European Commission organized a special track for 3D face retrieval in 2008, providing a data set for evaluation (SHREC08 data set) that is smaller, but includes stronger face expressions than FRGC v2.0. Final results were published in late March 2008 [24]. The SHREC08 evaluation in the 3D face track was performed on a data set of 3D face scans of 61 adult Caucasian individuals (45 males and 16 females) derived from the Gavab database [43]. For each individual, seven scans are taken that differ in the acquisition viewpoint and facial expressions, resulting in a total of 427 facial scans. In particular, for each individual, there are two frontal face scans with neutral expression, two face scans where the subject is acquired with a slightly rotated posture of the face (looking up or looking down) and neutral facial expression, and three frontal scans in which the person laughs, smiles, or shows a random expression. This results in about 57 percent of the scans having a neutral expression, but just 29 percent having neutral expression and frontal pose. Scans are given as triangular meshes with an average number of 10,000 points. No data were explicitly provided for training or tuning the participants recognition systems. Instead, each participant was allowed to run up to five versions of their algorithms with different tunings of the system parameters. Each result set was computed by measuring the distance of every face scan to any other face scan in the data set. In practice, each result set is organized in a set of ranked lists reporting all of the scans of the data set sorted in increasing values of distance from the query scan.

The best performing approach at the SHREC08 contest was that proposed in [8] (a preliminary version of the approach was presented in [7]). In this approach, all of the points of the face are taken into account so that the complete geometrical information of the 3D face model is exploited, but differently from registration methods where matching is obtained by iterative pointwise alignment; here, the relevant information is encoded into a compact representation in the form of a graph and face recognition is finally reduced to matching the graphs. Face graphs have a fixed number of nodes that, respectively, represent iso-geodesic facial stripes of equal width and increasing distance from the nose tip. Arcs between pairs of nodes are annotated with descriptors referred to as 3D *Weighted Walkthroughs* (3DWWs) [6] that capture the mutual spatial displacement between all the pairs of points of the corresponding stripes and show smooth changes of their values as the positions of face points change. Due to the fixed partitioning into isogeodesic stripes, 3DWWs between stripes are approximately calculated over the same portions of the face for all individuals, thus permitting discrimination between structural differences in face morphology. Besides, according to the property that iso-geodesic distances do not vary too much under facial expressions, 3DWWs are principally calculated over the same set of points of the stripes under any expressions, with limitations to the range of the possible local modifications, and therefore, to the effects of point shifting. This smooths the differences due to different expressions of the same individual. This representation has the great advantage of a very efficient computation of face descriptors and a very efficient matching operation for face recognition. Moreover, the approach appears very well suited for the task of face identification in very large data sets. In fact, face graphs can be arranged in an appropriate index structure so that the efficiency of search is even improved. The method obtained the best ranking at the SHREC08 contest, scoring Recognition Rate of 99.53 percent, Mean Average Precision of 93.49 percent, Mean Average Dynamic Precision of 97.73 percent, and Mean Normalized Discounted Cumulated Gain@5 of 99.03 percent. This approach was further developed in [9], by combining the local approach to 3D face recognition with a feature selection model so as to study the relative relevance of different regions of the face in discriminating between different subjects. This permitted the identification of the relevance of individual stripes, thus restricting the match to the pairs of stripes which are most informative. As a further contribution of this work, it is quantitatively demonstrated that the relevance of facial regions (stripes) changes for different ethnic groups, thus opening the way to further optimizations based on the preliminary recognition of the ethnicity of the subjects.

4.2 Semi-cooperative 3D Face Recognition

In a conventional face recognition experiment, it is assumed that both the probe and gallery scans are acquired cooperatively in a controlled environment so as to precisely capture and represent the whole face. Many of the existing methods followed this assumption, focusing on face recognition in the presence of expression variations and reporting very high accuracy on benchmark databases like the FRGC v2.0 dataset [48]. Differently, solutions enabling face recognition in uncooperative scenarios are now attracting an increasing interest. In such a case, probe scans are

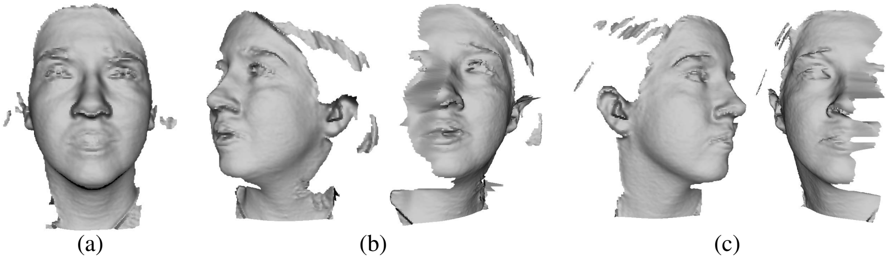


Fig. 11 Facial scans from the *Gavab* database [43]. (a) Frontal face scan with neutral expression. In (b) and (c) the left and right scans of the subject in (a) are given, respectively. The original side scans and their pose normalized frontal views are shown in both cases.

acquired in unconstrained conditions that may lead to *missing parts* (non-frontal pose of the face, see Fig. 11), or to *occlusions* due to hair, glasses, scarves, hand gestures, etc. These difficulties are further sharpened by the recent advent of 4D scanners (3D plus time) [1, 34], capable of acquiring temporal sequences of 3D scans. In fact, the dynamics of facial movements captured by these devices can be useful for many applications [10, 54], but also increases the acquisition noise and the variability in subjects' pose.

4.2.1 Existing Methods

In the following we present semi-cooperative 3D face recognition methods by classifying them as *holistic*, *local* and *point-based*.

Holistic Methods

Global 3D face representations for partial face matching have been proposed in a limited number of works. In [17], a canonical representation of the face is proposed which exploits the isometry invariance of the face surface to manage missing data obtained by randomly removing areas from frontal face scans. On a small database of 30 subjects they reported high recognition rates, but no side scans were used for recognition. Results on partial face matching removing quadrants of FRGC v2.0 probes and using face crops around the nose tip are reported in [62]. This approach relies on the symmetry of the 3D face scans in order to identify the nose tip and register depth maps so as to derive a Pure Shape Difference Map (PSDM) between pairs of matching scans. A Collective Shape Difference Classifier learns off-line the most discriminative local areas from the PSDM and trains them as weak classifiers for assembling a collective strong classifier using the real-boosting approach. Unfortunately, the symmetry hypothesis used for the registration and fiducial points detection is often violated when side views of the face are acquired in uncooperative scenarios. Instead, the experiments are conducted just removing parts of the face after the preprocessing has been performed on the entire scans. The fact that the same part of the face is removed from both probes and gallery models in order to generate the PSDM also reduces the concrete applicability of the approach. The same authors extended the approach in [61], but without providing any experimentation on face data with missing parts. These approaches provide a *global* modeling of both gallery and probe scans, but more successful and scalable solutions use *local* representations of the face.

Local Methods

A possible way to solve locally the problem of missing data in 3D face acquisition is to detect the absence of regions of the face and use the existing data to reconstruct the missing parts. The reconstructed scan can then be used as input to conventional 3D face recognition methods, that assume the entire scan is available. This approach is followed in [22], focusing on face occlusions induced by glasses, scarves, caps, or by the subject's hand. A generic facial model and thresholding on facial surface distances are used to detect occlusions. In this way, the occluded areas are detected and the missing regions are restored using information from the non-occluded parts. Detection experiments were carried out on a proprietary database containing a training set of 132 3D scans with various non-occluded facial expressions and a test set of 76 scans. However, face recognition accuracy was not evaluated. The inter-pose face recognition solution proposed in [47] and extended in [46], exploits the hypothesis of facial symmetry to recover missing data in facial scans with large pose variations. First, an automatic face landmarks detector is used to identify the pose of the facial scan by marking regions of missing data and roughly registering the facial scan with an Annotated Face Model (AFM) [33]. Then, the AFM is fitted using a deformable model framework that exploits facial symmetry where data are missing. Wavelet coefficients extracted from a geometry image derived from the fitted AFM are used for the match. Experiments have been performed using the *University of Notre Dame* (UND) database [58], with the FRGC v2.0 gallery scans and side scans with 45° and 60° rotation angles as probes. Since it is based on the left/right facial symmetry, this solution can work as long as half of the face with respect to the yaw axis is visible in the scan.

Tackling the problem from an opposite perspective, some methods divide the face into regions and try to restrict the match to uncorrupted parts of the face. For example, the approach in [19] relies on the accurate identification of the nose tip in order to extract multiple overlapping regions around the nose. These regions are matched using the ICP algorithm and the respective scores are combined together in order to evaluate face similarity. This idea is extended in [26] by using a set of 38 regions that densely cover the face, and selecting the best-performing subset of 28 regions to perform matching using the ICP algorithm. A recognition experiment accounting for partial match is reported that uses the left and right parts of the FRGC v2.0 probes. However, only experiments where some of the extracted regions have been removed are reported, rather than the more general case in which also parts of the regions are missing. And this latter effect is expected to substantially affect the ICP matching. In [2], a part-based 3D face recognition method is proposed which operates in the presence of both expression variations and occlusions. The approach is based on the use of Average Region Models (ARMs) for registration: The facial area is manually divided into several meaningful components such as eye, mouth, cheek and chin regions, and registration of faces is carried out by separate dense alignment of the regions with respect to the corresponding ARMs. The dissimilarities between gallery and probe faces obtained for individual regions are then combined to determine the final dissimilarity score. Under variations, like those caused by occlusions, the method can determine noisy regions and discard them. The performance of this approach is tested on the *Bosphorus* 3D face database [55] that includes facial expressions, pose differences and occlusions. However, a strong limitation of this solution is the use of manually annotated landmarks that are required for both face alignment and region segmentation. Instead of extended regions, a collection of radial curves originating from the nose tip is used in [25] to describe the facial surface. Face comparison is obtained by elastic matching of the curves. A quality control permits the exclusion of corrupted radial curves from the match, thus enabling the recognition also in the case of missing data. Results of partial matching are given for the 61 left and 61 right side scans of the *Gavab* database [43].

Point-Based Methods

The methods above use regions to perform face recognition. But regions are difficult to detect in that just some facial landmarks can be accurately identified when the pose significantly deviates from the frontal one. In addition, since parts of the regions can be missing or occluded, the extraction of effective region descriptors is hindered, so that regions comparison is often performed using rigid (ICP) or elastic registration (*deformable models*). Methods that use keypoints of the face promise to solve some of these limitations. Rather than relying on the detection of specific regions of the face – that can be error prone in the presence of occlusions and missing parts – they assume that detection of keypoints on the face surface and description of these keypoints yield robust yet accurate representation of facial traits, also in the presence of occlusions and missing parts, provided that the number of keypoints is sufficiently high. In this perspective, the use of keypoints instead of facial landmarks is advantageous. In fact, just few facial landmarks can be accurately detected in an automatic way – from three to ten are at most reported [29] – and detection of a larger number of landmarks is difficult even through partial manual assistance. In the case of partial face scans, up to half of these points are typically not detectable, so that description of such points and of their relationships is of limited effectiveness for face recognition. Differently, a much larger number of keypoints are typically detected – from tens to hundreds of keypoints can be easily derived – and their distribution is rather sparse, not being constrained to specific locations of the face. This makes keypoints more robust than landmarks to missing parts and also permits the extraction of a large number of local descriptors of the face.

A first result in the direction of using keypoints has been reported in [42], where a 3D keypoints detector and descriptor inspired by SIFT [38] has been designed. This detector/descriptor has been used to perform 3D face recognition through a hybrid 2D+3D approach that also uses the SIFT detector/descriptor to index 2D texture face images. However, results do not account for scans with pose variations and missing parts. Use of keypoints for partial face matching has been recently reported in [30, 31]. In this approach, Multi-Scale Local Binary Patterns (MS-LBP) and Shape Index (SI) are applied to face depth images, and the scalar values obtained at each pixel are used to create an MS-LBP map and an SI map. On both these maps, the SIFT detector and descriptor are used to represent local variations of the features extracted from the face. Finally, the matching scheme accounts for local and global face features by combining local matches between SIFT features, with global constraints originated by facial components. Partial face matching results are presented for the FRGC v2.0 scans where parts of the face are masked to simulate missing parts. However, as pointed out by the authors, the approach can deal automatically just with nearly frontal face data as those included in the FRGC v2.0 dataset. In the case of missing parts of the face due to large pose variations the approach is likely to fail. Methods in [37] and [21] use keypoints detection for the purpose of partial face matching, resulting the best performing approaches in the track on *3D Face Models Retrieval* of the SHREC'11 competition [59]. In particular, in [37] an extension of SIFT and index map based SIFT matching [31] is proposed. First, feature points are detected on each 3D face scan using *mesh*SIFT [40]; then, the quasi-daisy local shape descriptor [57] of each feature point is obtained using multiple order histograms of differential quantities extracted from the surface; Finally, these local descriptors are matched by computing their orientation angles (similarly to the SIFT-matching model). The number of matched points is used as similarity between two face scans. In [21], first a PCA based shape model is learned by registering a set of training scans to a reference template model (using 12 manually annotated landmarks) and subsequently warping the template on the training scans using a non-rigid registration based on variational implicit functions. The first 37 principal components are used for the analysis of the dense points correspondence of aligned scans. The learned model is then fitted to probe and gallery scans to generate model-based descriptions used to evaluate scans similarity. In this approach, *mesh*SIFT is used to detect keypoints whose correspondences in different scans permit to initialize the pose of probe and gallery scans with respect to the model (anyway, a manual initialization is required for about 2.5% of the scans). After pose initialization, the model is fitted following a Bayesian strategy with outliers detection and estimation. The result is an EM alike optimization, where the model updates are alternated with outlier updates, iteratively.

4.3 Very Large Pose Variations

Few studies investigated 3D facial recognition in the case large parts of the face are missing as a consequence of face acquisition with pose variations (i.e., yaw rotations from 45 up to 90 degrees) [25, 30, 46]. One recent approach that tackles this problem is that proposed in [12]. The approach starts from the observation that describing the face with local geometric information extracted at the neighbors of interest points easily permits partial face matching in that no particular assumption about the number or locations of the keypoints is necessary to perform sparse keypoints matching. However, in doing so, the size of the support used to compute the local descriptor at interest point locations becomes crucial: Small supports reduce the effectiveness of the descriptor and large supports are more sensible to missing parts that can alter the support itself. In addition, discriminant facial features are not only related to local characteristics of the face surface in the proximity of a set of keypoints, but also to mutual spatial relationships among the position of the keypoints on the face. Based on these premises, the approach relies on the detection of keypoints on the 3D face surface and the description of the surface in correspondence to these keypoints as well as along *facial curves* connecting pairs of keypoints. In contrast to solutions where keypoints correspond to meaningful face landmarks, such as the *eyebrows*, *eyes*, *nose*, *cheek* and *mouth* [29], this solution does not exploit any particular assumption about the position of the keypoints on the face surface. Rather, it is expected the position of keypoints to be influenced by the specific morphological traits of the face of each subject. In particular, the assumption of *within subject keypoints repeatability* is exploited: The position of the most stable keypoints – detected at the coarsest scales – do not change substantially across facial scans of the same subject. To further reduce the effect of surface noise and enhance the robustness of the position of keypoints, a spatial clustering approach is adopted so as to replace aggregated keypoints with their cluster centers. According to this, the combination of SIFT detection and spatial clustering is used to identify relevant and stable keypoints on the depth image of the face. Furthermore, facial curves are used to model the depth of face scans along the surface lines connecting pairs of keypoints. In doing so, distinguishing traits of a face scan are captured by the SIFT descriptors of detected keypoints, by the spatial arrangement of keypoints and by the set of facial curves identified by each pair of keypoints. Facial curves of gallery scans are also associated with a measure of *saliency* so as to distinguish those curves that model characterizing traits of some subjects from those curves that are frequently observed in the faces of different subjects. In the comparison of two faces, SIFT descriptors are matched to measure the similarity between pairs of keypoints identified on two depth images. Spatial constraints are imposed to avoid outliers matches. Then, the distance between the two faces is derived by composing the individual distances between facial curves (weighted by their saliency) that connect pairs of matching keypoints. In so doing, it is relevant to note that keypoints extraction and clustering are performed on depth images of the face. The derivation of these images requires pose normalization of 3D face scans to a common frontal position. This pre-processing still uses a few landmarks for raw registration and as initialization for the iterative rigid alignment procedure [13]. Differently, all the other processing steps do not rely on landmarks. The solution could be made totally independent of landmarks by the adoption of a different face alignment procedure. Recognition experiments from partial and full facial scans have been performed on the combined UND/FRGC v2.0 datasets and on the Gavab database. Experiments show that this approach can achieve state of the art results for face recognition from partial scans being also robust to facial expressions.

5 Discussion

In this Chapter, an overview of the ongoing research on 3D face analysis has been given. Among the many aspects involved in this topic, we first focussed on some issues that are common to any 3D face analysis solution, that is, face *acquisition* and *preprocessing* (see Sect. 2). For acquisition, laser and structured light scanners are now available that are capable to acquire together 3D static data and texture images at high-resolution. On this matter, the main challenges and expected outcomes are for high-resolution dynamic scanners capable to acquire sequences of highly detailed 3D scans at high frame rate and also with a sufficiently broad operative range. Techniques for 3D data preprocessing are becoming quite consolidated, including operations like noise removal, holes filling and surface smoothing. Detection of facial landmarks, face cropping and pose normalization are also preprocessing operations that are still required by many face analysis solutions, though some methods start to perform fairly good also renouncing to these operations. The advancements in both these two steps, acquisition and preprocessing, opened the way to the development of many of the existing 3D face analysis techniques.

The 3D face analysis research has been also pushed by the availability of large and challenging 3D *face datasets* with large variations in the subjects (number, gender, age, ethnicity), acquisition conditions (non-frontal pose and non-neutral expression) and in the characteristics of the acquired scans in terms of clutter and occlusions (scarves, eyeglasses, cap). Some of these datasets have also scans classified according to their facial expression, thus enabling the development and testing of methods for 3D facial expression classification. Moreover, manual annotations of facial landmarks are provided in some cases. The use of standard performance indicators on these common datasets has also facilitated the comparative evaluation of the proposed approaches. Since on many of the existing 3D face datasets several face recognition solutions achieved very high accuracy, it is a common feeling that the collections of more challenging datasets, that should include real world acquisitions rather than using laboratory settings, could further stimulate the development of more sophisticated solutions for 3D face recognition (see Sect. 3 for details on this topic).

Among the possible applications that use 3D face scans, in the last part of the Chapter we focussed on the 3D face recognition topic (see Sect. 4). In the last decade, many approaches have been developed addressing face recognition from static 3D face scans. The approaches that were first proposed tried to solve the basic problem of matching frontal neutral scans. Then, solutions accounted also for the facial expressions, thus resulting in methods capable to solve the recognition problem also in the case of moderate or large expression changes. However, these efforts were mainly devoted to cooperative scenarios where the subjects are aware and collaborate to the acquisition thus contributing to satisfy acquisition constraints. More recent solutions posed an increasing interest on methods capable to operate also in semi-cooperative or non-cooperative scenarios, where missing parts due to pose variations and occlusions can affect the acquired scans. Current and future research directions aim to solve the face recognition problem with the support of 3D data also in the case of completely non-cooperative scenarios, including outdoor environments, where dynamic acquisition of temporal sequences of 3D scans is also concerned. Promising solutions in this contexts are those trying to exploit and combine both 2D and 3D data.

References

1. 3DMD: `http://www.3dmd.com`
2. Alyüz, N., Gökberk, B., Akarun, L.: 3D face recognition system for expression and occlusion invariance. In: IEEE 2nd International Conferance on Biometrics: Theory, Applications, and Systems, Washington, DC, USA, pp. 1–7 (2008)
3. Amberg, B., Knothe, R., Vetter, T.: SHREC'08 entry: Shape based face recognition with a morphable model. In: IEEE International Conference on Shape Modeling and Appli-

cations, Stoney Brook, NY, pp. 253–254 (2008)

4. Bagdanov, A.D., Del Bimbo, A., Masi, I.: The Florence 2D/3D hybrid face dataset. In: Joint ACM Workshop on Human Gesture and Behavior Understanding (J-HGBU 2011), Arizona, USA, pp. 79–80 (2011)
5. Berretti, S., Ben Amor, B., Daoudi, M., del Bimbo, A.: 3D facial expression recognition using SIFT descriptors of automatically detected keypoints. The Visual Computer 27(11), 1021–1036 (2011)
6. Berretti, S., Del Bimbo, A.: Modeling spatial relationships between 3D objects. In: 18th International Conference on Pattern Recognition (ICPR 2006), Honk-Kong, China, vol. 1, pp. 119–122 (2006)
7. Berretti, S., Del Bimbo, A., Pala, P.: Description and retrieval of 3D face models using iso-geodesic stripes. In: ACM International Workshop on Multimedia Information Retrieval, Santa Barbara, CA, pp. 13–22 (2006)
8. Berretti, S., Del Bimbo, A., Pala, P.: 3D face recognition using iso-geodesic stripes. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(12), 2162–2177 (2010)
9. Berretti, S., Del Bimbo, A., Pala, P.: Facial curves between keypoints for recognition of 3D faces with missing parts. In: IEEE CVPR Workshop on Multi Modal Biometrics, Colorado Springs, Colorado, pp. 49–54 (2011)
10. Berretti, S., del Bimbo, A., Pala, P.: Real-time expression recognition from dynamic sequences of 3D facial scans. In: Proc. 5th Eurographics/ACM SIGGRAPH Workshop on 3D Object Retrieval (3DOR 2012), Cagliari, Italy, pp. 85–92 (2012)
11. Berretti, S., Del Bimbo, A., Pala, P.: Superfaces: A super-resolution model for 3D faces. In: Fusiello, A., Murino, V., Cucchiara, R. (eds.) ECCV 2012 Ws/Demos, Part I. LNCS, vol. 7583, pp. 73–82. Springer, Heidelberg (2012)
12. Berretti, S., Del Bimbo, A., Pala, P.: Sparse matching of salient facial curves for recognition of 3d faces with missing parts. IEEE Transactions on Information Forensics and Security (to appear, 2013)
13. Besl, P.J., Mc Kay, N.D.: A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), 239–256 (1992)
14. Beumier, C., Acheroy, M.: Face verification from 3D and grey level clues. Pattern Recognition Letters 22(12), 1321–1329 (2001)
15. Bowyer, K.W., Chang, K.I., Flynn, P.J.: A survey of approaches to three dimensional face recognition. In: International Conference on Pattern Recognition, Cambridge, United Kingdom, pp. 358–361 (2004)
16. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Three dimensional face recognition. International Journal of Computer Vision 64(1), 5–30 (2005)
17. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Robust expression-invariant face recognition from partially missing data. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 396–408. Springer, Heidelberg (2006)
18. Chang, K.I., Bowyer, K.W., Flynn, P.J.: An evaluation of multimodal 2d+3D face biometrics. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(4), 619–624 (2005)
19. Chang, K.I., Bowyer, K.W., Flynn, P.J.: Multiple nose region matching for 3D face recognition under varying facial expression. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(6), 1695–1700 (2006)
20. Chen, Y., Medioni, G.: Object modelling by registration of multiple range images. Image and Vision Computing 10(3), 145–155 (1992)
21. Claes, P., Smeets, D., Hermans, J., Vandermeulen, D., Suetens, P.: SHREC'11 track: Robust fitting of statistical model. In: Eurographics Workshop on 3D Object Retrieval, Llandudno, UK, pp. 89–95 (2011)

22. Colombo, A., Cusano, C., Schettini, R.: Gappy PCA classification for occlusion tolerant 3D face detection. Journal of Mathematical Imaging and Vision 35(3), 193–207 (2009)
23. Cook, J., Chandran, V., Fookes, C.: 3D face recognition using log-gabor templates. In: British Machine Vision Conference, Edinburgh, United Kingdom, vol. 2, pp. 769–778 (2006)
24. Daoudi, M., ter Haar, F., Veltkamp, R.: SHREC contest session on retrieval of 3D face scans. In: Shape Modeling International, Stoney Brook, NY (2008)
25. Drira, H., Ben Amor, B., Daoudi, M., Srivastava, A.: Pose and expression-invariant 3D face recognition using elastic radial curves. In: British Machine Vision Conference, Aberystwyth, UK, pp. 1–11 (2010)
26. Faltemier, T.C., Bowyer, K.W., Flynn, P.J.: A region ensemble for 3D face recognition. IEEE Transactions on Information Forensics and Security 3(1), 62–73 (2008)
27. Faltemier, T.C., Bowyer, K.W., Flynn, P.J.: Using multi-instance enrollment to improve performance of 3D face recognition. Computer Vision and Image Understanding 112(2), 114–125 (2008)
28. Farkas, L.G., Munro, I.R.: Anthropometric Facial Proportions in Medicine. Thomas Books, Springfield (1987)
29. Gupta, S., Markey, M.K., Bovik, A.C.: Anthropometric 3D face recognition. International Journal of Computer Vision 90(3), 331–349 (2010)
30. Huang, D., Ardabilian, M., Wang, Y., Chen, L.: 3-D face recognition using eLBP-based facial facial description and local feature hybrid matching. IEEE Transactions on Information Forensics and Security 7(5), 1551–1564 (2012)
31. Huang, D., Zhang, G., Ardabilian, M., Wang, Y., Chen, L.: 3D Face Recognition using Distinctiveness Enhanced Facial Representations and Local Feature Hybrid Matching. In: IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), Washington D.C., USA, pp. 1–7 (2010)
32. Husken, M., Brauckmann, M., Gehlen, S., Malsburg, C.: Strategies and benefits of fusion of 2d and 3D face recognition. In: IEEE Workshop Face Recognition Grand Challenge, San Diego, CA (2005)
33. Kakadiaris, I.A., Passalis, G., Toderici, G., Murtuza, N., Lu, Y., Karampatziakis, N., Theoharis, T.: Three-dimensional face recognition in the presence of facial expressions: An annotated deformable approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(4), 640–649 (2007)
34. Kinect: http://www.xbox.com
35. Konica Minolta: http://sensing.konicaminolta.us/products/vivid-910-3d-laser-scanner/
36. Lee, J.C., Milios, E.: Matching range images of human faces. In: International Conference on Computer Vision, Osaka, Japan, pp. 722–726 (1990)
37. Li, H., Chen, L.: SHREC'11 track: Salient points. In: Eurographics Workshop on 3D Object Retrieval, Llandudno, UK, pp. 89–95 (2011)
38. Lowe, D.: Distinctive image features from scale-invariant key points. International Journal of Computer Vision 60(2), 91–110 (2004)
39. Lu, X., Jain, A.K.: Deformation modeling for robust 3D face matching. In: Conference on Computer Vision and Pattern Recognition, New York, NY, pp. 1377–1383 (2006)
40. Maes, C., Fabry, T., Keustermans, J., Smeets, D., Suetens, P., Vandermeulen, D.: Feature detection on 3D face surfaces for pose normalisation and recognition. In: IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), Washington D.C., USA, pp. 1–6 (2010)
41. Mian, A.S., Bennamoun, M., Owens, R.: An efficient multimodal 2D-3D hybrid approach to automatic face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(11), 1927–1943 (2007)

42. Mian, A.S., Bennamoun, M., Owens, R.: Keypoint detection and local feature matching for textured 3D face recognition. International Journal of Computer Vision 79(1), 1–12 (2008)
43. Moreno, A.B., Sánchez, Á.: Gavabdb: A 3D face database. In: Workshop on Biometrics on the Internet, Vigo, Spain, pp. 75–80 (2004)
44. Pan, G., Han, S., Wu, Z., Wang, Y.: 3D face recognition using mapped depth images. In: Conference on Computer Vision and Pattern Recognition, San Diego, CA, vol. 3, pp. 175–181 (2005)
45. Passalis, G., Kakadiaris, I.A., Theoharis, T., Toderici, G., Murtuza, N.: Evaluation of 3D face recognition in the presence of facial expressions: an annotated deformable model approach. In: IEEE Workshop on Face Recognition Grand Challenge Experiments, San Diego, CA, vol. 3, pp. 171–179 (2005)
46. Passalis, G., Perakis, P., Theoharis, T., Kakadiaris, I.A.: Using facial symmetry to handle pose variations in real-world 3D face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(10), 1938–1951 (2011)
47. Perakis, P., Passalis, G., Theoharis, T., Toderici, G., Kakadiaris, I.A.: Partial matching of interpose 3D facial data for face recognition. In: International Conference on Biometrics: Theory, Applications, and Systems, Washington, DC, pp. 1–8 (2009)
48. Phillips, P.J., Flynn, P.J., Scruggs, T., Bowyer, K.W., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the face recognition grand challenge. In: IEEE Workshop on Face Recognition Grand Challenge Experiments, San Diego, CA, pp. 947–954 (2005)
49. Phillips, P.J., Flynn, P.J., Scruggs, T., Bowyer, K.W., Worek, W.: Preliminary face recognition grand challenge results. In: International Conference on Automatic Face and Gesture Recognition, Southampton, UK, pp. 15–24 (2006)
50. Phillips, P.J., Grother, P., Micheals, R.J., Blackburn, D., Tabassi, E., Bone, M.: FRVT 2002: Evaluation report. Tech. rep., National Institute of Standards and Technology, NIST (2003)
51. Phillips, P.J., Scruggs, W.T., O'Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. Tech. rep., National Institute of Standards and Technology (NIST), Gaithersburg, MD (2007)
52. Queirolo, C.C., Silva, L., Bellon, O.R., Segundo, M.P.: 3D face recognition using simulated annealing and the surface interpenetration measure. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(2), 206–219 (2010)
53. Samir, C., Srivastava, A., Daoudi, M.: 3D face recognition using shapes of facial curves. In: International Conference on Acoustics, Speech, and Signal Processing, Toulouse, France, vol. V, pp. 933–936 (2006)
54. Sandbach, G., Zafeiriou, S., Pantic, M., Rueckert, D.: Recognition of 3D facial expression dynamics. Image and Vision Computing (2012) (in press)
55. Savran, A., Alyüz, N., Dibeklio**u**glu, H., **c**Celiktutan, O., Gökberk, B., Sankur, B., Akarun, L.: Bosphorus database for 3D face analysis. In: Schouten, B., Juul, N.C., Drygajlo, A., Tistarelli, M. (eds.) BIOID 2008. LNCS, vol. 5372, pp. 47–56. Springer, Heidelberg (2008)
56. ter Haar, F., Veltkamp, R.: SHREC'08 entry: 3D face recognition using facial contour curves. In: IEEE International Conference on Shape Modeling and Applications, Stoney Brook, NY, pp. 259–260 (2008)
57. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: International Conference on Computer Vision and Pattern Recognition, Anchorage, AK, pp. 1–8 (2008)

58. University of Notre Dame Biometrics Datasets (2008), `http://www3.nd.edu/~cvrl/CVRL/Data_Sets.html`
59. Veltkamp, R., van Jole, S., Drira, H., Ben Amor, B., Daoudi, M., Li, H., Chen, L., Claes, P., Smeets, D., Hermans, J., Vandermeulen, D., Suetens, P.: SHREC'11 track: 3D face models retrieval. In: Eurographics Workshop on 3D Object Retrieval, Llandudno, UK, pp. 89–95 (2011)
60. Wang, Y., Chiang, M.C., Thompson, P.M.: Mutual information-based 3D surface matching with applications to face recognition and brain mapping. In: International Conference on Computer Vision, Beijing, China, pp. 527–534 (2005)
61. Wang, Y., Liu, J., Tang, X.: Robust 3D face recognition by local shape difference boosting. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(12), 1858–1870 (2010)
62. Wang, Y., Tang, X., Liu, J., Pan, G., Xiao, R.: 3D face recognition by local shape difference boosting. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 603–616. Springer, Heidelberg (2008)
63. Xu, D., Hu, P., Cao, W., Li, H.: SHREC'08 entry: 3D face recognition using moment invariants. In: IEEE International Conference on Shape Modeling and Applications, Stoney Brook, NY, pp. 261–262 (2008)
64. Yan, P., Bowyer, K.W.: A fast algorithm for icp-based 3D shape biometrics. Computer Vision and Image Understanding 107(3), 195–202 (2007)
65. Yin, L., Wei, X., Sun, Y., Wang, J., Rosato, M.: A 3d facial expression database for facial behavior research. In: Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition, Southampton, UK, pp. 211–216 (2006)
66. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. International Journal of Computer Vision 13(2), 119–152 (1994)
67. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: A literature survey. ACM Computing Survey 35(4), 399–458 (2003)

Socially-Driven Computer Vision for Group Behavior Analysis

Marco Cristani and Vittorio Murino

Abstract. The analysis of human activities is one of the most intriguing and important open issues in the video analytics field. Since few years ago, it has been handled following primarily Computer Vision and Pattern Recognition methodologies, where an activity corresponded usually to a temporal sequence of explicit actions (run, stop, sit, walk, etc.). More recently, video analytics has been faced considering a new perspective, that brings in notions and principles from the social, affective, and psychological literature, and that is called Social Signal Processing (SSP). SSP employs primarily nonverbal cues, most of them are outside of conscious awareness, like face expressions and gazing, body posture and gestures, vocal characteristics, relative distances in the space and the like. This paper will discuss recent advancements in video analytics, most of them related to the modelling of group activities. By adopting SSP principles, an age-old problem -what is a group of people?- is effectively faced, and the characterization of human activities in different respects is improved.

Introduction

Detecting human interactions represents one of the most intriguing frontiers of the automated surveillance since more than a decade. Recently, sociologic and psychological findings have been considered into video surveillance algorithms, especially thanks to the advent of Social Signal Processing work, a recent multi-disciplinary area where computer vision and social sciences converge. This chapter follows this

Marco Cristani
Università degli Studi di Verona and Pattern Analysis & Computer Vision, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova
e-mail: marco.cristani@iit.it

Vittorio Murino
Pattern Analysis & Computer Vision, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova
e-mail: vittorio.murino@iit.it

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_10, © Springer-Verlag Berlin Heidelberg 2014

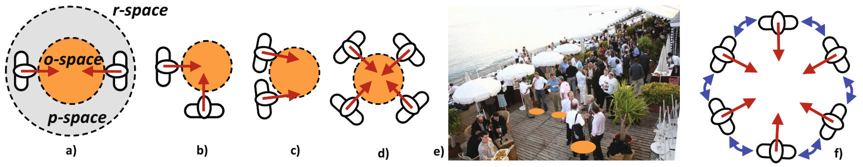


Fig. 1 F-formations: a-d) The component spaces of an F-formation: vis-a-vis, L, side-by-side, and circular F-formations, respectively. O-spaces are drawn in orange. e) Cocktail-party scene where some o-spaces are superimposed in orange.

direction and proposes a detailed overview on our recent activity on the analysis of group activities. In particular, we will present three scenarios where a group of interacting people is first detected, using positional and orientation features [15]; subsequently, the group is characterized by inferring the social relations between the participants exploiting proxemics cues [18]; finally, voice activity is detected by employing solely visual cues [16].

1 Analysis of Social Interactions Using F-formations

The first contribution is devoted to detect social interactions using statistical analysis of spatial-orientation arrangements that have a sociological relevance ([15]). As social interactions we intend the acts, actions, or practices of two or more people mutually oriented towards each other; more in general, any dynamic sequence of social actions between individuals (or groups) that modify their actions and reactions by their interaction partner(s). We analyze quasi-stationary people in an unconstrained scenario identifying those subjects engaged in a face-to-face interaction, i.e., a scene monitored by a single camera where a variable amount of people (10-20) is present. We import into the analysis the sociological notion of F-formation as defined by Adam Kendon in the late 70s ([39]).

Simply speaking, F-formations are spatial patterns maintained during social interactions by two or more people. Quoting Kendon, "*an F-formation arises whenever two or more people sustain a spatial and orientational relationship in which the space between them is one to which they have equal, direct, and exclusive access*.". In practice, an F-formation is the proper organization of three social spaces: o-space, p-space and r-space (see Fig. 1a-d).

The o-space is a convex empty space surrounded by the people involved in a social interaction, where every participant looks inward into it, and no external people is allowed in this region. This is the most important part of an F-formation. The p-space is a narrow stripe that surrounds the o-space, and that contains the bodies of the talking people, while the r-space is the area beyond the p-space.

There can be different F-formations as visible in Fig. 1a-d. In the case of two participants, typical F-formation arrangements are vis-a-vis, L-shape, and side-by-side. When there are more than three participants, a circular formation is typically formed [44].

Our approach aims at detecting the o-space, taking as input a calibrated scenario, in which the position of the people and their head's orientations have been estimated. In particular, we design an F-formation recognizer which is the main contribution of the work. This algorithm is based on a Hough-voting strategy, which lies between an implicit shape model [47], where weighted local features vote for a location in the image plane, and a mere generalized Hough procedure where the local features have not to be in a fixed number as in the implicit shape model. This approach provides the estimation of the o-spaces, so as of the identity of the people that form them, thus individuating people which are socially interacting. In such regard, our approach is the first to use F-formations detection in order to discover social interactions solely from visual cues.

Our approach has been tested on about a hundred of simulated scenarios, and two real annotated datasets, one of which is novel. In these last two cases tens of individuals were captured while they were enjoying coffee breaks, in indoor and outdoor environment, giving rise to heterogeneous real crowded scenarios. Our approach obtains convincing results, that are reported in a comparative way, quoting the unique (to the best of our knowledge) previous work dealing with the same topic.

The rest of the Section is organized as follows. In Sec.1.1, a review of the literature concerning the interaction modelling in surveillance settings is given. The proposed approach is detailed in Sec. 1.2, and the experiments are reported in Sec. 1.3. Finally, Sec. 1.6 concludes the paper with remarks and a discussion on the several possible future developments.

1.1 Group Interaction Discovery: State of the Art

A dated but interesting review on methods that consider human interactions is presented in [2], that focuses especially on motion cues. Pioneering studies on interactions focus on two-agent behaviors, employing statistical learning [56], a mix between syntactical and statistical pattern recognition paradigms [36], or Action-Reaction Learning [37]. Interactions among a larger number of people are usually modeled in meeting scenarios or smart rooms, exploiting a large number of heterogeneous sensors, thus solving many problems of occlusions and low image quality. In this case, many subtle social interactions can be observed and modeled, mostly by encoding turn-taking mechanisms. The interested reader may refer to [24] for a comprehensive review. Moving to unconstrained scenarios, as those typical of the videosurveillance field, the spectra of the activities modeled becomes narrower. In [34] a Semi Markov framework captures simple events (as running, approaching, etc.), where interaction is modeled by logic operators that assembly together simple events (performed by a single person) into multi-thread events. More recently, in [55, 12], group activities are encoded with three types of localized causalities, namely self-causality, pair-causality, and group-causality, which characterize the local interaction/reasoning relations within, between, and among motion trajectories of different humans, respectively. In [48], group interactions with a varying number of subjects are investigated, employing an asynchronous hidden Markov model as a hierarchical activity model. They distinguish symmetric (like i talks with j) and asymmetric dynamics activities (like i follows j). A discriminative approach is proposed in [45], in which two kinds of interactions are introduced. The first, group-person interaction, helps in individuating the action of a person by suggesting a context; the second, person-person interaction, identifies a group activity.

These approaches suffer from lack of generalization: they focus on a restricted set of actions, which are specific for a particular scenario. In this sense, a versatile generative model is presented in [72], where interacting events in crowded scene are modelled in an unsupervised way, and interactions are modeled as co-occurrences of atomic events. No tracking is performed due to the high people density, and local motions are considered as low-level features instead.

Approaches where sociological aspects are taken into account are [59, 65, 58, 45, 62, 6]. The keystone model that explains and simulates the human dynamics in crowd as a gas-kinetic phenomenon is the social force model (SFM) [32]. Here, interacting means being close each other during a walk or a run, and is explained as a balance between repulsive and attractive terms. The social force model has been modified in [59], where SFM is embedded as model for the dynamics in a tracking framework. Independently, a variational learning strategy is proposed in [65], where a dynamic model is trained for predicting the position of moving subjects, employing the SFM. In [58], a versatile synergistic framework for the analysis of multi-person interactions and activities in heterogeneous situations is presented. An adaptive context switching mechanism is designed to mediate between two stages, one where the body of an individual can be segmented into parts, and the other facing the case where people are assumed as rigid bodies. The concept of spatio-temporal personal space is also introduced to explain the grouping behavior of people. They extend the notion of *personal space* [3] to that of *spatio-temporal personal space*. Personal space is the region surrounding each person, that is considered personal domain or territory. Spatio-temporal personal space takes into account the motion of each person, modifying the geometry of the personal space into a sort of cone. This multi-person interaction approach share some similarities with our proposal, however, the sequences presented in the paper show very few people (max 3), and simpler situations. A quite novel perspective for detecting interactions in video surveillance scenarios come from the estimation of the human gaze (i.e., the head direction) in low resolution images [61]: in [6] the head direction serves to infer a 3D visual frustum as approximation of the focus of attention (FOA) of a person. Given the FOA and proximity information, interactions are estimated: the idea is that close-by people whose view frustum is intersecting are in some way interacting. In the experiments, we compared with this approach, abbreviated as IRPM. The same idea has been explored, independently, in [62]. Our approach improves this intuition, studying more in detail how people are usually located w.r.t each other during the interaction.

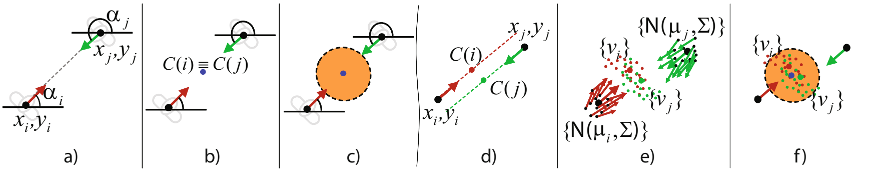


Fig. 2 Scheme exemplifying the proposed approach. (a-c) Two subjects exactly facing each other at a fixed distance vote for the same center of the circumference representing the o-space. (d) The 2 subjects do not face each other exactly in real cases. (e-f) Several positions and head orientations are drawn from Gaussian distributions associated to the subjects so as to deal with the uncertainty of real scenarios, making the proposed approach more robust.

1.2 A Socially-driven Method for Detecting Group Interactions

An F-formation can be specified by the related o-space and the oriented positions of the participants. Suppose we know the oriented positions of the subjects in the scene on the ground plane. Our algorithm jointly estimates the o-space(s) and the subjects involved in the related F-formation(s). The main idea is sketched through the toy example of Fig. 2a-c. Let us focus on $K = 2$ subjects, i and j, located at positions (x_i, y_i) and (x_j, y_j) with head orientation α_i and α_j, respectively. They are exactly facing each other, as depicted by the dashed line connecting their heads (Fig. 2a). Let us also suppose they are at a distance where social interaction can take place, i.e., $d = 1.5$ meters [1]. Given these (hard) constraints, each k-th subject votes for a candidate center $C(k)$ of the o-space, which has coordinates $x_{C(k)}, y_{C(k)}$:

$$C(k) = \left[x_{C(k)}, y_{C(k)}\right] = \left[x_k + r \cdot cos(\alpha_k), y_k + r \cdot sin(\alpha_k)\right], \quad k = 1, \ldots, K \qquad (1)$$

where the radius $r = d/2 = 0.75$. Each vote is accumulated in an *intensity accumulation space* $\mathscr{A}_I$, at entry $\tilde{x}_{C(k)}, \tilde{y}_{C(k)}$, where the tilde refers to the closest integer approximation (opportunely rounding the real value resulting from Eq. (1)) determined by the discretisation of the space $\mathscr{A}_I$. At the same time, the *ID labels* i and j are stored at the same entry of a *label accumulation space* $\mathscr{A}_L$, having the same size of $\mathscr{A}_I$. In the toy example of Fig. 2a, both people vote for a coincident location (Fig. 2b), which becomes the center of a *candidate* o-space (Fig. 2c).

To recover the subjects related to this candidate o-space, it is sufficient to access the labels in $\mathscr{A}_L$ associated to the votes in that location. We now know that the center of the candidate o-space has been voted by subjects i and j. At this point, the important condition of *"no-intrusion"* should be checked for the sociological consistence of the candidate o-space. The no-intrusion condition states: a candidate o-space for the subjects i and j does not have to contain other subjects different from i and j. If the no-intrusion condition is fulfilled the candidate o-space becomes a *valid* o-space.

[1] We will discuss this assumption later in the experiments.

One could object that the scenario depicted in Fig. 2a-c would be very rare. In fact, our experiments on real data suggest that people engaged in a discussion are rarely positioned on an exact circumference and facing its center. Moreover, computer vision methods are still not capable of estimating head orientation with high precision, and only a coarse quantization of this angle is typically considered in the current state of the art [10]. These two facts make the above deterministic, hard scheme ineffective. For example, no candidate o-space would be detected for the case in Fig. 2d where the subjects do not lie on the same diameter.

In order to deal with this problem, we inject uncertainty in the voting procedure, proposing an algorithm which is sketched in Fig. 2e-f. The proposed procedure is structured in three distinct stages and in the following we present an explanation for each step[2].

Sampling. We assume the positions and the (head) orientation of the different subjects as uncertain to some extent and modeled as random Gaussian variables, i.e.,

$$[x_k, y_k, \alpha_k]^T \sim \mathcal{N}(\mu_k, \Sigma_k) \tag{2}$$

where $\mu_k = [x_k, y_k, \alpha_k]^T$ and $\Sigma_k = \Sigma = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_\alpha^2)$. We transfer this uncertainty in the voting approach by drawing $N-1$ (being μ_k the N-th sample) i.i.d samples from every k-th distribution[3], as depicted in Fig. 2e. Each n-th sample of the k-th subject $s_{n,k} = [x_{n,k}, y_{n,k}, \alpha_{n,k}]^T$ has associated a weight $w_{n,k}$, which is the likelihood of being extracted from its generating distribution, i.e., $w_{n,k} = \mathcal{N}(s_{n,k}|\mu_k, \Sigma)$ and a label $l_{n,k} = k$, that links it to the related k-th individual.

Voting. Each sample votes for a candidate position in the same way of Eq. 1. The vote in the accumulation space $\mathscr{A}_I$ given by the n-th sample with weight $w_{n,k}$ adds $w_{n,k}$ in the accumulator, thus modeling the uncertainty associated to that sample. In this way, the accumulation space grows in number of votes, which are sparsely distributed. The accumulation of identity labels in $\mathscr{A}_L$ is done similarly for each sample as explained for the toy example in Fig. 2. Once the accumulation process is finished, the matrix $\mathscr{A}_I$ is revised with $\tilde{\mathscr{A}_I}$:

$$\tilde{\mathscr{A}_I}(x,y) = \mathrm{card}(x,y) \cdot \mathscr{A}_I(x,y) \quad \text{for each}\, x,y \in \mathscr{A}_I(x,y) \tag{3}$$

where $\mathrm{card}(x,y)$ counts the different subjects that voted in $\mathscr{A}_I(x,y)$. Such information is easily extracted from $\mathscr{A}_L(x,y)$. In this way, a high vote is given in those positions that have been voted with strong weights by many subjects. After that, the o-space may be found by looking for the maximum values of $\tilde{\mathscr{A}_I}$, and the associated subjects can be identified by checking $\mathscr{A}_L$.

O-space validation. The evaluation of the no-intrusion condition is performed by analyzing how strong is the presence in the o-space of an external subject. Following

[2] Additional material at `http://profs.sci.univr.it/~cristanm/publications.html`includes a pdf with a summary of the algorithm as a scheme.

[3] In this paper, we fix Σ and the number of samples for all the people observed. However, interesting policies can be adopted in dependence on the certainty we have in the k-th subject (for example due to the tracker providing the subject position, or to the classifier estimating the head orientation).

a probabilistic approach, we compute the maximum weight $w^*_{n,h}$ of a sample of an external subject h which falls in the candidate o-space. A high $w^*_{n,h}$ in an o-space of center (x_c, y_c) mirrors a high probability that h is invading that o-space. A threshold τ_{INTR} is used to detect the invading external subject. If this happens, the o-space is invalid, and the intensity accumulator is updated imposing $\tilde{\mathscr{A}_I}(x_c, y_c) = 0$, and the search for the maximum value on the updated $\mathscr{A}_I$ is repeated.

This algorithm extends naturally to F-formations composed by more than two subjects and to more F-formations in the same scene thanks to the characteristics of the Hough voting scheme. Actually, in a crowded situation, there could easily be more than one F-formation. Thus, we need to check all the possible o-spaces efficiently, and this is done in the following way. Consider the case of two subjects i and j with their o-space detected as described in the *O-space validation* stage. The accumulators $\mathscr{A}_I$ and $\mathscr{A}_L$ are then updated by pruning away the votes given by $\{w_{n,i}\}$ and $\{w_{n,j}\}$ in $\mathscr{A}_I$, respectively, and removing the labels i and j from $\mathscr{A}_L$. Then, $\tilde{\mathscr{A}_I}$ is re-computed. The max search process on $\tilde{\mathscr{A}_I}$ and the no-intrusion check are thus repeated, and this is iterated until no more o-spaces are found. This strategy has also the beneficial effect of providing the F-formations in decreasing order of likelihood, assuming the likelihood of an F-formation proportional to the accumulation of votes (which can be assimilated to probabilities) in the center of the related o-space stored in $\mathscr{A}_I$.

1.3 Experiments

Our algorithm has been tested on synthetic and real data. The former proves the effectiveness of our algorithm in detecting groups disregarding a-priori errors due to bad tracking or wrong head orientation estimations. The latter considers two different real scenarios, one indoor and one outdoor, where errors may occur.

As accuracy measures, we estimate that a group has been correctly estimated if at least $\lceil (2/3 \cdot |G|) \rceil$ of their components are found, where $|G|$ is the cardinality of group G. This rule has an exception that holds in the case $|G| = 2$. In that case, all the components must be detected. Given this, for each situation analyzed we estimate the *precision* and *recall* of finding groups, averaged over time.

In addition, to further promote the versatility of our framework, we build for each sequence a *relation matrix* P_2 that represents how many times two people stand in the same group for a certain period of time. Actually, during a party, people may change groups, standing alone for a while, re-joining a conversation, etc.. P_2 analyzes the strength of pairwise relations and, for example, is capable to indicate, given a person, who is the subject with which she/he is interacting most. This matrix has been employed in other social signalling techniques [22], and we can compare it with the analogous matrix built employing the ground-truth data. A measure of the similarity between the two matrices has been performed employing the *Mantel Test* [50], which is commonly used in cluster analysis to test the correlation between two distance matrices. It operates by evaluating correlations scores from repeated randomizations of the entries of the matrices. If randomizations frequently produce a correlation stronger or as strong as the original data, there is little evidence that the correlation between the two matrices differs from zero. In rough terms, it is a measure of similarity between matrices which actively takes into account their structure.

The proposed method is compared with the Inter-Relation Pattern Matrix method[4] (IRPM) proposed in [6], whose description is reported in Sec. 1.1.

The free parameters of the method are the radius r, the variances $\sigma_x^2, \sigma_y^2, \sigma_\alpha^2$, the number of samples per-person N, and the threshold τ_{INTR} of the no-intrusion condition. Choosing such values is very intuitive, and it can be driven by sociological and empirical considerations. As an example, the setting of the radius r is a matter of pure sociological aspects: Hall [29] defines 4 relational ranges of distances that witness the type of relation a subject has with the others, and are (expressed in meters): $[0, 0.45]$ for *intimate* relations, $(0.45, 1.2]$ for *casual/personal* relations, $(1.2, 3.5]$ for *social/consultive* relations, and > 3.5 for no-relation. Now, suppose that two people are involved in a vis-a-vis interaction. They may make a circular o-space whose diameter is $2r$. In all the other F-formations, the distance among two people is $< 2r$. Therefore, r represents half of the maximal distance two people may lie in the space and being judged as connected in an F-formation. If we set $r = 60cm$, we are interested in a upper bound that becomes the *casual/personal* range, because $2r = 120cm$

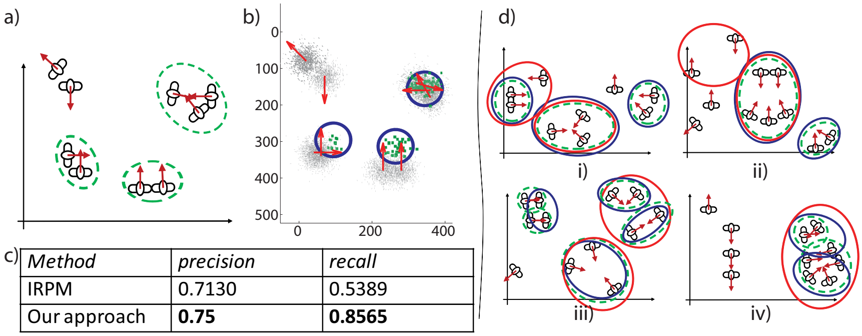
The parameters σ_x^2 and σ_y^2 allow to project the position of the people in different positions, covering a range of $3\sigma_{x(y)}$. In other words, these values allow to be flexible about the classes of relations taken into account by the r parameter. We fix $\sigma = \sigma_x^2 = \sigma_y^2 = 400cm$, considering thus a range of maximal distances for the F-formations of $2[r - 3\sigma, r + 3\sigma] = [0, 240]cm$. The value of σ_α^2 depends on the quantization of the head orientation. We employ 4 head orientations, so $\sigma_\alpha^2 = 0.005$ is a reasonable value. The parameter N can be instead chosen by considering computational aspects. In the current, non-optimized MATLAB version it takes averagely 15 second per frame using $N = 800$.

Finally, the last parameter τ_{INTR} checks the weights (i.e. likelihood probabilities) of the intruder samples. Therefore, its setting mirrors how tolerant we want to be in considering a sample as a genuine representative of an intruder, depending on its weight. We fix $\tau_{INTR} = 0.7$. Once the parameters are set, they are kept fixed for all the experiments.

1.4 Synthetic Data

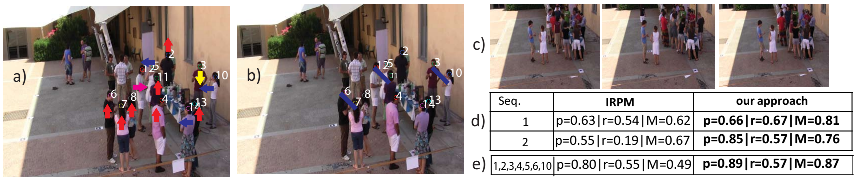
A psychologist provided 100 different *situations*, where some subjects take part in an F-formation and other do not (examples in Fig. 3d). The input of the tested algorithms is the actual position and head orientation of each subject. The data has been annotated to obtain ground truth of the F-formations. We apply our algorithm and IRPM to all the situations, averaging the precision and the recall scores of all the

[4] The code is available at
`http://www.lorisbazzani.info/code-datasets/irpm/`



Method	*precision*	*recall*
IRPM	0.7130	0.5389
Our approach	**0.75**	**0.8565**

Fig. 3 Experiments with synthetical data (see text). The figure is better viewed in color.



<table>
<tr><th>Seq.</th><th>IRPM</th><th>our approach</th></tr>
<tr><td>1</td><td>p=0.63|r=0.54|M=0.62</td><td><b>p=0.66|r=0.67|M=0.81</b></td></tr>
<tr><td>2</td><td>p=0.55|r=0.19|M=0.67</td><td><b>p=0.85|r=0.57|M=0.76</b></td></tr>
</table>

<table>
<tr><td>1,2,3,4,5,6,10</td><td>p=0.80|r=0.55|M=0.49</td><td><b>p=0.89|r=0.57|M=0.87</b></td></tr>
</table>

Fig. 4 Experiments with real data (see text). In the tables, p, r, M stand for (mean) precision, recall, and Mantel score, respectively.

situations. Fig. 3a shows an examplar situation from the synthetic dataset. Fig. 3b depicts how the sampling process propagates instances of a subject in gray, the votes of the intensity accumulator in green, and the resulting o-spaces in blue. A qualitative analysis has been reported in Fig. 3b. The ground truth is depicted in dotted green, whereas the results of our approach and IRPM are in blue and in red, respectively. Our approach is able to model interactions where IRPM fails. In case (iii), our approach fails in estimating the two vis-a-vis interaction, being them very close. In general, looking at the global results in Fig. 3c, one can note that our proposal gets higher rates for both precision and especially for the recall.

1.5 Real Scenarios

The outdoor situation is represented by a novel dataset, dubbed *CoffeBreak* and downloadable at `http://profs.sci.univr.it/~cristanm/datasets.html`. It represents a coffee-break scenario of a social event that lasted 4 days, captured by two cameras. The dataset is part of a social signaling project whose aim is to monitor how social relations evolve over time. Nowadays, only 2 sequences of a single day of a single camera have been annotated, each one covering a period of averagely 1 minute. A psychologist annotated the videos indicating the groups present in the scenes, for a total of 45 frames for *Seq1* (a frame in Fig. 4a-b) and 75 frames for *Seq2* (see Fig. 4c). The annotations have been done by analyzing each frame and a set of questionnaires that the subjects filled in. The dataset is still challenging from the tracking and head pose estimation point of view, due to multiple occlusions. This enables us to test our technique in a very noisy situation.

Since CoffeBreak is a crowded scenario, occlusions make extremely hard full human bodies detection. Thus, the subjects' heads are the only cues to perform tracking in a robust way. To extract the head locations of all the subjects in the scene we adopted a system based on class-specific Hough forests [23] trained on human heads in all possible orientations. This allowed us to reliably detect all the possible head candidates in the scene, independently from their orientation with respect to the ground plane. After performing head detection in all the frames, such detections needed to be filtered and linked in order to generate plausible ground plane trajectories of all the subjects. To this end, the ground plane homography and an estimation of the average height of the subjects were used to compute the ground plane location corresponding to each head detection. Consecutive detections corresponding to the same subject were linked by matching appearance descriptors. Finally, head orientation detection has been performed on 4 classes employing the covariance based approach of [69] (see Fig. 4a). Once the oriented positions of the head are given, we estimate the ground plane homography given a set of measurements obtained on site.

The mean precision, recall score and the Mantel correlation reported in Fig. 3d show that our approach outperforms IRPM. In Fig. 3a-b some qualitative results are depicted: in Fig. 3a we have the head detection results together with the orientation. In Fig. 3b, the blue segments indicate the groups found by our approach (the ground truth is (6,7),(11,12,5),(3,10)). IRPM did not find any groups in that frame.

The indoor data come from a publicly available dataset for group detection, called GDet 2010 and downloadable at `http://www.lorisbazzani.info/code-datasets/multi-camera-dataset/`. The dataset is made by 12 subsequences of about 2 minutes each, with the availability of the full camera calibration parameters. GDet 2010 videos consider a vending machines area where people take coffee and other drinks, and chat in the spare time. The videos have been acquired with two monocular cameras, located on opposite angles of a room close to the floor. People involved in the experiments were not aware of the aim of the trials and behaved naturally. The ground truth has been made by a psychologist like in the CoffeeBreak scenario. Afterwards, some of them were asked to fill in a form inquiring if they talked to someone in the room and to whom. The videos have been analyzed by a psychologist, that noted the social exchanges occurred and produced the ground truth of social interactions. In this case, people tracking has been performed using Hybrid Joint-Separable (HJS) filter proposed in [46], for its capability of dealing with occlusions by means of the estimation of the occlusion maps exploiting the camera calibration. Given the bounding boxes of the tracked people, the head is approximately located within a bounding box. Then, head pose estimation is performed like in the CoffeeBreak scenario.

A quantitative analysis of the results on a subset of sequences is reported in Fig. 4e. Even in this case, our approach outperforms IRPM. Note the values of the

Mantel tests: in general, our approach draws a social situation in terms of pairwise relations which is close to the ground truth.

1.6 Remarks

This section presents a sociologically principled method for the detection and analysis of human interactions exploiting F-formations. An F-Formation is a plausible ensemble of possible spatial and orientational organisation people assume during the course of an interaction. Our approach aims at automatically detecting the main social space identified by the sociological findings, the so called o-space, which is a space internal to the interacting people in which no other people are allowed to lie. The net result is a brand new robust interaction detection algorithm based on a well-established sociological theory able to deal with simple to moderately crowded scenes.

The approach has been tested on synthetic data and real scenarios proving its robustness and accuracy in the disparate situations addressed. This is appreciable per se (as compared to ground truth) and also ameliorates the current state of the art results of the IRPM-based method. These results are obtained dealing with complex scenario in which the people detection, the orientation of their heads, and tracking are difficult, likely producing inaccurate input data. Still, our algorithm performs quite well in detecting interactive groups thanks to the statistical voting process.

So far in the literature, this is the first approach that discovers social interactions based on the automatic detection of F-formations solely from visual cues. Many improvements can be certainly envisaged for the future work. From the algorithmic point of view, clear and obvious improvements may derive from the use of the temporal information provided by the tracking, so as from the adoption of more reliable and efficient people detection and head orientation classification methods. From the application perspective, additional features extracted from the detected F-formations may support the comprehension of the interactions, possibly predicting the likely outcome, which can be useful in evaluating situations of social interest.

2 Inferring Social Relations from Interpersonal Distances

The second contribution is about the characterization of the group interaction aimed at the recognition of the relations among the interlocutors by using proxemic cues [18]. Proxemics can be defined as the “[...] *the study of man’s transactions as he perceives and uses intimate, personal, social and public space in various settings* [...]”, quoting Hall [30, 31], the anthropologist who first introduced this term in 1966. In other words, proxemics investigates how people use and organize the space they share with others to communicate, typically outside conscious awareness, socially relevant information such as personality traits (e.g., dominant people tend to use more space than others in shared environments [49]), attitudes (e.g., people that discuss tend to seat in front of the other, whereas people that collaborate tend to seat side-by-side [63]), etc..

This section focuses on one of the most important aspects of proxemics, namely the relationship between physical and social distance. In particular, the section shows that interpersonal distance (measured automatically using computer vision techniques) provides physical evidence of the social distance between two individuals, i.e. of whether they are simply acquainted, friends, or involved in a romantic relationship. The proposed approach consists of two main stages: the first is the automatic measurement of interpersonal distances, the second is the automatic analysis of interpersonal distances in terms of proxemics and social relations (see Section 2.3 for details).

The choice of distance as a social relation cue relies on one of the most basic and fundamental findings of proxemics: People tend to unconsciously organize the space around them in concentric zones corresponding to different degrees of intimacy [30, 31]. The size of the zones changes with a number of factors (culture, gender, physical constraints, etc.), but the resulting effect remains the same: the more two people are intimate, the closer they get. Furthermore, intimacy appears to correlate with distance more than with other important proxemic cues like, e.g., mutual orientation [26]. Hence, it is reasonable to expect that the distance accounts for the social relation between two people.

One of the main contributions of the paper is that the experiments consider an ecological scenario (standing conversations) where more than two people are involved. This represents a problem because in this case distances are not only determined by the degree of intimacy, but also by the need of ensuring that every person can participate in the interaction. This leads to the emergence of stable spatial arrangements, called F-formations (see Section 2.1 for more details) [39], that impose a constraint on interpersonal distances and need to be detected automatically. Furthermore, not all distances can be used because, in some cases, they are no longer determined by the degree of intimacy, but rather by geometric constraints. The approach proposed in this work is to consider only the distances between people adjacent in the F-formation (see Section 2.4 for more details) [39].

The other important contribution is that, in contrast with other works in the literature, the radii of the concentric zones corresponding to different degrees of intimacy are not imposed a-priori, but rather learned from the data using an unsupervised approach. This makes the technique robust with respect to the factors affecting proxemic behavior, like culture, gender, etc., as well as environmental boundaries. In particular, the experiments show how the organization into zones changes when decreasing the space at disposition of the subjects and how the unsupervised approach is robust to such an effect.

Standing conversations are an ideal scenario not only because they offer excellent examples of proxemic behavior, but also because they allow one to work at the crossroad between surveillance technologies, often applied to monitor the behavior of people in public spaces, and domains like Social Signal Processing that focus on automatic understanding of social behavior. This is expected to lead, on the long-term, to socially intelligent surveillance and monitoring technologies [17].

The rest of this section is organized as follows. Section 2.1 introduces the main concepts of proxemics, and Section 2.2 provides a brief survey of the state-of-the-art in computational proxemics. Section 2.3 presents the approach, and Section 2.4 reports the experiments and results. Finally, Section 2.5 draws some conclusions.

2.1 Fundamentals of Proxemics

The wide spectrum of nonverbal behavioral cues displayed during social interactions (facial expressions, vocalizations, gestures, postures, etc.) is well known to convey information about social and affective aspects of human-human interaction (attitudes, personality, emotions, etc.) [60]. Proxemics has shown that the way people use, organize and share space during gatherings and encounters is a nonverbal cue and it conveys, like all other cues, social and affective meaning [42]. This section provides a short description of the main findings of the discipline, with particular attention to phenomena that can be observed in standing conversations, the scenario investigated in the experiments of this work.

From a social point of view, two aspects of proxemic behavior appear to be particularly important, namely interpersonal distances and spatial arrangement of interactants.

The rest of this section focuses on both aspects, including the most important factors that influence them.

2.1.1 Interpersonal Distances

Interpersonal distances have been the subject of the earliest investigations on proxemics and one of the main and seminal findings is that people tend to organize the space around them in terms of four concentric zones associated to different degrees of intimacy:

- *Intimate Zone*: distances for unmistakable involvement with another body (lover or close friend). This zone is typically forbidden to other non-intimate persons, except in those situations where intrusion cannot be avoided (e.g. in elevators).
- *Casual-Personal Zone*: distances established when interacting with familiar people, such as colleagues or friends. This zone is suitable for having personal conversations without feeling hassled. It also reflects mutual sympathy.
- *Socio-Consultive Zone*: distances for formal and impersonal relationships. In this zone, body contact is not possible anymore. It is typical for business conversations, consultation with professionals (lawyers, doctors, officers, etc.) or seller-customer interactions.
- *Public zone*: distances for non-personal interaction with others. It is a zone typical for teachers, speakers in front of a large audience, theater actors or interpersonal interactions in presence of some physical barrier.

In the case of Northern Americans, the four zones above correspond to the following ranges: less than 45 *cm* (intimate), between 45 and 120 *cm* (casual-personal), between 120 and 200 *cm* (socio-consultive), and beyond 200 *cm* (public). While the actual distances characterizing the zones depend on a large number of factors (e.g., culture, gender, physical constraints, etc.), the partition of the space into concentric areas seems to be common to all situations.

2.1.2 Spatial Arrangement: The F-Formations

The spatial arrangement during social interactions addresses two main needs: The first is to give all persons involved the possibility of participating, the second is to separate the group of interactants from other individuals (if any). The result are the *F-formations*, stable patterns that people tend to form during social interactions (including in particular standing conversations): "*an F-formation arises whenever two or more people sustain a spatial and orientational relationship in which the space between them is one to which they have equal, direct, and exclusive access*" [39].

In practice, an F-formation is the proper organization of three social spaces (see Figure 1): O-space, P-space and R-space. The O-space (the most important component of an F-formation) is a convex empty space surrounded by the people involved in a social interaction, every participant looks inward into it, and no external people are allowed in this region. The P-space is a narrow stripe that surrounds the O-space and that contains the bodies of the interactants, the R-space is the area beyond the P-space. There can be different F-formations:

- *Vis-à-vis*: An F-formation in which the absolute value of the angle between participants is approximately 180^o, and both participants share an O-space.
- *L-shape*: An F-formation in which the absolute value of the angle between participants is approximately 90^o, and both participants share an O-space.
- *Side-by-side*: An F-formation in which the absolute value of the angle between participants is approximately 0^o, and both participants share an O-space.
- *Circle*: An F-formation where people is organized in a circle, so that the configuration between adjacent participants can be considered as a hybrid between a L-shape and a Side-by-side F-formation.

The same contextual factors that influence the concentric zones described above, affect F-formations as well.

2.1.3 Context Effects on Proxemics

Proxemic behavior is affected by a large number of factors and culture seems to be one of the most important ones, especially when it comes to the size of the four concentric zones described above. In particular, cultures seem to distribute along a continuum ranging from "contact" (when the size of the areas is smaller) to "non-contact" (when the size of the areas is larger) [31]. Further evidence in this sense is proposed in [73], where people from "contact" cultures are shown to approach one another more than the others, and in [66], where the culture effect has been shown to depend on whether one considers shape of territory, size, central tendencies of encroachment, or encroachment variances (the observations were conducted on beaches). In the same vein, interpersonal distances seem to be affected by ethnicity: e.g., black Americans and Mexicans living in the States appear to have different

”contact” tendencies [31, 5]. The effect of culture seems to change when interaction participants have seats at disposition. In this case, people from supposedly “non-contact” cultures tend to seat closer than the others [33]. Furthermore, the seating arrangement seems not to depend on culture [51].

Seating is just one of the many environmental characteristics that can influence the requirements on interpersonal distance and personal space. The literature has investigated the effect of many other characteristics as well, including lighting [1], indoor/outdoor [13], crowding [27] and room size [75, 64, 14]. The work in [1] investigates the effect of lighting with stop-distance techniques: Experimenters get closer and closer to a subject that remains still and says “stop” when she starts feeling uncomfortable. Subjects in bright conditions (600 lx) allow the experimenters to come significantly closer than the subjects in dim conditions (1.5 lx). A similar effect has been observed for the size of the place where people interact: people allow others to come closer in larger rooms [75], when the ceiling is higher [64][14], and in outdoor spaces [13]. The effects of crowding have been studied as well [27]: Social density was increased in a constant size environment for a limited period of time and participants of larger groups reported greater degrees of discomfort and manifested other forms of stress.

2.2 Computational Proxemics: State-of-the-Art

To the best of our knowledge, only a few works have tried to apply proxemics in computing. One probable reason is that current works on analysis of human behavior have focused on scenarios where proxemics do not play a major role or have relied on laboratory settings that impose too many constraints for spontaneous proxemic behavior to emerge (e.g., small groups in smart meeting rooms) [25, 70].

Most of the computing works that can be said to deal with proxemics concern the dynamics of people moving through public spaces. These works typically model repulsive/attractive phenomena by adopting the Social Force Model (SFM) [32]. In particular, the work in [59, 65] improves the perfomance of a tracking approach by taking into account the distance between a subject being tracked and the other subjects appearing in a scene. An attempt to interpret the movement of people in social terms has been presented in [28], where nine subjects (asked to speak among them about specific themes) were left free to move in a $3m \times 3m$ area for 30 minutes. An analysis of mutual distances in terms of the zones described in Section 2.1 allowed to discriminate between people who did interact and people who did not. In a similar way, mutual distances have been used to infer personality traits of people left free to move in a room [76]. The results show that it is possible to predict Extraversion and Neuroticism ratings based on velocity and number of intimate/personal/social contacts (in the sense of Hall) between pairs of individuals looking at one other.

Another frequent application area is social robotics. Early approaches in the domain simply aimed at making robots to respect the personal space of users [54], but more recent works deal with the initiation, maintenance, and termination of social interactions by modulating reciprocal distances, showing that people use similar proxemic rules when interacting with robots and when interacting with other people [68]. In [9] a generative model has been developed for selecting a set of reactive behaviors that depend on the distance, speed, and sound of interactants. Distance cues are used by the Roboceptionist [53] for recognizing "Present", "Attending", "Engaged", and "Interacting" people at the entrance of the Robotics Institute at Carnegie Mellon University. In [57], a model for human-robot interaction in a hallway is proposed. The idea is to exploit proxemic cues for letting the robot to react properly at the passage of an individual in a narrow corridor. In [43], a user study focuses on the interaction between a human and a robot in a domestic environment. Interactions were analyzed exploiting the four zones and the F-formations introduced in Section 2.1. The researchers found the Personal zone to be the most commonly occupied one and the "vis-à-vis" F-formation to be the most frequent spatial arrangement.

2.3 Detecting a Flexible Set of Socially Meaningful Distances

The proposed approach includes two main stages: the first is the detection of F-formations, and the second is the inference of social relations from interpersonal distances.

2.3.1 Detection of F-Formations

The goal of this stage is to detect F-formations in videos portraying people involved in standing conversations. The first step is to track the people with a fish-eye camera pointing at interactants in a bird-eye view setting (see Figure 5 for an example). This corresponds to a realistic surveillance scenario and allows one to track people with satisfactory precision (tracking has been performed by exploiting a particle filter on each person [4], employing a standard background subtraction algorithm for highlighting the moving objects [67]. The results of our approach that have been obtained with this tracking strategy have been compared with those obtained via manual tracking, showing very similar results). The detection of the F-formations is performed over the output of the tracking step using the approach described in [15]. The output of the F-formations detection algorithm has been validated by hand and it did not produce any error.

F-formations lasting for less than 5 seconds (50 frames in our implementation) have not been taken into consideration in the experiments of this work. The reason is that the next stage of the processing requires the application of a clustering algorithm and 50 frames is a reasonable amount of data needed to avoid the so-called "curse of dimensionality" [20].

2.3.2 Inference of Social Relations

The output of the first stage is a list of pairs where each element includes two subjects that are adjacent in a detected F-formation. Furthermore, the first stage provides the $2D$ position of each subject on the surface of the room. Such data is accumulated during a time interval (called the "stable period" hereafter) that does not include creation, break or modifications of an F-formation (e.g., no people change their position in the P-region). This ensures that during the time interval under analysis all causes that might change the current F-formation are absent. Such causes can be novel people being involved, people leaving, a change in the environmental conditions like rain (people look for a repair), an intruder (e.g., a vehicle passing by and disrupting the F-formation), etc.. The satisfaction of the conditions above is automatically verified by checking that the relative distances between subjects in a F-formation do not change abruptly (i.e., the changes do not exceed a threshold learned automatically from the data).

During the stable period, the approach collects and pools together all pairwise distances between individuals (for a sketch, see Figure 1 (f)). Distances are collected between the centers of mass of the tracked blobs, where each blob corresponds to a separate person. These are shown to distribute according to different modes (see Section 2.4) that should correspond to the concentric zones described in Section 2.1. The modes have been separated via Gaussian clustering by employing the Expectation-Maximization (EM) [19] learning method. The EM employed here is a variation of the original formulation [21]; it is performed by means of a model selection strategy that is injected in the learning stage and that shows several properties that fit well with the situation at hand. First, it allows one to automatically select in an unsupervised way the right number of Gaussian components (in an Information Theory sense). This is a very important aspect, that permits to let the natural separation of the data emerge without human intervention. Second, it deals satisfactorily with the initialization issue, i.e., the Gaussian parameters fit the data realizing a nearly-global optimal fit, minimizing the probability of overfitting (i.e., a Gaussian component that fit only a few data). In addition, the Gaussian clustering takes into account in a principled way the noise due to possibly unprecise tracking, incorporating it as a variance of the measures.

2.4 Testing the Flexible Distances

This section presents experiments and results obtained in this work.

2.4.1 Experimental Setup

The goal of the experiments is to investigate spontaneous standing conversations in a public space, hence the tests have been performed in an outdoor area of size $3m \times 7m$ (see Fig. 5, row (i), column (a)). The area is empty (no physical constraints or obstacles) and two groups of subjects have been invited, in two separate sessions, to move and behave normally through it. The subjects were told that the experiments were aimed at testing a tracking approach and were unaware of the real motivations behind the experiments. During the sessions, the subjects were left alone and no researcher involved in this work was present.

The experiment took place on February 2011, on a sunny day. The area was monitored with a Unibrain Fire-i Digital Camera, on which fisheye optics was mounted.

The camera was located 7 meters above the floor, and it was held to an architectural element of the infrastructure. Therefore, the impact of the capture device onto the ecology of the environment was minimal. The acquisition frame rate was 10 frames per second. After the data acquisition, video data were rectified for correcting the spherical distortion. The two sessions were 15 minutes long for a total of around 20000 frames. One quarter of hour is a duration long enough to collect evidence of pre-existing social relations and short enough to avoid the emergence of new relations. The first session was recorded at 11 AM and the second at 2 PM.

Each session was split into three 5 minutes long segments corresponding to three different experimental conditions:

- Condition 1: the subjects are free to move through the entire area
- Condition 2: the movements of the subjects are restricted to an area of size $3m \times 3.5m$
- Condition 3: the movements of the subjects are restricted to an area of size $1.5m \times 2.0m$

The physical restrictions were represented by lines and marker on the floor. The goal was to measure the effect of the amount of available space on proxemic behavior.

2.4.2 Results of Session 1

The first session involved six subjects (see Fig. 5): two undergraduate students (*a* and *b*), an assistant professor (*c*), and three PhD students working in the same laboratory (*d*, *e*, *f*), two of them working on the same topic (*e* and *d*). The PhD students and the assistant professor were acquainted before the experiment. The undergraduate students are friends, but they never met before the other subjects. In Fig. 5 row (i) we show the results obtained in the longest stable period (subjects free to move in the entire area, see Section 2.3.2), that in this case lasted 108 frames. The image in column (a)-(b) is the last of the period[5]. In that interval, the group was split into three dyads. The histogram in Figure 5-row (i) shows the distribution of the interpersonal distances between members of the same dyad. The application of a clustering approach shows the existence of two modes centered on 48 and 64 *cm*, respectively. The tables in the figure report the fraction of time distances between each pair of adjacent individuals belong to a given mode for each condition, with the value in bold red indicating the highest (most frequent cluster membership) fraction. The two modes seem to account for two of the zones identified by Hall and, not surprisingly, the dyad involving the assistant professor is the only one where the distance belongs with higher probability to the second mode most of the times. This confirms that the higher social distance between the assistant professor and the PhD student results into a physical distance that is higher (on average) than the one between subjects *a* and *b* (who are friends and both undergraduate students), as well as the one between subjects *d* and *e* (who are both PhD students).

[5] The same applies for all the other pictures in the column (a)-(b), i.e., they are the last frames of the corresponding stable period.

In Condition 2 (3×3.5 meters), the longest stable interval (122 frames) corresponds to a circle F-formation, including all subjects (see Fig. 5-row (ii), pictures at left). The clustering of the interpersonal distances of adjacent subjects reveals this time a three-mode distribution with modes at 44, 69 and 99 *cm*, respectively. The first mode accounts for the distance between a and b (the two undergraduate friends). The second mode accounts for the distances between c, d, e and f (the three PhD students and the assistant professor belonging to the same research group). The third mode accounts mainly for the distances between a and e and between b and c (the only pairs where the members were unacquainted before the experiments). In this condition too, the physical distances comply with the social information, even though the distance between the assistant professor and the PhD students does not reflect the difference of status.

In Condition 3 (1.5×2 meters), the longest stable period lasted for 914 frames. People form a circular F-formation, giving now rise to four distinct modes in the space of the pairwise distances (see Fig. 5-row (iii)). Once again, two close friends a and b stand at the closest distances, separated from the rest of the subjects. In particular, subjects b and c stand at a very high distance if compared to the other measurements. This highlights the separation that holds between subjects that have different status, i.e., the student and the assistant professor.

The variations across the different conditions suggest the following considerations:

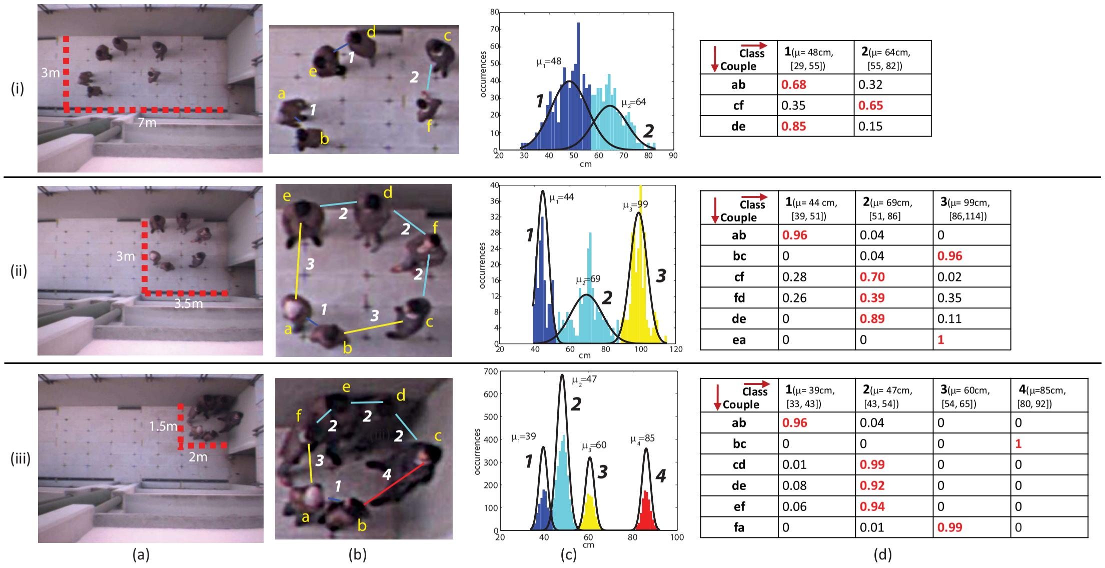
- The histograms show that the modes correspond to shorter distances as the space gets smaller. However, different social relations still result into different modes.
- The fraction of distances that fall in the first mode is 67% in Condition 1, 34% in Condition 2, and 22% in Condition 3.

In other words, the results confirm the findings about the effect of the space at disposition of interpersonal distances and, in particular, the effects of [75] stating that subjects prefer to keep higher distances when the environment gets smaller.

The results shown here analyzed the longest stable period in each session. Anyway, in all the other stable periods, the results were qualitatively similar.

2.4.3 Results of Session 2

The second session involved 7 subjects (see Fig. 6): five undergraduate students acquainted with one another (subjects a, b, c, d and g), two PhD students that are close friends (subjects e and f), and the representative of the students in the School of Computer Science (subject c).

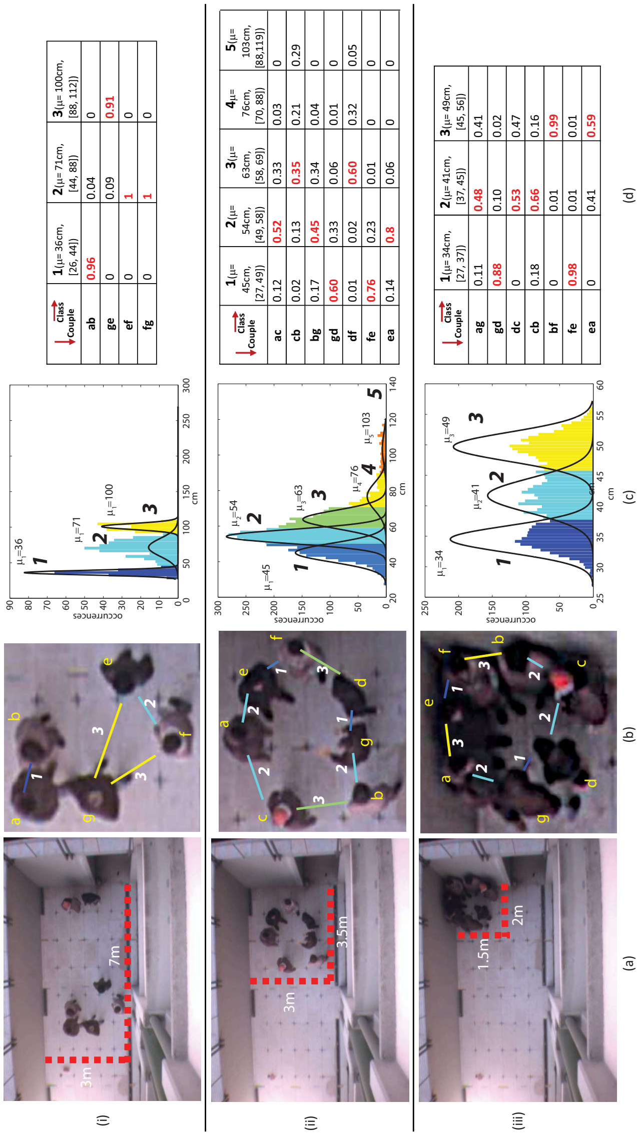


Class → Couple ↓	1(μ= 48cm, [29, 55])	2(μ= 64cm, [55, 82])
ab	0.68	0.32
cf	0.35	0.65
de	0.85	0.15

Class → Couple ↓	1(μ= 44 cm, [39, 51])	2(μ= 69cm, [51, 86]	3(μ= 99cm, [86,114])
ab	0.96	0.04	0
bc	0	0.04	0.96
cf	0.28	0.70	0.02
fd	0.26	0.39	0.35
de	0	0.89	0.11
ea	0	0	1

Class → Couple ↓	1(μ= 39cm, [33, 43])	2(μ= 47cm, [43, 54])	3(μ= 60cm, [54, 65])	4(μ=85cm, [80, 92])
ab	0.96	0.04	0	0
bc	0	0	0	1
cd	0.01	0.99	0	0
de	0.08	0.92	0	0
ef	0.06	0.94	0	0
fa	0	0.01	0.99	0

Fig. 5 The pictures of column (a) show the physical space in which people were free to move. The pictures in column (b) are zoomed versions of those in (a), showing the F-formations detected in each of the three stages. The color of the links corresponds to the color of the most frequent mode to which the distances between the linked individuals belongs to. Rows (i)-(ii)-(iii) refer to Condition 1-2-3, respectively (see text). Histograms in column (c) show the distributions of the distances and the related clustering. The tables in column (d) report the fraction of time distances between each pair of adjacent individuals belong to a given mode. Each mode is identified by the mean, and by the range (in centimeters) of distances it covers (written in squared parentheses). The figure is best viewed in colors.



Couple ↓ Class →	**1**(μ= 36cm, [26, 44])	**2**(μ= 71cm, [44, 88])	**3**(μ= 100cm, [88, 112])
ab	**0.96**	0.04	0
ge	0	0.09	**0.91**
ef	0	**1**	0
fg	0	**1**	0

Couple ↓ Class →	**1**(μ= 45cm, [27, 49])	**2**(μ= 54cm, [49, 58])	**3**(μ= 63cm, [58, 69])	**4**(μ= 76cm, [70, 88])	**5**(μ= 103cm, [88,119])
ac	0.12	**0.52**	0.33	0.03	0
cb	0.02	0.13	**0.35**	0.21	0.29
bg	0.17	**0.45**	0.34	0.04	0
gd	**0.60**	0.33	0.06	0.01	0
df	0.01	0.02	**0.60**	0.32	0.05
fe	**0.76**	0.23	0.01	0	0
ea	0.14	**0.8**	0.06	0	0

Couple ↓ Class →	**1**(μ= 34cm, [27, 37])	**2**(μ= 41cm, [37, 45])	**3**(μ= 49cm, [45, 56])
ag	0.11	**0.48**	0.41
gd	**0.88**	0.10	0.02
dc	0	**0.53**	0.47
cb	0.18	**0.66**	0.16
bf	0	0.01	**0.99**
fe	**0.98**	0.01	0.01
ea	0	0.41	**0.59**

Fig. 6 The pictures of column (a) show the physical space in which people were free to move. The pictures in column (b) are zoomed versions of those in (a), showing the F-formations detected in each of the three stages. The color of the links corresponds to the color of the most frequent mode to which the distances between the linked individuals belongs to. Rows (i)-(ii)-(iii) refer to Condition 1-2-3, respectively (see text). Histograms in column (c) show the distributions of the distances and the related clustering. The tables in column (d) report the fraction of time distances between each pair of adjacent individuals belong to a given mode. Each mode is identified by the mean, and by the range (in centimeters) of distances it covers (written in squared parentheses). The figure is best viewed in colors.

In Condition 1 (see Fig. 6-row (i)), the group has split into F-formations including $2-3$ people each. Fig. 6 shows the picture of the configuration that has lasted for the longest time (152 frames). The interpersonal distances cluster according to three modes. In the F-formation including three people, the two PhD students (who are close friends) appear to be closer (on average) than the third component (an undergraduate student they are not acquainted with them).

In Condition 2 (see Fig. 6-row (ii)), the most stable configuration is a circle that holds for 629 frames. In this case, the modes are five, but only the first three are used to a significant extent (see the tables of column (d) with the fractions of time distances belong to a given Gaussian component). The two PhD students (e and f) and two undergraduate students (g and d) appear to be closer to one another than the other participants. In the former case, this reflects the fact that they were close friends before the experiment, whereas in the latter, it corresponds to the fact that the two students have a romantic relationship, as it emerged from the questionnaires collected after the experiments. The situation for the other participants is less clear, but this probably happens because all participants are students and their social distances are thus similar. The only factors that seem to make some students closer (see above) are then personal.

In Condition 3 (see Fig. 6-row (iii)), a circular F-formation holds for 592 frames and corresponds to the longest stable interval. There are three modes visibile in the histogram. The PhD students are clearly separated from the rest of the circle (distances belonging to the third mode), while they are very close to one other. The couple (d and g) is tighter than the other dyads as well. In this case again, closer personal relations result into smaller distances.

It is worth to note that the effect of the amount of space at disposition leads to the same conclusions as in session 1 (see end of Section 2.4.2).

2.5 Remarks

This section has presented a study and preliminary experiments on the inference of social relations from interpersonal distances measured automatically via a computer vision approach. The results show that, in accordance with the findings of proxemics, people involved in casual standing interactions tend to get closer when their social relation is more intimate. The experiments have been performed on a limited number of individuals (13 in total), but the setting is fully unconstrained and spontaneous and the results appear to be consistent with the expectations.

An unsupervised analysis of interpersonal distances reveals that the four zones predicted by Hall in his seminal work emerge independently of the space at disposition of the interactants. The radii of the concentric zones are smaller than those measured in [30, 31] for Northern-Americans, but this should not be surprising as the subjects are from Italy, a culture likely to be more "contact" than the American one. Furthermore, the space available to the subjects has been progressively reduced and this has further contributed to reduce the size of the zones. The effects expected from the reduction of the space have been actually observed, especially when it comes to the tendency to increase interpersonal distances.

The detection of the F-formations appears to be crucial to perform a correct analysis of the interpersonal distances. In fact, previous works in the literature did not consider the geometric constraints imposed by the F-formations and the results have been inconsistent. In contrast, by limiting the analysis only to the distances of neghboring (adjacent) people, our experiments obtain results where social and physical distances match one another.

The next steps to be performed include not only experiments including a larger number of subjects, but also an attempt to use the statistical distributions learned from the data to predict automatically the degree of intimacy between individuals. This would represent a major step towards the development of socially intelligent surveillance technologies.

3 Voice Activity Detection Using Visual Cues in Groups

Following the analysis of non-verbal cues for the detection of social signals, our last contribution is related to the characterization of the group interactions by proposing a Voice Activity Detection approach only based on the automatic measurement of the persons' gesturing activities [16]. This work takes inspiration from the observation that people accompany speech with gestures, the range of visible bodily actions that are, more or less, generally regarded as part of a person‘s willing expression ([40]). Far from being independent phenomena, speech and gestures are so tightly intertwined that every important investigation of language has taken gestures into account, from De Oratore by Cicero (1st Century B.C.) to the latest studies in cognitive sciences ([52, 38]) showing that the two modalities are components of a single overall plan ([40]).

This work presents a method for estimating the level of gesturing as a means to perform Voice Activity Detection (VAD), i.e. to automatically recognize whether a person is speaking or not. The main rationale is that audio, the most natural and reliable channel when it comes to VAD, might be unavailable for technical, legal, privacy related issues or simply for a noisy scenario. A condition that applies in particular to surveillance scenarios where people are monitored in public spaces and are not necessarily aware of being recorded.

Previous works take advantage of restrictive experimental setups in a smart meeting room [35], deploying a system ”in the wild” designing a more credible setup for a video surveillance system. We use solely visual cues obtained from only one camera positioned 7 meters above the scene. In particular, the experiments focus on people involved in standing conversations, with an automatic person tracking system that follows each individual. Our VAD method is based on a local optical flow-based descriptor extracted for each individual body, that encodes its energy and complexity using an entropy-like measure. This allows one to discriminate between body oscillations or noise introduced by the tracker, where the optical flow is low and homogeneous, and genuine gestures, where the movement of head, arms and trunk produces a local flow field which is diverse in both intensity and direction.

The descriptor extracted for each participant produces a signal that can be used for VAD. The proposed approach is interesting for three main aspects. First, the relationship between speech and gestures has been widely documented and studied, but relatively few quantitative investigations of this phenomenon have been made. Second, approaches similar to ours might help to infer information about privacy protected data (speech in this case) from publicly accessible data (gestures in this case): this is also important for establishing whether the simple absence of a certain channel is sufficient to protect the privacy of people and how much. Finally, inferring missing data from available ones can make techniques dealing with challenging scenarios more effective and reliable.

As in the previous section, we suppose to have tracked each individual and additionally to have detected the F-formation. Thus, a square *Region of Interest* (ROI) is defined around each person. The size of the ROI is set automatically to include all gestures of the individual. Areas where multiple ROIs overlap have been ignored to avoid possible confusions between neighboring people.

The measurement technique is applied to each ROI individually and it is expected to accomplish two goals: the first is to discriminate between gestures and postural oscillations typically observed when people stand. The second is to normalize the tracking errors that cause abrupt and spurious shifts of the ROI. The body parts most commonly involved in gesturing are hands, arms, head, and trunk. Their individual movements tend to be very different during gesturing and the measurement values associated to a given ROI try to capture such an aspect:

$$v(t) = \max_{\text{int}}(\{f(t)\}) \times S_{\text{int}}(\{f(t)\}) \times S_{\text{ori}}(\{f(t)\}) \tag{4}$$

where $\{f(t)\}$ is the set of motion flow vectors associated to each pixel of the ROI at time t, $S_{\text{int}}(\{f(t)\})$ is the entropy of the motion flow intensities, and $S_{\text{ori}}(\{f(t)\})$ is the entropy of the orientation values, both calculated over $\{f(t)\}$[6]. The maximum over the flow intensities values $\max_{\text{int}}(\{f(t)\})$ encodes the "energy" associated to the movement, while the two entropic terms serve to highlight those motion flow values which exhibit higher variability in intensity and orientation. In this way, postural oscillations and shifts due to unprecise tracking receive a low score because they cause a global, homogeneous set of intensities and orientations, corresponding to low entropy values. Alternative expressions of $v(t)$ have been considered that use mean and median rather than maximum, or do not include one of the entropy terms. In all cases, the resulting performance is lower than the one obtained with the expression above. A graphical idea of the measurement is given in Figure 7 where colours shift towards red when gesturing activity is higher.

[6] The optical flow has been obtained with the package available at the following URL: `http://server.cs.ucf.edu/~vision/source.html`.

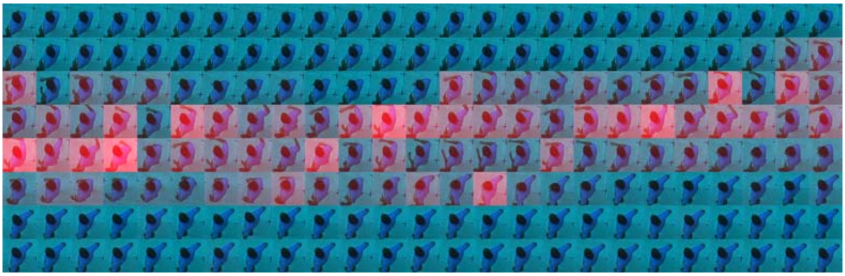


Fig. 7 Qualitative analysis of our descriptor: in the sequence above, an high tonality of red means great gesture activity.

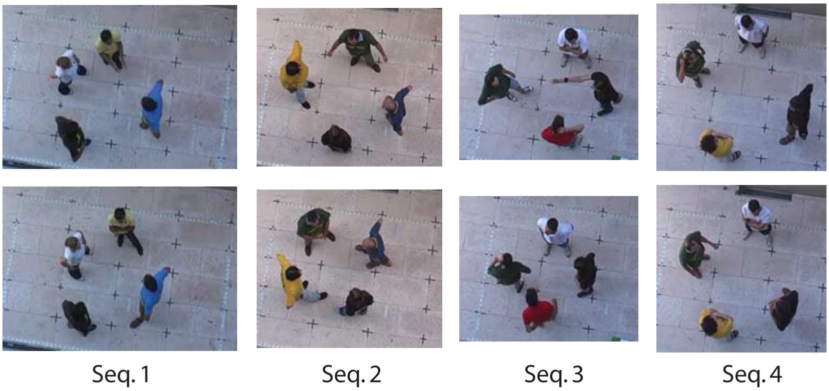


Fig. 8 Some frames of the video sequences used

3.1 Experiments on the Visual VAD

The goal of the experiments is twofold: first, to provide a quantitative measure of the correlation between gestures and speech; second, to measure the effectiveness of the function $v(t)$ (see Section 3.1.2) in a VAD task. Both tasks have been accomplished over *TalkingHeads*, a new dataset publicly available upon request[7] (see some frames in Figure 8).

The dataset contains four conversations lasting, on average, 6 minutes. The data was recorded in a 3.5 × 2.5 meters wide outdoor area, during a cloudy day in summer. The total number of subjects is 15 (1 female and 14 males), with 4 different participants per conversation (only one subject participated in two conversations). The subjects include 4 academics, 5 undergraduate students, 2 MSc students, 3 postdoctoral researchers, and 1 PhD student. The ages range between 20 and 40 years and the subjects were unaware of the actual goals of the experiments.

[7] http://profs.sci.univr.it/~cristanm/datasets/TalkingHeads/

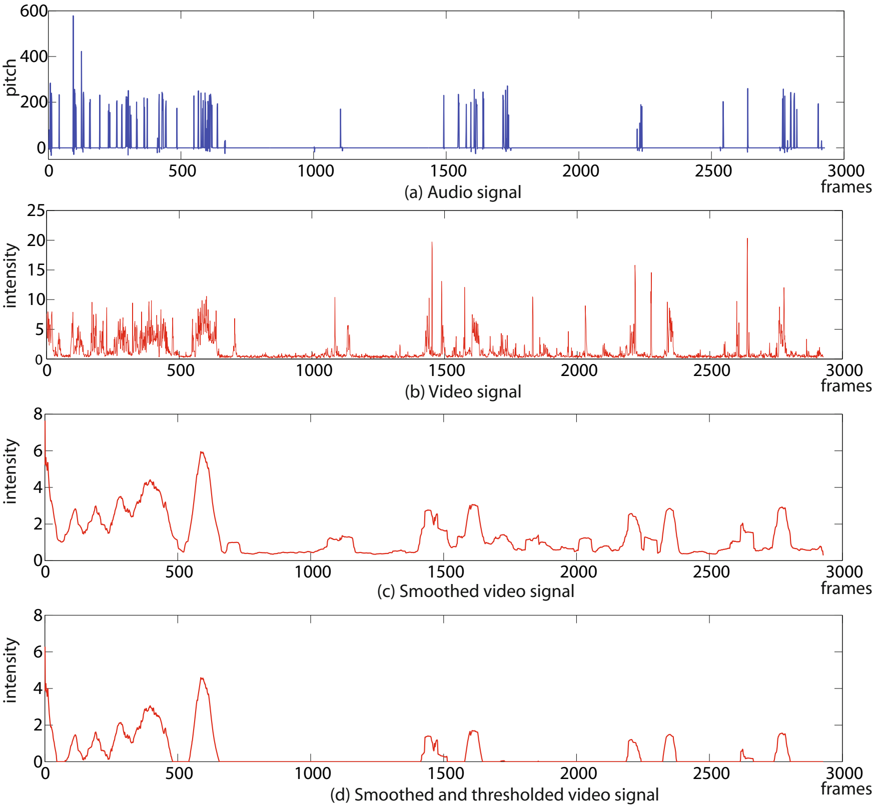


Fig. 9 Examples of signals employed in the analysis. (a) Audio input signal. (b) Video signal produced by our descriptor of a subject involved in the Seq.1. (c) The video signal was smoothed for evaluating the crossmodal correlation (Sec. 3.1.1). (d) The video signal was thresholded for the audio classification (Sec. 3.1.2).

Data were captured at 25 frames per second with a camera positioned 7 meters above the floor and facing downward. The subjects were asked to wear differently colored shirts, in order to make the tracking/localization easier. Tracking has been performed by simple template association. The motion flow has been computed by considering one frame every 4, reducing the video sampling period to 160 *ms*. The audio was recorded at 44100 Hz with 4 wireless headset microphones, each transmitting to its own receiver.

Each audio recording has been segmented into speech and non-speech segments using a robust VAD algorithm based on pitch [41]. This latter was extracted at regular time steps of 10 *ms* with Praat [8], a package including the pitch extraction technique described in [7]. The motivation behind this choice is not only that silence segments are characterized by frequencies way higher than those observed in speech, but also that the pitch tends to be correlated with the "*beat*" gesture

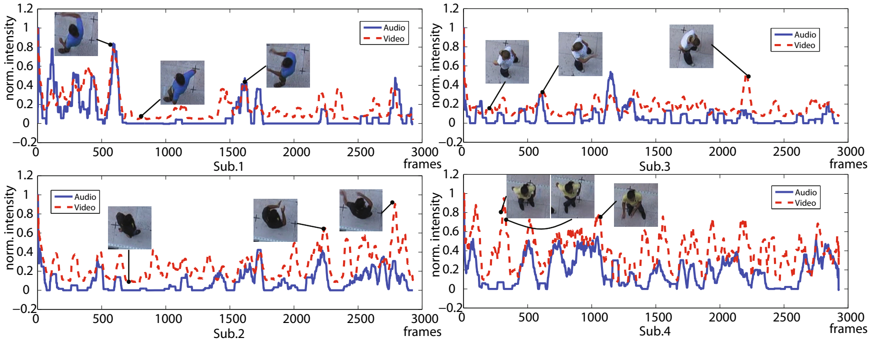


Fig. 10 Visual analysis of the audio and video smoothed data: each plot depicts the smoothed audio (solid blue) and the smoothed video (dashed red) signals for each participant to the dialog. The thumbnails give the feeling of the gesturing activity carried out in a particular instant.

typically accompanying syllables where the intonation is stressed [11, 74]. Then, in order to synchronize audio and video data, audio was resampled according to the video frame rate, averaging the pitch values occurring in each time period. The averaged pitch values constituted the samples of the audio signal that will be analyzed in the following.

3.1.1 Pitch-Gesturing Correlation Analysis

This section shows how the correlation between the pitch (as extracted with Praat), and the gesturing activity (as measured with the approach proposed above) has been measured.

After the application of the techniques described in the previous sections, each sequences results into two signals per person, showing the value of pitch and $v(t)$ at regular time steps of 160 ms. Plots (a) and (b) of Figure 9 provide an example of such signals. The simple visual inspection shows that the two signals tend to change according to one another. However, $v(t)$ appears to be more noisy of the pitch because of the sensibility of the optical flow. Hence, both signals have been smoothed with an average filter applied to 8 s long windows. Figure 9 (c) shows the smoothed version of $v(t)$, while the smoothed audio and video signals of a complete conversation, normalized with respect to their maximum value, are compared in Fig. 10.

Table 1 reports the Pearson correlation coefficients between $v(t)$ and pitch. Off-diagonal values account for correlations between signals extracted from different individuals. In this way, it is possible to better assess how strong is the correlation between speech and gestures for a given individual.

Table 1 Quantitative measures: correlation coefficients matrix for Seq. 1 . The matrix rows and columns corresponds respectively to the four subsampled video signals (Vsub) and the four subsampled audio signals (Asub) (the non-significant coefficients (p-value$\geq$ 0.05) are underlined in red.

	A sub.1	A sub.2	A sub.3	A sub.4
V sub.1	**0.7310**	0.1338	0.2490	0.0670
V sub.2	0.1900	**0.6454**	0.4460	0.0254
V sub.3	0.1867	0.1966	**0.4838**	-0.0356
V sub.4	-0.2592	0.0472	0.0389	**0.4204**

We performed a similar analysis on the other conversations, with the same parameters, obtaining in total four correlation matrices. Mediating over all the entries in the main diagonal (they were all statistically significant), we obtained a mean correlation score of 0.53, while considering the statistically significant off-diagonals entries we get 0.19. This suggests that $v(t)$ might be a reliable indicator of voice activity. Hence, in the following section, we show how the video signal can be employed to perform VAD.

3.1.2 Voice Activity Detection

The VAD task proposed in this section consists of labeling each frame as *speech* or $non-speech$. As an approximation, each person is treated independently of the others even though the exchange of turns (the opportunity of speaking) tends to follow regularities that might be helpful in improving the performance. The original pitch signal, which has non-zero entries only when the subjects talk, is used as groundtruth.

As a video signal to be used to infer speech, we considered the smoothed signal described above for the correlation analysis. In this way, high frequency components of the original signal have been filtered. The discrimination between speech and non-speech samples has been performed with a thresholding technique. Essentially, as suggested by Fig. 9 and Fig. 10, the video signal has a continuous component caused by small values of optical flow that are always present in the analysis. For this reason, we subtracted the mean to the signal, and we keep the intensities above zero, setting them at 1's. Smoothing and subtraction of the mean represent a thresholding operation that does not need the tuning of any parameter.

At this point, we can compare the two signals, and the detailed analysis of Seq. 1 is shown in Fig. 11.

For the sake of clarity, we report in the figure the (normalized) continuous signals, and not their binary versions which were actually used. As visible, many of the speech samples are correctly captured by the video signal. The figure also reports the precision, recall and accuracy values. In this sequence, the classifier tends to have low recall and high precision (assuming the speech as positive values). Considering

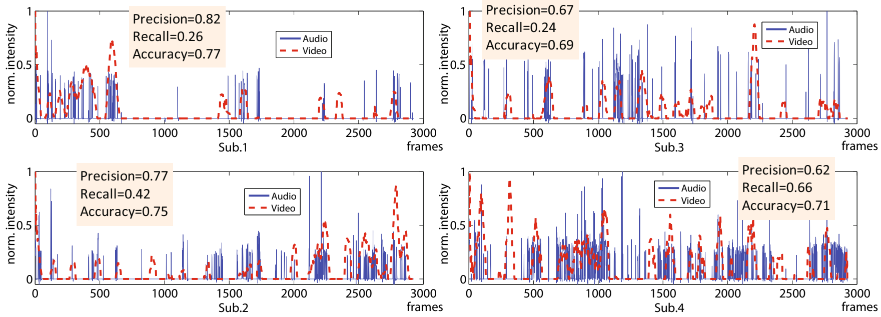


Fig. 11 Audio classification by video analysis. Each plot portrays the audio (solid blue) and the video (dashed red) signals for each participant to the dialog. For the sake of clarity, we report the (normalized) continuous signals, and not their binary versions (that we used). Precision, recall and accuracy scores related to each individual are also indicated.

all the subjects employed, we reach an average accuracy of 71%, average precision of 67%, and average recall of 40%.

4 Remarks

This work has proposed a gesturing-based approach for performing VAD, the automatic detection of people that speak. The reason for using gestures in VAD, typically performed using speech recordings, is that the use of microphones is difficult or illegal in many scenarios of potential interest, including surveillance of public spaces, monitoring of potentially dangerous plants, etc. The core idea behind the approach is that cognitive sciences have demonstrated that speech and gestures, far from being independent expression modalities, are two faces of the same phenomenon. Therefore, gestures can be considered a reliable evidence of speech taking place at the same time.

The preliminary results presented in this paper provide a quantitative confirmation of the finding above and, most importantly, show that the detection of gesturing activity helps to predict whether a person is speaking or not with an accuracy of 71 percent (on a frame-by-frame basis). While not being conclusive about the possibility of reconstructing the actual turns and of performing diarization, the results are certainly promising in the direction of reconstructing conversational dynamics in absence of audio. This appears particularly important as turn-organization has been widely shown to be fundamental in inferring socially important information such as roles, dominance, personality, etc [71].

Besides, this work shows that it is possible to infer information about missing data (speech in this case) from available evidence (videos in this case). In a surveillance setup like the one of the experiments, this opens two conflicting perspectives: on one hand, surveillance approaches can be significantly improved by predicting phenomena considered so far non-accessible with the sensors at disposition. On the other hand, privacy protection measures applied so far (i.e., legal limitation on the use of microphones in public spaces) might become obsolete and uneffective. In this respect, experiments of the type presented in this work might change the notion of privacy and of its protection.

Future work can take two major directions: the first is to move from VAD to full diarization. This requires the application of probabilistic sequential models taking into account temporal constraints between neighboring frames and a larger amount of data. The second is to try automatic conversation analysis based on gestures and to verify whether (and to what extent) it is possible to perform tasks like role recognition, conflict detection, etc., typically performed using turn-organization and other conversational cues.

5 Conclusions

The realms of automated surveillance and monitoring tend to focus solely on Computer Vision and Pattern Recognition (CVPR) techniques, neglecting social, affective and emotional aspects of human behavior even if this is, in ultimate analysis, their main subject of interest. Actually, the cross-pollination between social psychology and CVPR could lead to new research questions as well as to application domains that, so far, have not been the subject of attention in the computing community. In this chapter we show how the modeling of groups of people may be performed by considering social and psychological theories: in particular, we analyze the detection of groups, their characterization in terms of social links among the participants, and the inference of speech data from video cues only. Due to our initial good results, we are deeply convinced that the cross-fertilization of human and computer sciences for surveillance and monitoring is going to be inevitably extended, and only in this way a new generation of surveillance systems can be designed, making the necessary jump to go beyond the current technology, so far advanced in incremental steps.

Acknowledgements. The authors will thanks Andrea Fossati, Alessio Del Bue, Loris Bazzani, Alessandro Vinciarelli, Anna Pesarin, Giulia Paggetti, Diego Tosato for their remarkable help in the development of the approaches reported in this chapter.

References

1. Adams, L., Zuckerman, D.: The effects of lighting conditions on personal space requirement. Journal of General Psychology 118(4), 335–340 (1991)
2. Aggarwal, J.K., Park, S.: Human motion: Modeling and recognition of actions and interactions. In: 2nd International Symposium on Proceedings of the 3D Data Processing, Visualization, and Transmission, 3DPVT 2004, pp. 640–647. IEEE Computer Society Press, Washington, DC (2004)
3. Altman, I.: The Environment and Social Behavior: Privacy, Personal Space, Territory, and Crowding. Brooks/Cole Publishing Company, Monterey, CA (1975)

4. Arulampalam, M., Maskell, S., Gordon, N.: A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Transactions on Signal Processing 50, 174–188 (2002)
5. Baxter, J.: Interpersonal spacing in natural settings. Sociometry 33(4), 444–456 (1970)
6. Bazzani, L., Tosato, D., Cristani, M., Farenzena, M., Pagetti, G., Menegaz, G., Murino, V.: Social interactions by visual focus of attention in a three-dimensional environment. Expert. Systems 30(2), 115–127 (2013)
7. Boersma, P.: Accurate short term analysis of the fundamental frequency and the harmonics to noise ratio of a sampled sound. IEEE Transactions on Image Processing 17, 97–110 (1993)
8. Boersma, P.: Praat, a system for doing phonetics by computer. Glot International 5(9/10), 341–345 (2001)
9. Breazeal, C.: Designing Sociable Robots. MIT Press, Cambridge (2002)
10. Brown, L., Tian, Y.: Comparative study of coarse head pose estimation. In: Proc. Motion and Video Computing Workshop, pp. 125–130 (2002)
11. Cassell, J., Steedman, M., Badler, N., Pelachaud, C., Stone, M., Douville, B., Prevost, S., Achorn, B.: Modeling the interaction between speech and gesture. In: Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society, pp. 153–158 (1994)
12. Cheng, Z., Qin, L., Huang, Q., Jiang, S., Tian, Q.: Group activity recognition by gaussian processes estimation. In: 2010 20th International Conference on Pattern Recognition (ICPR), pp. 3228–3231 (August 2010)
13. Cochran, C., Personal, D.: space requirements in indoor versus outdoor locations. Journal of Psychology 117, 121–123 (1984)
14. Cochran, C., Urbanczyk, D., The, S.: The effect of availability of vertical space on personal space. Journal of Psychology 111, 137–140 (1982)
15. Cristani, M., Bazzani, L., Paggetti, G., Fossati, A., Bue, A.D., Tosato, D., Menegaz, G., Murino, V.: Social interaction discovery by statistical analysis of f-formations. In: Proceedings of British Machine Vision Conference (2011)
16. Cristani, M., Pesarin, A., Vinciarelli, A., Crocco, M., Murino, V.: Look at who's talking: Voice activity detection by automated gesture analysis. In: Proceedings of the Workshop on Interactive Human Behavior Analysis in Open or Public Spaces, InterHub 2011 (2011)
17. Cristani, M., Murino, V.: Vinciarelli, A.: Socially intelligent surveillance and monitoring: Analysing social dimensions of physical space. In: First IEEE International Workshop on Socially Intelligent Surveillanceand Monitoring (SISM 2010), San Francisco, California (2010)
18. Cristani, M., Paggetti, G., Vinciarelli, A., Bazzani, L., Menegaz, G., Murino, V.: Towards computational proxemics: Inferring social relations from interpersonal distances. In: SocialCom/PASSAT, pp. 290–297 (2011)
19. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc. B 39, 1–38 (1977)
20. Duda, R., Hart, P., Stork, D.: Pattern Classification. John Wiley and Sons (2001)
21. Figueiredo, M., Jain, A.: Unsupervised learning of finite mixture models. IEEE Trans. on Pattern Analysis and Machine Intelligence 24(3), 381–396 (2002)
22. Freeman, L.: Social networks and the structure experiment. In: Research Methods in Social Network Analysis, pp. 11–40 (1989)
23. Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition. IEEE (2009)
24. Gatica-Perez, D.: Automatic nonverbal analysis of social interaction in small groups: a review. Image and Vision Computing (2009)

25. Gatica-Perez, D.: Automatic nonverbal analysis of social interaction in small groups: a review. Image and Vision Computing 27(12), 1775–1787 (2009)
26. Gifford, R., O'Connor, B.: Nonverbal intimacy: clarifying the role of seating distance and orientation. Journal of Nonverbal Behavior 10(4), 207–214 (1986)
27. Griffitt, W., Veitch, R.: Hot and crowded: Influences of population density and temperature on interpersonal affective behavior. Joumal of Personality and Social Psychology 17, 92–98 (1971)
28. Groh, G., Lehmann, A., Reimers, J., Friess, M.R., Schwarz, L.: Detecting social situations from interaction geometry. In: Proceedings of the 2010 IEEE Second International Conference on Social Computing, SOCIALCOM 2010, pp. 1–8. IEEE Computer Society, Washington, DC (2010), http://dx.doi.org/10.1109/SocialCom.2010.11
29. Hall, E.: The hidden dimension. Doubleday New York (1966)
30. Hall, E.: Handbookfor proxemic research. Studies in the anthropologyof visual communication series. Society for the Anthropology of Visual Communication, Washington, DC (1974)
31. Hall, R.: The hidden dimension, New York (1966)
32. Helbing, D., Molnár, P.: Social force model for pedestrian dynamics. Physical Review E 51(5), 4282–4287 (1995)
33. Heshka, S., Nelson, Y.: Interpersonal speaking distance as a function of age, sex, and relationship. Sociometry 35(4), 491–498 (1972)
34. Hongeng, S., Nevatia, R.: Large-scale event detection using semi-hidden markov models. In: IEEE International Conference on Computer Vision, vol. 2 (2003)
35. Hung, H., Ba, S.O.: Speech/non-speech detection in meetings from automatically extracted low resolution visual features. In: ICASSP, pp. 830–833 (2010)
36. Ivanov, Y.A., Bobick, A.F.: Recognition of visual activities and interactions by stochastic parsing. IEEE Trans. Pattern Anal. Mach. Intell. 22, 852–872 (2000)
37. Jebara, T., Pentland, A.: Action reaction learning: Automatic visual analysis and synthesis of interactive behaviour. In: Proceedings of the First International Conference on Computer Vision Systems, ICVS 1999, pp. 273–292. Springer, London (1999)
38. Kendon, A.: Gesticulation and speech: Two aspects of the process of utterance. The Relationship of verbal and Nonverbal Communication, 207–227 (1980)
39. Kendon, A.: Conducting Interaction: Patterns of behavior in focused encounters (1990)
40. Kendon, A.: Language and gesture: unity or duality?, pp. 47–63. Cambridge University Press (2000)
41. Khondaker, A., Ghulam, M.: Improved noise reduction with pitch enabled voice activity detection. In: ISIVC 2008 (2008)
42. Knapp, M., Hall, J.: Nonverbal Communication in Human Interaction. Harcourt Brace College Publishers (1972)
43. Koay, K.L., Syrdal, D.S., Walters, M.L., Dautenhahn, K.: Living with robots: Investigating the habituation effect in participants? preferences during a longitudinal human-robot interaction study. In: ROMAN 2007 the 16th IEEE International Symposium on Robot and Human Interactive Communication, pp. 564–569 (2007), http://hdl.handle.net/2299/1880
44. Kuzuoka, H., Suzuki, Y., Yamashita, J., Yamazaki, K.: Reconfiguring spatial formation arrangement by robot body orientation. In: Proceeding of the 5th ACM/IEEE International Conference on Human-Robot Interaction, HRI 2010, pp. 285–292. ACM, New York (2010), http://doi.acm.org/10.1145/1734454.1734557
45. Lan, T., Wang, Y., Yang, W., Mori, G.: Beyond actions: Discriminative models for contextual group activities. In: Advances in Neural Information Processing Systems, NIPS (2010)

46. Lanz, O.: Approximate bayesian multibody tracking. IEEE Trans. on Pattern Analysis and Machine Intelligence (2006)
47. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV Workshop on Statistical Learning in Computer Vision, pp. 17–32 (2004)
48. Lin, W., Sun, M.T., Poovendran, R., Zhang, Z.: Group event detection with a varying number of group members for video surveillance. IEEE Transactions on Circuits and Systems for Video Technology 20(8), 1057–1067 (2010)
49. Lott, D., Sommer, R.: Seating arrangements and status. Journal of Personality and Social Psychology 7(1), 90–95 (1967)
50. Mantel, N.: The detection of disease clustering and a generalized regression approach. Cancer Research 27(2), 209 (1967)
51. Mazur, A.: On Wilson's Sociobiology. American Journal of Sociology 82(3), 697–700 (1976)
52. McNeill, D.: Hand and mind: What gestures reveal about thought. Chicago University Press, Chicago (1992)
53. Michalowski, M.P.: A spatial model of engagement for a social robot. In: Proceedings of the 9th International Workshop on Advanced Motion Control, AMC 2006 (2006)
54. Nakauchi, Y., Simmons, R.: A social robot that stands in line. In: Proceedings of the Conference on Intelligent Robots and Systems (IROS 2000) (October 2000)
55. Ni, B., Yan, S., Kassim, A.A.: Recognizing human group activities with localized causalities. In: CVPR 2009, pp. 1470–1477 (2009)
56. Oliver, N., Rosario, B., Pentland, A.: Graphical models for recognising human interactions. In: Advances in Neural Information Processing Systems (1998)
57. Pacchierotti, E., Christensen, H.I., Jensfelt, P.: Human-robot embodied interaction in hallway settings: A pilot user study. In: Proceedings of the 2005 IEEE International Workshop on Robots and Human Interactive Communication, pp. 164–171 (2005)
58. Park, S., Trivedi, M.M.: Multi-person interaction and activity analysis: a synergistic track- and body-level analysis framework. Mach. Vision Appl. 18, 151–166 (2007)
59. Pellegrini, S., Ess, A., Schindler, K., Gool, L.V.: You'll never walk alone: modeling social behavior for multi-target tracking. In: Proc. 12th International Conference on Computer Vision, Kyoto, Japan (2009)
60. Richmond, V., McCroskey, J.: Nonverbal Behaviors in interpersonal relations. Allyn and Bacon (1995)
61. Robertson, N., Reid, I.D.: Estimating gaze direction from low-resolution faces in video. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 402–415. Springer, Heidelberg (2006)
62. Robertson, N., Reid, I.: Automatic reasoning about causal events in surveillance video 2011 (2011)
63. Russo, N.: Connotation of seating arrangements. The Cornell Journal of Social Relations 2(1), 37–44 (1967)
64. Savinar, J.: The effects of ceiling height on personal space. Man-Environment Systems 5, 321–324 (1975)
65. Scovanner, P., Tappen, M.: Learning pedestrian dynamics from the real world, pp. 381–388 (2009)
66. Smith, H.: Territorial spacing on a beach revisited: A cross-national exploration. Social Psychology Quarterly, 132–137 (1981)
67. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: Int. Conf. Computer Vision and Pattern Recognition (CVPR 1999), vol. 2, pp. 246–252 (1999)

68. Takayama, L., Pantofaru, C.: Influences on proxemic behaviors in human-robot interaction. In: Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009, pp. 5495–5502. IEEE Press, Piscataway (2009), http://portal.acm.org/citation.cfm?id=1732643.1732940
69. Tosato, D., Farenzena, M., Spera, M., Murino, V., Cristani, M.: Multi-class classification on riemannian manifolds for video surveillance. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 378–391. Springer, Heidelberg (2010)
70. Vinciarelli, A., Pantic, M., Bourlard, H.: Social Signal Processing: Survey of an emerging domain. Image and Vision Computing Journal 27(12), 1743–1759 (2009)
71. Vinciarelli, A., Pantic, M., Heylen, D., Pelachaud, C., Poggi, I., D'Errico, F., Schröder, M.: Bridging the gap between social animal and unsocial machine: A survey of social signal processing. IEEE Transactions on Affective Computing (2011) (to appear)
72. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. IEEE Trans. Pattern Anal. Mach. Intell. 31, 539–555 (2009)
73. Watson, O.: Proxemic behavior: A cross-cultural study. Mouton De Gruyter (1970)
74. Wells, G., Petty, R.: The effects of over head movements on persuasion. Basic and Applied Social Psychology 1(3), 219–230 (1980)
75. White, M.J.: Interpersonal distance as affected by room size, status, and sex. The Journal of Social Psychology 95(2), 241–249 (1975)
76. Zen, G., Lepri, B., Ricci, E., Lanz, O.: Space speaks: towards socially and personality aware visual surveillance. In: Proceedings of the 1st ACM International Workshop on Multimodal Pervasive Video Analysis, MPVA 2010, pp. 37–42. ACM, New York (2010)

Mobile Computational Photography with FCam

Kari Pulli and Alejandro Troccoli

Abstract. In this chapter we cover the FCam (short for Frankencamera) architecture and API for computational cameras. We begin with the motivation, which is flexible programming of cameras, especially of camera phones and tablets. We cover the API and several example programs that run on the NVIDIA Tegra 3 prototype tablet and the Nokia N900 and N9 Linux-based phones. We discuss the implementation and porting of FCam to different platforms. We also describe how FCam has been used at many universities to teach computational photography.

1 Frankencamera: An Experimental Platform for Computational Photography

The Frankencamera platform creates an architecture for computational photography. The system was originally created in a joint research project between Nokia Research Center and Stanford University, in teams headed by Kari Pulli and Marc Levoy, respectively. It was described at SIGGRAPH 2010 by Adams *et al.*[1], and an open source implementation of the FCam API was also released in summer 2010. In this chapter we describe the motivation for this architecture, its key components, existing implementations, and some applications enabled by FCam.

1.1 Computational Photography

The term computational photography is today understood as a set of imaging techniques that enhance or extend the capabilities of digital photography. Often the output is an ordinary photograph, but one that could not have been taken by a traditional camera. Many of the methods try to overcome the limitations of normal cameras, often by taking several images with varying image parameters, and then combining the images, computing to extract more information out of the images, and synthe-

Kari Pulli · Alejandro Troccoli
NVIDIA Research, 2701 San Tomas Expressway, Santa Clara, CA 95050
e-mail: {karip,atroccoli}@nvidia.com

R. Cipolla et al. (eds.), *Registration and Recognition in Images and Videos*,
Studies in Computational Intelligence 532,
DOI: 10.1007/978-3-642-44907-9_11, © Springer-Verlag Berlin Heidelberg 2014

sizing an image that is in some way better than any of the input images [5, 6]. Some approaches modify the camera itself, especially the optical path, including the lens system and aperture through which the light travels before hitting a sensor [4].

Even though much of the computational photography predates modern mobile devices such as camera phones, a smartphone is in some sense an ideal platform for computational photography. A smartphone is a full computer in a convenient and compact package, with a large touch display, and at least one digital camera. The small form factor precludes some of the plays with novel optics, and makes it challenging to manufacture a high-quality camera system with sufficiently large lens and sensor that can obtain good images in reasonable lighting conditions. Precisely because of this challenge, the opportunity to collect more data from several input images, and combine them to produce better ones, makes computational photography an important part of mobile visual computing. However, mobile computational photography comes with the added requirement of being able to deal with hand-held cameras that are likely to move either during the image exposure (causing blur) or also between capturing of the images in a burst (causing ghosting as the same objects have moved).

1.2 FCam Architecture and API

Traditional camera APIs have usually been optimized for the common simple use cases such as taking an individual still image or capturing a video clip. If the setting of all camera parameters is automated, and the precise parameters and intervening image processing steps are not documented, it is difficult to properly combine the images to create better ones. This lack of control and transparency motivated the design of an experimental platform for computational photography. The FCam API has so far been implemented at Stanford for a large camera that accepts Canon SLR lenses (Frankencamera V2, or F2, see the inset in Figure 8), for two commercial Nokia smartphones (N900 and N9) running Linux, and on an NVIDIA Tegra 3 development tablet running Android.

Figure 1 illustrates the abstract Frankencamera architecture. A key innovation of this architecture with respect to the previous camera architectures lies in how the camera state is represented. Most traditional camera APIs combine the image sensor and image processor into a single conceptual camera object that has a global state: the current set of parameter values. However, a real camera sensor is a pipeline: while an image is being exposed, the capture parameters for the next image are being configured, and the previous image is being read out. Also the image processor is a pipeline: it first preprocesses the image in RAW or Bayer format, then demosaicks the image into an RGB and YUV image, and finally tonemaps the image so it can be displayed. If you now change the “state” of this camera system, the changed parameters may affect non-deterministically different images. In a streaming video application this is not so important, as the control algorithms change the values gradually and adaptively: the knowledge of exactly which frame is affected is often not crucial. For still imaging it is important that exactly the correct parameters affect

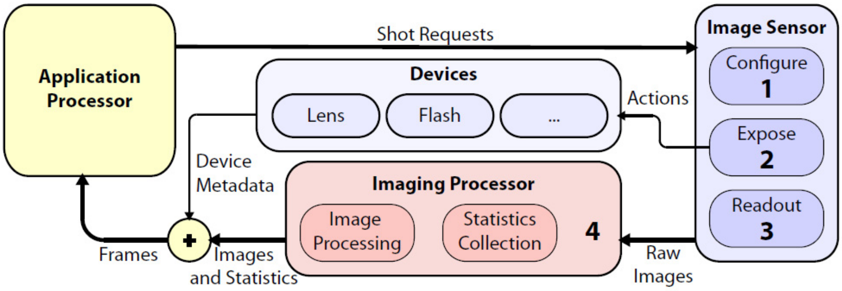


Fig. 1 The Frankencamera architecture: The application processor generates **`Shot`** requests that are sent to the **`Sensor`**. When the image sensor is exposed, any registered actions related to the **`Shot`** are executed on the **`Lens`**, **`Flash`**, or other devices. The image processor accepts the image data, computes statistics, and performs any requested image processing tasks. Finally, the image, the statistics plus tags from the devices are combined into a **`Frame`**. From Eino-Ville Talvala's dissertation [11].

deterministically only a single image. To guarantee determinism, the whole system may have to be reinitialized and the image streaming restarted, which creates latency especially if several images need to be captured. FCam takes a different approach to state handling by associating the state not with the camera, but with an individual image request called **`Shot`**. Now the state travels through the pipeline and allows the system to proceed at a higher speed even when different images have different parameters and state.

This innovation allows the following key capabilities to be applied at higher speeds:

- Burst control (per-frame parameter control for a collection of images),
- Synchronization of flash, lenses, etc.,
- Specialized algorithms for auto focus, auto exposure, and auto white balance.

In the following sections we describe in more detail how these features can be used via the FCam API, as well as some of the applications they enable.

2 Capture Control

A salient feature of the FCam API is that the camera does not have any global state. Instead, the **`Sensor`** object receives capture requests that contain the state for the request, and turns these requests into image data, metadata, and actions, as shown in Figure 2. In this section we discuss the **`Shot`**, i.e. the request, and the **`Frame`**, the data container.

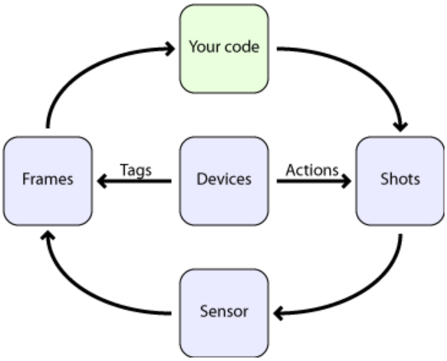


Fig. 2 The typical request generation and processing cycle. User code configures Shots, which control the Sensor, which again fills Frames with image data and metadata, which again are delivered back to user code. Devices can associate Actions with Shots, the Actions are executed at a time specified with respect to the image exposure. The Devices can also tag frames with additional metadata.

2.1 Shots and Frames

A capture request takes the form of a **`Shot`** class instance. A **`Shot`** defines the desired image sensor parameters such as exposure time, frame time, and analog gain. In addition, a **`Shot`** also has properties to configure the Image Signal Processor (ISP) to process the image with a given color temperature for white-balance and to configure the generation of statistics such as a sharpness map and image histogram. Finally, a **`Shot`** also takes an **`Image`** object that defines the image resolution and format. Figure 3 illustrates a piece of FCam API code that performs a capture request.

The call into the **`Sensor`** to capture a **`Shot`** is non-blocking, so we can keep doing more work while the image is being captured, and even issue additional requests. For each **`Shot`** that we pass down we can expect a corresponding **`Frame`** to be returned. That **`Frame`** object contains the image data and additional information that describes both the *requested* and the *actual* parameters that were used for the capture, plus the statistics that we have requested. The actual parameters may differ from the requested ones when the **`Shot`** includes a request that cannot be completely satisfied as specified, such as too short or long an exposure time. As we will see in Section 3, a **`Frame`** can also contain additional metadata about devices such as the state of the flash and the position of the lens. To retrieve a **`Frame`** we call `Sensor::getFrame()`, which is a blocking call that will only return once the **`Frame`** is ready.

```
FCam::Tegra::Sensor sensor;
FCam::Tegra::Shot   shot;
FCam::Tegra::Frame  frame;

shot.gain         = 1.0f;  // Unit gain
shot.exposure     = 25000; // Exposure time in microseconds
shot.whiteBalance = 6500;  // Color temperature
// Image size and format
shot.image = FCam::Image(2592, 1944, FCam::YUV240p);

// Enable the histogram generation and sharpness computation
shot.histogram.enabled = true;
shot.sharpness.enabled = true;

// Send the request to the Sensor
sensor.capture(shot);

// Wait for the Frame
frame = sensor.getFrame();
```

Fig. 3 A typical capture request.

2.2 Image Bursts

Many computational photography applications need to capture several images taken in quick succession, and often with slightly different parameters. We call such a set of images a burst, and represent it in the API as a vector of **`Shot`** instances. The FCam API runtime will do its best to capture the burst with the minimum latency.

The prototypical application of image bursts is high-dynamic-range (HDR) imaging. A scene we are interested in may contain a much larger dynamic range than we can capture with a single image. That is, if we set the exposure parameters so that details in bright areas can be seen, the dark areas remain too dark to resolve any details, and vice versa. By combining information from images taken with different exposure times we can generate a new image that preserves details both in the dark and bright regions.

In Figure 4 we show sample code to generate a burst of varying exposure times with the FCam API using a vector of **`Shot`** instances, and in Figure 5 we show the results of a varying exposure burst which we combined into a single image using exposure fusion [8].

3 External Devices and Synchronization

A camera subsystem consists of the imaging sensor plus other devices, such as the flash and the lens focusing motor. It is important that the image sensor and the devices are synchronized properly to achieve the highest throughput and correct results. The FCam API provides a mechanism to set the behavior of these devices per **`Shot`**, as we will describe below.

```
FCam::Tegra::Sensor sensor;

std::vector<FCam::Tegra::Shot>  burst(3);
std::vector<FCam::Tegra::Frame> frames(3);

// Prepare shot with color temperature 6500K,
// unity gain and 10,000 microseconds exposure
burst[0].gain         = 1.0f;
burst[0].whiteBalance = 6500;
burst[0].exposure     = 10000;

// Copy the shot parameters
burst[1] = burst[2] = burst[0];

// Change the exposure time for the other shots
burst[1].exposure = 20000;
burst[2].exposure =  5000;

// Reserve one storage image for each frame
burst[0].image = FCam::Image(2592, 1944, FCam::YUV420p);
burst[1].image = FCam::Image(2592, 1944, FCam::YUV420p);
burst[2].image = FCam::Image(2592, 1944, FCam::YUV420p);

// Send the request to the Sensor
sensor.capture(burst);

// Read back the Frames as they are produced
frame[0] = sensor.getFrame();
frame[1] = sensor.getFrame();
frame[2] = sensor.getFrame();
```

Fig. 4 Example code that produces a burst capture of 3 consecutive frames while varying the exposure time.

3.1 Devices and Actions

For each external device that needs to be synchronized with the exposure, there is a corresponding proxy class in the FCam implementation. We represent such devices under a class called **Device**, and its behavior can be either programmed to take effect immediately, as the exposure of a given **Shot** starts, or at some later time. The behavior is controlled using another class called **Action**. A basic **Action** has a time field that defines the execution time relative to the beginning of the **Shot** exposure. When an **Action** is added to a **Shot**, the FCam runtime will take all the necessary steps so it is ready to execute it, synchronized with the **Shot** exposure. This synchronization is possible when the underlying camera subsystem has predictable latencies.

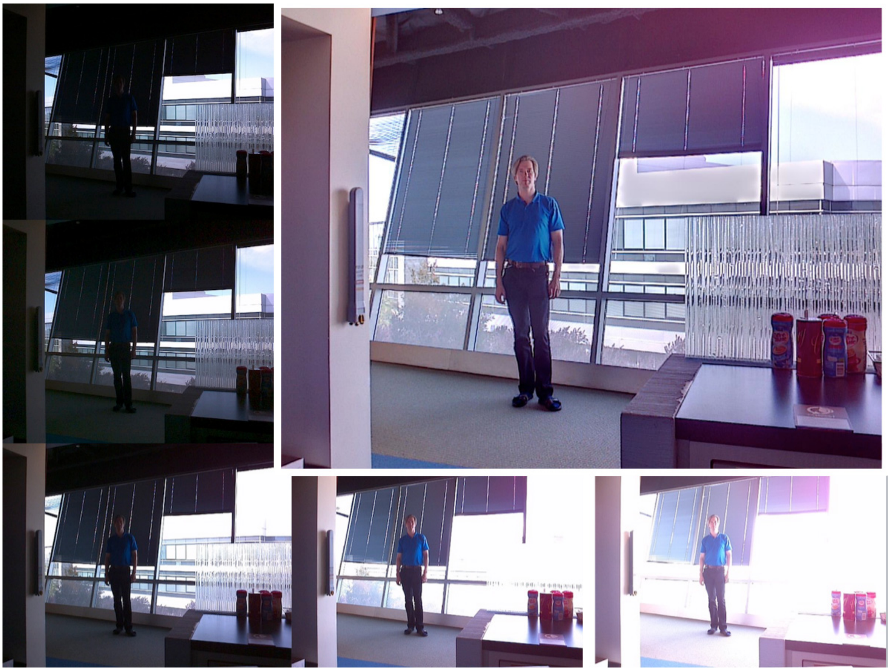


Fig. 5 A burst of five image taken with different exposures (left and bottom) are fused into a single image that shows details both in the bright and dark areas better than in any of the input images.

3.1.1 Flash

As a first example, we will take a look at the **`Flash`** device and its **`FireAction`**. The **`Flash`** class represents the camera flash and has methods to query its properties, such as maximum and minimum supported duration and brightness. It also has a method called `fire()` that sends the commands to the hardware device to turn the flash on. The latency between the call to `fire()` and the actual flash being fired can be queried with the method `fireLatency()`.

In addition, to synchronize the flash with a given **`Shot`**, the **`Flash`** class provides a predefined **`FireAction`**, which specifies the starting time plus the duration and brightness for the flash. By setting the brightness and the duration of the **`FireAction`** we can trigger the flash. Figure 6 puts these concepts together and shows an example of flash/no-flash photography, in which two different requests are sent to the **`Sensor`**: a **`Shot`** with a **`FireFlash`** action followed by a shot without flash.

```
FCam::Tegra::Sensor sensor;
FCam::Tegra::Flash  flash;

sensor.attach(&flash);

std::vector<FCam::Tegra::Shot>  shots(2);
std::vector<FCam::Tegra::Frame> frames(2);

// Prepare the shots
shots[0].gain         = 1.0f;
shots[0].whiteBalance = 6500;
shots[0].exposure     = 30000;
shots[1] = shots[0];

// Add flash action to fire the flash for the duration of
// the entire frame and with maximum brightness
FCam::Flash::FireAction fire(&flash);

fire.duration   = shots[0].frameTime;
fire.time       = 0;
fire.brightness = flash.maxBrightness();

shots[0].addAction(fire);

// Reserve one storage image for each frame
shots[0].image = FCam::Image(2592, 1944, FCam::YUV420p);
shots[1].image = FCam::Image(2592, 1944, FCam::YUV420p);

// Send the request to the Sensor
sensor.capture(burst);

// Read back the frames as they are produced
frame[0] = sensor.getFrame();
frame[1] = sensor.getFrame();
```

Fig. 6 Example code that produces a flash/no-flash image pair.

3.1.2 Lens

Another device that is readily available in FCam is the **`Lens`**. The **`Lens`** device has query methods to retrieve the lens focal range, aperture range, and zoom range; and state setting methods to set the lens to a particular focus position, zoom focal length, or aperture. Of course, not all lenses will support all settings and the query functions return a single-valued range for those properties that are fixed. For functions that affect the focus of the lens, the unit that is used is called a diopter; lens position and lens speed are given in diopters and diopters/sec, respectively. Diopters can be obtained from $100\text{cm}/f$, where f is the focusing distance, with zero corresponding to infinity, and 20 corresponding to a focusing distance of 5cm. This unit is particularly suitable for working with lens positions because lens movement is linear in diopters, and depth of field is a fixed number in diopters regardless of the depth you are focused at.

The **Lens** device provides three different kinds of **Action** classes: **FocusAction**, **ApertureAction**, and **ZoomAction** to control focus, aperture, and focal length, respectively. In the Tegra implementation of FCam, there is also a **FocusStepping** action that allows to cover a focal range in a given number of steps and is useful for covering the focal range during auto focus.

Figure 7 contains a code snippet that shows how to move the lens to the nearest focus position and capture a shot.

```
FCam::Tegra::Sensor sensor;
FCam::Tegra::Lens   lens;

sensor.attach(&lens);

FCam::Tegra::Shot  shot;
FCam::Tegra::Frame frame;

// Setup the shot parameters
shotgain         = 1.0f;
shotwhiteBalance = 6500;
shot.exposure    = 30000;
shot.image = FCam::Image(2592, 1944, FCam::YUV420p);

// Move the lens to the closest focus position
lens.setFocus(lens.nearFocus(), lens.maxFocusSpeed());
while(lens.focusChanging()){;}

// Send the request to the Sensor
sensor.capture(shot);

// Get the frame
frame = sensor.getFrame();
```

Fig. 7 Capture a shot at near focus.

3.2 Tags

When using the FCam API there is no need to keep track of the state for each **Device**. Instead, each **Frame** that is returned by the **Sensor** is tagged with the parameter‘s of all devices that had been attached to the **Sensor**. Each **Device** that is in use has to be attached to the **Sensor** by calling `Sensor::attach()`

before triggering the first capture. This allows the **Sensor** to know which devices to notify that a Frame capture has been completed.

Tags are parameter values that are added to a Frame instance. Each device class has an inner class to retrieve its corresponding tags from a **Frame**. Following our **Flash** and **Lens** examples, the **Flash** provides **Flash::Tags** and the **Lens** provides **Lens::Tags**. The **Flash** tags indicate the flash firing time relative to the start of the exposure, its duration, its brightness, and its peak time. If the flash was not fired, the tags will show a brightness of zero. Similarly, the **Lens** provides tags that indicate the initial and final focus positions, the focus speed, and the average focus setting for a **Frame**. If the lens did not move during the exposure, the three values for initial, final, and average focus position will all be the same. There are also tags for aperture and zoom settings.

It is important to stress that accurate tagging and **Frame** parameters makes a big difference in computational photography applications. Knowing the states of the camera during the exposure allows plugging this information into our algorithms or making a decision about the usefulness of the **Frame** we have just captured. For example, one might decide to discard a **Frame** if the lens moved during the exposure of the shot.

3.3 Application: Second-curtain Flash Synchronization

The richness of the API and its ability to synchronize the exposure with external devices can be exemplified with second-curtain flash synchronization. Using the F2 Frankencamera and two Canon flash units, one doing low-intensity strobing, while the other emits a second-curtain high-intensity flash at the end of the exposure, it is possible to produce the effect shown in Figure 8. A long exposure captures the path of the cards as they fly into the air, and the final bright flash freezes the cards to their final positions in the image.

4 Automating Capture Parameter Setting

Early photographers had full control of every stage of photography, and they had to make explicit selections of all the variables affecting the creation of a photo. They had to estimate the amount of light in the scene and how that should be taken into account in selection of the lenses, aperture setting, or exposure time. Some of the exposure problems could be still treated during the interactive film development and printing stages. Modern cameras make photography much easier as they have automated most of these decisions. Before the actual image is taken, the camera measures and tries some of the parameters. This process typically consists at least of these three tasks: auto exposure, auto focus, and auto white balance, also known collectively as 3A. Video is controlled continuously: the camera analyses the previous frames, and based on the analysis the exposure, focus, or white balance values are slowly and continuously modified for the following frames.

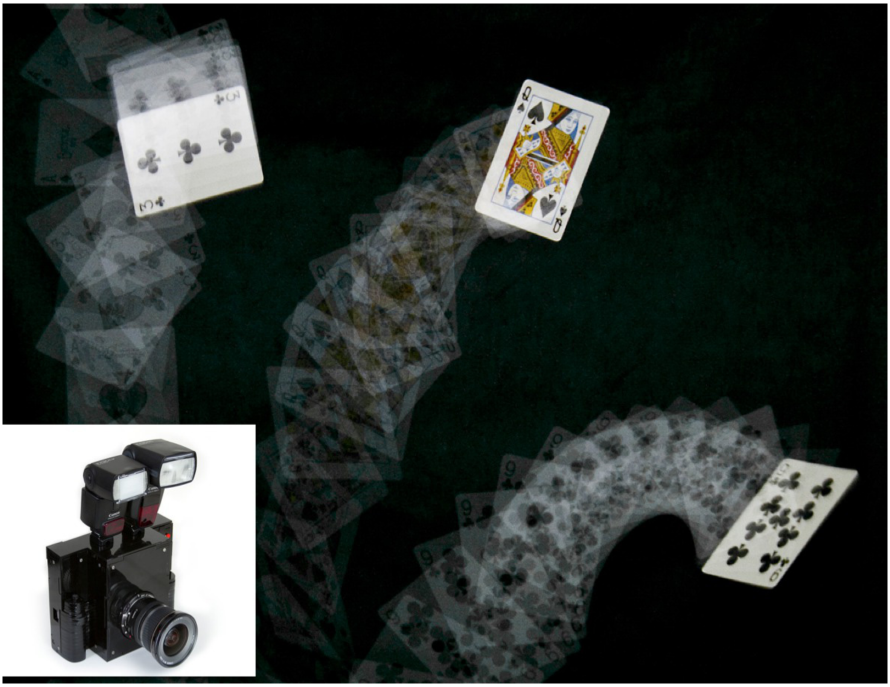


Fig. 8 Second-curtain flash: one flash strobes to illuminate the path of the flying card, while the other one freezes the motion at the end of the exposure. Image by David Jacobs.

Traditional camera control APIs completely automate these tasks and do not allow the user to modify them. Sometimes the user can override the precise camera control parameter values, but it is not possible to provide different metrics or algorithms for determining those values automatically. The default 3A produces values that in most cases provide a good image, but is optimized for the average situation, not for the current application. For example, in a security application the camera should make sure that the faces of the people remain recognizable, or the register plates of the cars can be deciphered, but it does not matter if the sky is completely saturated. FCam, on the other hand, allows you to implement your own parameter setting algorithms that are suitable for your needs.

In this section we discuss the default 3A algorithms provided by FCam, together with some advanced algorithms.

4.1 Auto Exposure

The auto exposure algorithm determines how much light should be collected to create an image so that it is not too dark and does not saturate. There are several parameters that affect the exposure, the most obvious one being the duration of the exposure. Other parameters include the amount of gain applied in the conversion of analog sensor signal to digital, and the size of the aperture in the lens system. Since the size of the aperture affects also other parameters such as depth of field, it is usually kept fixed by the auto exposure routines. In dark conditions it is better to increase the exposure time to collect more light, but on the flip side this allows both the camera and objects in the scene to move, which causes blur. By increasing the analog gain, also known as the ISO value, one can shorten the exposure time and still get a sufficient large signal, but by amplifying the signal, the noise is amplified as well, and the likelihood of saturating the light representation increases, so the limits for modifying gain are fairly narrow. The gain and exposure time are usually multiplied together, and this product is called exposure.

FCam provides a sample auto exposure function that allows the user to set two numbers for an exposure target. The first number is a percentage P, and the second number is target luminance value Y. For example, values $P = 0.995, Y = 0.9$ mean that the system tries to find an exposure value so that 99.5% of the pixels have a value that is at most 0.9 (1.0 means the pixel is saturated). These values mean metering for highlights, so that the details in bright areas remain visible. Setting $P = 0.1, Y = 0.1$ can be interpreted so that at most 10% of the pixels should have a value 0.1 or less, metering the image so that details in the shadows remain visible.

What makes choosing the perfect exposure value difficult is the inconvenient fact that the dynamic range of the sensor is quite narrow, so it is often impossible to take a single image in which details both in the dark and bright areas remain visible, as discussed earlier with HDR imaging. A typical heuristic for capturing an HDR burst is to meter for one normal image, and then choose a fixed number of images that are taken with increasing and decreasing exposure settings. For example, if the auto exposure routine gives an exposure duration of 20ms, the bracketing heuristic could choose durations of 1.25, 5, 20, 80, and 320 milliseconds for the five shots in the burst. A better heuristic would find first the shortest exposure so that no pixel is (or only a few pixels are) saturated, and then increase the exposure times as long as there are pixels that remain very dark. However, neither heuristic adapts well to the actual distribution of the light in the scene.

Gallo *et al.*[7] developed a metering method for HDR imaging that attempts to take the smallest number of images while still accurately capturing the scene data. An advantage of taking only a few shots is that the capture takes shorter amount of time, leaving the scene objects less time to move around. Also, a burst containing fewer images can be processed faster, and there is a smaller chance to create spurious artifacts, especially when objects are moving.

Figure 9 illustrates the method of Gallo *et al.*[7]. The red curve on the left shows the light histogram for the office scene shown in the middle. The areas marked by red rectangles in the office scene are shown enlarged on the right. Gallo's method selects only three images with exposure times of 0.03, 1.3, and 20 seconds, producing the detail HDR images on the first column on the right, while an auto-bracketing method with five images produces more noisy result, which is illustrated in the second column from the right. The trick is that Gallo first estimates the whole

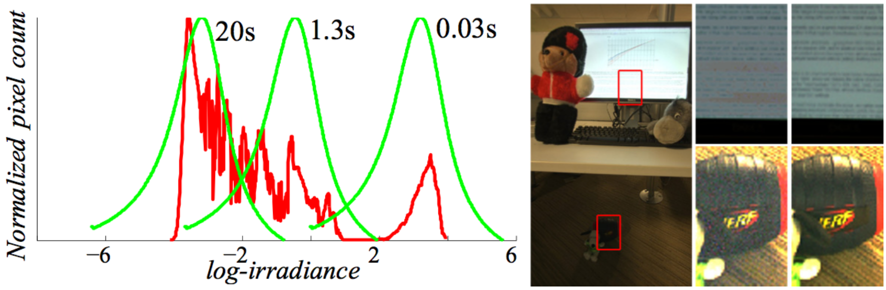


Fig. 9 HDR metering: luminance histogram (red) with three chosen exposures (green); office scene; noisy HDR with 5 bracketed images; better HDR with 3 better-metered images. Image by Orazio Gallo.

luminance histogram, and then places the images such that they capture in their most sensitive region the parts of the histogram that are strongly represented in the scene. The method was implemented on an NVIDIA Tegra 3 developer board using FCam on Android.

4.2 Auto Focus

Another important parameter to choose is the focus distance. Sometimes it is desirable that as much of the image as possible remains sharp, but at other times only the object at the center of attention needs to be sharp, while the unimportant background should be somewhat blurred. Most automated focus routines try to maximize the sharpness either over the whole image or over its center.

FCam provides a sample auto focus implementation. It uses a simple sharpness measure evaluated either over the whole image or a user-specified window. The sharpness measure is a sum of absolute differences of the intensity of neighboring pixels. The idea is that, if the image is blurred, the neighboring pixels have quite similar intensity values, while textured surfaces that are in focus have pixels with higher variance in their intensity values. The sample implementation simply sweeps the lens and gathers sharpness statistics, estimating the single lens position that maximizes the sharpness within the evaluation window.

Vaquero *et al.*[13] implemented a method that computes an all-in-focus image. If some of the scene objects are very close and others are far, it may be impossible to take a single image in which everything is in focus. However, if one captures several images focused at different depths, one can then afterwards combine them by selecting pixels separately from different images based on their sharpness estimate, as illustrated in Figure 10.

The benefits of auto focusing for focal stacks are similar to metering for HDR stacks. Even though one could simply take an image focused at every depth, it is

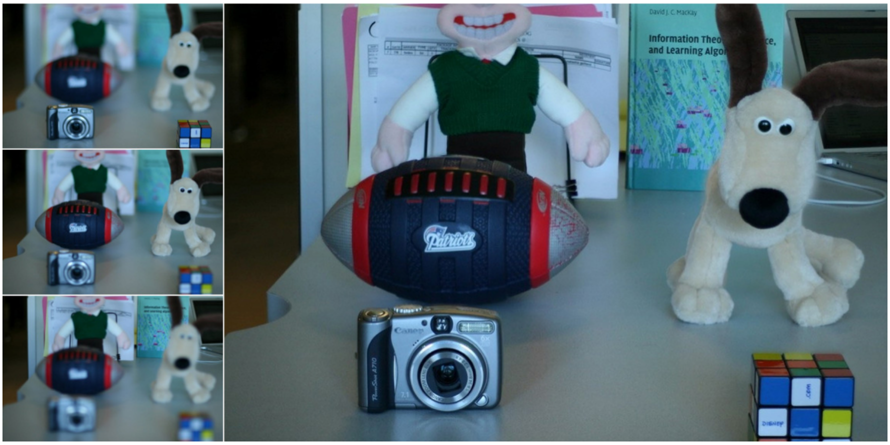


Fig. 10 Three images (left) were taken, focused in objects in foreground (top), middle ground (middle), and background (bottom), and combined into a single image that is sharp everywhere (right). Image by Daniel Vaquero.

better to only take those images that actually bring new information, yielding a faster capture time, faster processing time, and fewer chances of creating processing artifacts.

4.3 Auto White Balance

The human visual system quickly adapts to the color of ambient illumination and mostly discounts it, allowing good color perception under varying lighting conditions. This is much more difficult to do for a camera, and may result in images with a strong color tint, and appear both unnatural and very different from how a humans perceive the same situation. One reason for this is that the camera has a much smaller field of view than people and thus cannot as accurately estimate the color of the ambient illumination. Another reason is that the mechanisms of color constancy in human perception are still not completely understood.

The FCam sample auto white balance implementation uses a simple heuristic called the gray world assumption. The idea is that many scenes have many objects that do not have a color that differs from some shade of gray, including white, black, and anything in between. FCam further simplifies the assumption so that it attempts to balance the amount of blue and red light, as they correspond psycho-physically to cold and warm colors, while green does not have as strong perceptual effect. The sensor is pre-calibrated with two color correction matrices, one to correct a scene with blueish tint and another to correct a scene with reddish tint. The relative amounts of blue and red light in the captured image determine how these two color correction matrices are interpolated before they are applied to correct the image colors.

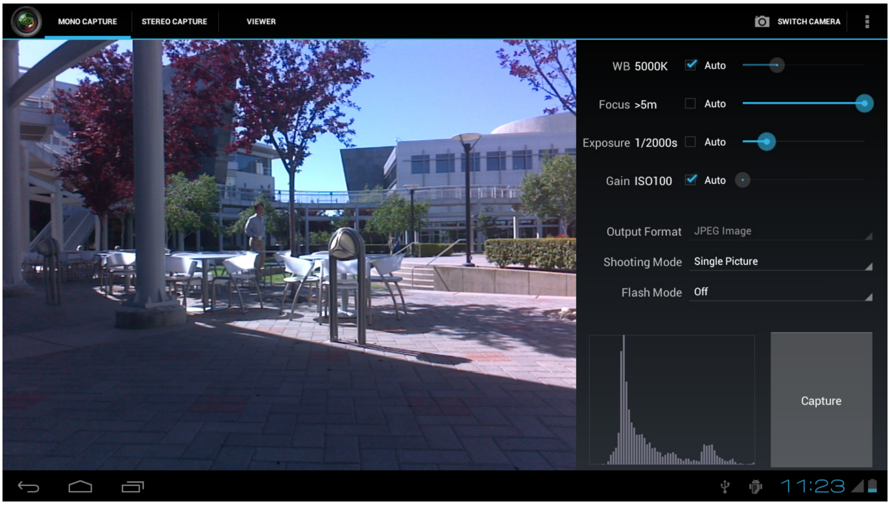
4.4 Building your Own Camera Application

It is easy to write your own custom camera application using FCam. A basic camera application streams frames continuously, and for each captured **`Frame`** the user is provided with statistics that allow 3A to be performed, either by using the sample implementation or the user's own, more sophisticated heuristics. The streaming **`Shot`** parameters are updated and the **`Frame`** displayed to the user. These functionalities could be implemented using the simple example code shown in Figure 11.

While the FCam API can take care of the camera control aspect of the camera application implementation, the display and UI are system-dependent. On the N900 platform the Qt framework is used for the UI and display. On the Tegra 3 platform the Android framework is used instead. An Android UI is built as a Java component that sets up the Android views. The FCam API is a native API, and therefore requires using the Java Native Interface (JNI) for the communication between the Java Virtual Machine and the native library that contains the FCam code. The UI generates events that are passed to the native camera implementation that runs on

```
FCam::Tegra::Sensor    sensor;
FCam::Tegra::Shot      shot;
FCam::Tegra::Frame     frame;
FCam::Tegra::Lens      lens;
FCam::Tegra::AutoFocus autoFocus(&lens);

// Viewfinder resolution
shot.image = FCam::Image(1280, 720, FCam::YUV240p);

// Enable the histogram generation and sharpness computation
shot.histogram.enabled = true;
shot.sharpness.enabled = true;

// Attach the lens device to sensor
sensor.attach(&lens);

// Send a streaming request to the sensor
sensor.stream(shot);

while(1) {
    // Wait for the frame
    frame = sensor.getFrame();

    // Display the frame
    display(frame);

    // Do 3A
    FCam::autoExpose(&shot, frame);
    FCam::autoWhiteBalance(&shot, frame);
    FCam::autoFocus(frame, &shot);

    // run the shot with the updated parameters
    sensor.stream(shot);
}
```

Fig. 11 A basic camera implementation.



WB 5000K ☑ Auto
Focus >5m □ Auto
Exposure 1/2000s □ Auto
Gain ISO100 ☑ Auto
Output Format JPEG Image
Shooting Mode Single Picture
Flash Mode Off

Fig. 12 FCameraPro: A custom camera application built using FCam.

FCam. We have built a sample application called FCameraPro depicted in Figure 12. This application can be modified without much effort to try out new algorithms and techniques.

5 FCam Platforms

The Frankencamera architecture and FCam API began as a joint research project between Nokia Research Center and Stanford University Graphics Lab. Nokia was at the time working on a family of smartphones that ran Linux (the version was earlier called Maemo, later renamed to Meego) and that used the TI OMAP 3 processor. Mistral had also made OMAP 3 boards available even for hobbyists, and the project chose OMAP 3 and Linux as the common HW and SW platform. This led in parallel to two related implementations: the Nokia N900 used the standard hardware that the phone shipped with and allowed much more flexible use of that hardware than the camera stack that came with the phone, and the Stanford Frankencamera V2 (F2) used the Mistral OMAP 3 board together with a Birger lens controller that accepts Canon EOS lenses. The N900 allowed a relatively cheap mass-marketed FCam solution, while F2 provided an extensible research platform that allowed experimentation with different optics and hardware choices.

The FCam in N900 was not “officially” supported by the Nokia product program, it was a “community effort” maintained by the Nokia Research Center. However, the follow-up product N9 now provides official support for the FCam API. Currently, most active FCam development happens on NVIDIA’s Tegra-3-powered development tablets running Android.

5.1 The FCam Runtime

At the core of any of the FCam implementations is the FCam runtime. The runtime is made of a set of components that take FCam API calls, configure the camera subsystem for execution, and return captured frames. In Figure 13 we show the block-diagram of the FCam implementation running on the NVIDIA Tegra 3 prototype board. The runtime is made of the FCam API objects and runs a number of threads. The first is a **`Setter`** thread that manages the incoming requests queue, programs the hardware, and computes the absolute time at which actions should be executed. Secondly, an **`Action`** thread manages the action queue; it wakes up for each scheduled action and launches its execution. It is important that the work an **`Action`** launches on execution is bounded, otherwise the thread could miss the execution deadline for the following **`Action`**. If necessary, an **`Action`** could spawn a new thread to achieve completion. Finally, a **`Handler`** thread receives callbacks from the camera driver with image data and metadata, assembles these data into a **`Frame`** instance, and delivers it to the **`Sensor`** output queue. On the left side of Figure 13 is the camera hardware, the NVIDIA Tegra 3 SoC (system-on-chip), the Linux kernel drivers and the NVIDIA camera driver. The NVIDIA camera driver takes parameter requests and assembles commands to configure the ISP or calls the corresponding kernel device driver, according to the request.

Having given the basic components of an FCam implementation, we now enumerate the steps necessary to convert an application request into image data:

1. The application makes a capture request into the **`Sensor`** passing a **`Shot`**.
2. The **`Sensor`** takes the **`Shot`** and places it in the request queue that it shares with the **`Setter`** thread.
3. At the next indication that the camera subsystem is ready to be configured, the **`Setter`** thread takes the first element of the request queue. For each **`Action`** in the **`Shot`** it computes its execution time and schedules it in the action priority queue. It also sends commands to the NVIDIA camera driver to configure the image sensor and ISP with the requested parameters.
4. The **`Action`** thread wakes up and executes any **`Action`** that is synchronized with the current **`Shot`**. Each **`Action`** will trigger a command into the NVIDIA camera driver.
5. The NVIDIA camera driver abstracts the underlying camera hardware. It receives commands and programs the corresponding kernel device drivers.
6. When the image data and metadata are ready, the NVIDIA camera driver delivers them to the **`Handler`**.
7. The **`Handler`** assembles a **`Frame`** and puts it into the frame output queue.
8. When the frame output queue receives a new **`Frame`** the **`Sensor`** delivers it to the application.

To further expand on the pipeline aspects of the FCam runtime, we now turn our attention to the timeline of events that are needed for proper configuration of the image sensor. An image sensor might require state changes to be precisely timed. For example, an image sensor could have a dual set of registers, and the system

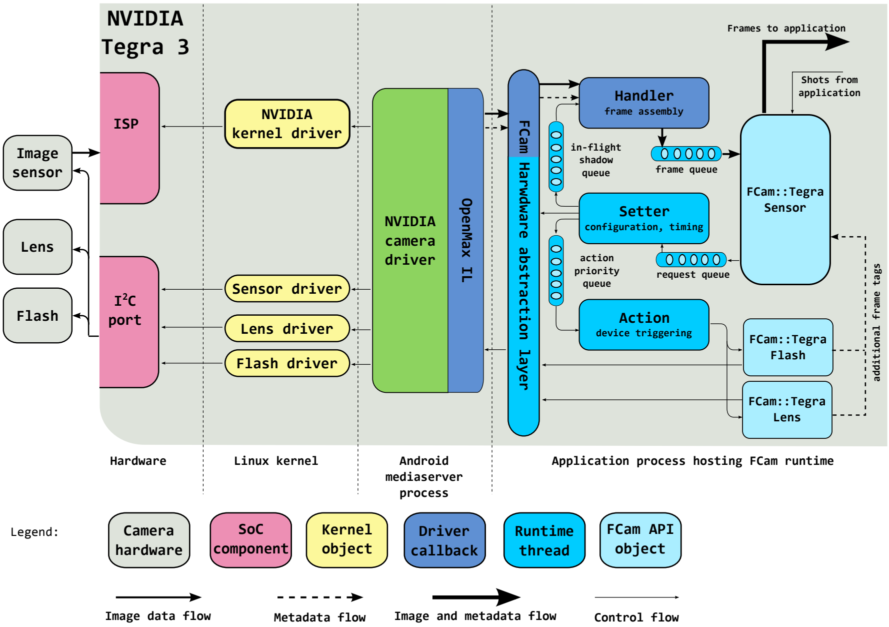


Fig. 13 A block-diagram of the FCam implementation on the Tegra 3 Prototype. Adapted from the original block-diagram by Eino-Ville Talvala [11].

writes into one of them the parameters that will become active at the frame reset. Or it could be the case that the change to a particular register is applied immediately.

The image sensor in the Nokia N900 is a Toshiba ET8EK8 rolling shutter CMOS. The sensor requires that the exposure time and frame duration be programmed one frame ahead. The sensor emits a vertical synchronization (VSync) interrupt that is used to synchronize the FCam runtime. At the VSync interrupt the FCam runtime sets up the exposure time and frame duration for the following frame and the sensor gain for the current one, as shown in Figure 14. A similar timeline is implemented on the Tegra 3 Prototype running the Omnivision 5650 CMOS sensor.

5.2 *Porting FCam*

As we have seen from the implementation details, porting the FCam API to a new platform requires deep knowledge of the underlying OS and camera stack. It is also necessary that some of the system drivers be flexible enough to accommodate all the parameters that the FCam runtime needs to set. Finally, it is important that consistent latencies can be computed in order to schedule actions correctly.

The N900 implementation required the modification of the Video For Linux 2 (V4L2) kernel driver. Once the changes were done, the FCam runtime was implemented calling the device drivers directly. However, not all platforms allow for user

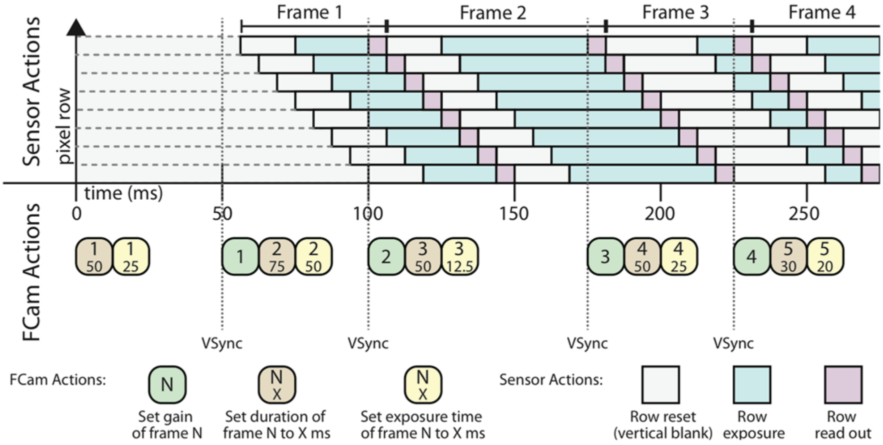


Fig. 14 Timeline of events for configuration of the image sensor. From Eino-Ville Talvala's dissertation [11].

applications to call functions running in hardware device drivers. Porting the FCam API to the Tegra 3 platform required tweaks at different levels of the software stack because only system processes are allowed to access the camera drivers in Android. User applications need to connect to the Android mediaserver process to send requests to the camera hardware.

As camera APIs evolve, it is expected these will become more flexible and enable high-throughput computational photography applications.

6 Image Processing for FCam Applications

FCam is meant for camera control, not for intensive image processing. For that there are other tools and APIs. In this section we describe three: OpenCV computer vision library, OpenGL ES 2.0 graphics API, and NEON intrinsics (NEON is a SIMD-type co-processor for ARM CPUs).

6.1 OpenCV

OpenCV [3] is the de-facto standard computer vision API. It originated at Intel, and the original alpha version was released in 2000. After Intel stopped development of OpenCV, companies such as Willow Garage, Itseez, and NVIDIA have supported its development. It has over 500 algorithms for all types of computer vision and image processing tasks. It is available on most operating systems, including Windows, Linux, MacOS, and Android. Figure 15 illustrates a subset of OpenCV functionality, and shows a sample OpenCV program running on an Android smartphone.

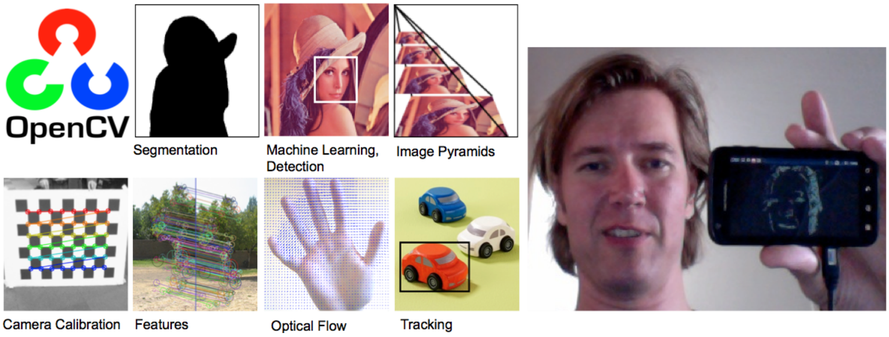


Fig. 15 OpenCV supports a large array of computer vision and image processing functions. The image on the right shows an OpenCV example running on an Android phone, doing a real-time edge detection on the input video stream coming from the camera.

Originally OpenCV was developed and optimized for execution on Intel CPUs, but it has now been compiled for many different hardware platforms, including the ARM processor that powers most smartphones and tablets. A relatively recent development is the addition of the GPU module, that leverages the processing power of modern CUDA-capable graphics cards on desktop and laptop computers [10]. NVIDIA is also tailoring OpenCV so that it can use the hardware capabilities on its Tegra 3 mobile processor, which includes four ARM CPU cores, each with a NEON co-processor, and GPU supporting OpenGL ES 2.0.

OpenCV is a well-documented library that makes cross-platform vision or image processing applications easy. You can develop and test the application first on a desktop computer, and once the basic logic is working, easily port the application to a mobile device for further finetuning and optimizations.

6.2 OpenGL ES 2.0

In addition to the CPU, most computers have another powerful processor, the GPU (Graphics Processing Unit). The first generation of mobile graphics processors supported OpenGL ES 1.0 and 1.1, which had the traditional fixed-function graphics pipeline that makes the use of the GPU for anything other than traditional computer graphics cumbersome and inefficient. OpenGL ES 2.0 [9] increased the flexibility considerably by introducing segments called vertex and fragment shader, where the programmer can provide a compilable program. In particular, the fragment shader, which is run for each pixel, is a useful tool for image processing. The typical sequence is to upload the input image into a texture map, map the texture into a pair of triangles that cover as many pixels as the size of the output image, perform the image processing in the fragment shader, and finally read back the processed image to your own program.

6.3 NEON Intrinsics

Most mobile devices such as smartphones and tablets use ARM CPUs, and most high-end mobile devices have also a co-processor called NEON [2]. NEON provides SIMD (Single Instruction, Multiple Data) architecture extension, allowing one instruction to operate on multiple data items in parallel. NEON extensions are particularly useful when you have to operate on several pixels in parallel, and can provide up to 10 times speed increase on some image processing algorithms. The NEON instructions can be accessed via C intrinsics, which provide similar functionality to inline assembly, and some additional features such as type checking and automatic register allocation, which make their use easier than inline assembly. The programmer needs to map the data to special NEON datatypes, then call the intrinsics that actually operate on the data, and finally map the processed data back to regular C data structures.

6.4 How Should you Choose Which Solution to Use?

Each of the cited options for performing the image processing has its own limitations. If we list the choices in order of ease-of-use, OpenCV is probably the easiest to get started with. NEON is more flexible than OpenGL ES, which has some surprises such as limited floating point precision and limited storage precision for storing intermediate results. However, when one considers the speed of execution, and energy consumption, the order becomes the reverse. Pulli *et al.*[10] report measurements of several image processing algorithms implemented on ARM CPU, ARM with NEON instructions, and OpenGL ES. Use of GPU is more efficient both in time and energy than the other options, followed by NEON, and pure CPU remaining the last one. To make the developers' lives a bit easier, NVIDIA optimizes OpenCV for its Tegra mobile SoC so that the implementation internally uses multithreading (making use of up to four ARM cores), NEON intrinsics, and GPU via OpenGL ES, when it makes sense. Although the result is still not quite as optimal as if the programmer would hand-tune the whole application to these execution units, the user gets still a significant speedup compared to a naive implementation with relatively little programming effort.

7 FCam in Teaching

One of the design goals of FCam was that it should be simple to use, and this feature makes it also an excellent tool for projects in university courses on computational photography and other related topics. In fact, an inspiration for FCam was a 2008 Stanford University course on Mobile Computational Photography (taught by Marc Levoy, Andrew Adams, and Kari Pulli). The students did the course projects using standard Symbian camera APIs, and that API was too restrictive to implement really interesting computational photography projects. Two years later, in winter 2010 FCam was ready for a new version of the same course (taught by Marc Levoy, Fredo

Durand, and Jongmin Baek). The first homework for the students was to implement their own auto focus routine on a Nokia N900. That is a task in which professional engineers invest several months if not years, and would normally be too cruel a task for just getting started on programming a camera. The fact that all the students could finish the assignment in a week, and that some even delivered a better solution than the one the camera phone shipped with, shows that with good tools great things can be achieved.

After the first course, different universities have used FCam in their courses on a couple of dozens top schools in North and South America, Europe, and Asia. We next describe two representative projects from those courses.

7.1 A Borrowed Flash

During the first FCam-based course, at Stanford in 2010, students Michael Barrientos and David Keeler decided to address the problem of red eyes due to flash. If the flash is close to the camera, the light enters the eye, is colored by the blood vessels feeding the retina, and is reflected back to the camera, and the eyes appear red, as illustrated in Figure 16 left. One red eye reduction technique briefly flashes a light, tricking the pupils to contract, which reduces the red eye phenomenon significantly. Another way is to move the light source further away from the camera.

In a camera phone there is not much room to move the flash more than a few centimeters away from the camera — that is, if the flash is still to remain in the same device. This project utilized the synchronization capabilities of the FCam API to borrow the flash from another device. When the main device is ready to take a photo, it signals the other device that intent over the Bluetooth wireless connection. The students were able to synchronize the two cameras accurately enough so that the flash on the second camera went off exactly as the first camera took the image, producing the image in Figure 16 right, where they eyes are not red. This project was implemented on a Nokia N900.

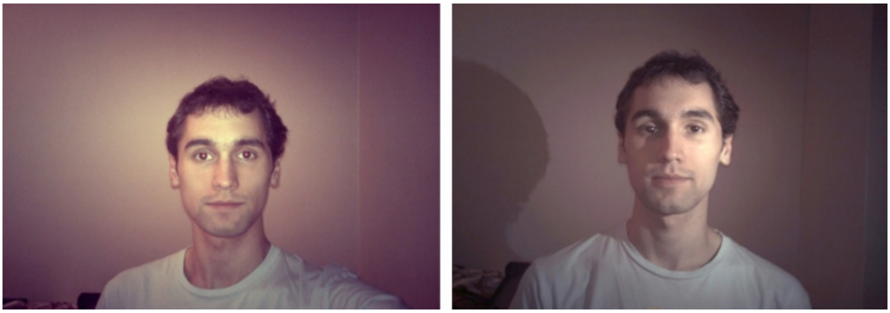
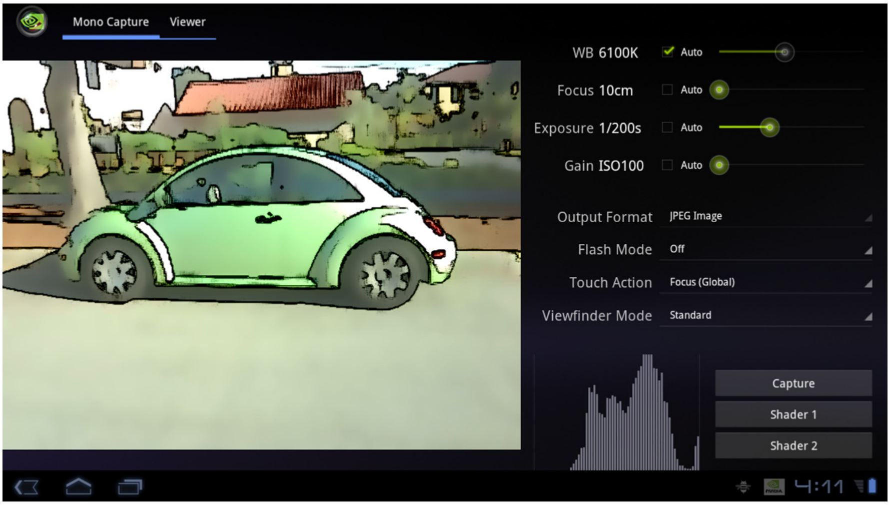


Fig. 16 A borrowed flash. The left image shows the results when the flash is too close to the camera: the eyes appear red. On the right image the Nokia N900 communicated with a second N900 so that the flash of the second camera illuminated the target while the first one took the image. Image by Michael Barrientos and David Keeler.

7.2 Non-Photorealistic Viewfinder

During the winter of 2012 version of the Stanford course (taught by Jongmin Baek, David Jacobs, and Kari Pulli), students Tony Hyun Kim and Irving Lin developed a non-photorealistic camera application. The application was developed for the Tegra 3 prototype board that the students used to implement their assignments. Output frames are post-processed using OpenGL ES before being displayed. Two shaders were written to give the non-photorealistic feeling: a bilateral filtering shader and an edge detection shader. The bilateral filter creates a flat, cartoonish rendition of the viewfinder image, and the edge detector further enhances the edges between different regions, providing more of a hand-drawn feeling. FCam was used to control flash to help separate foreground from background. The resulting application runs at an interactive frame rate on the NVIDIA Tegra 3 GPU. A screenshot is shown in Figure 17. Such an effect could be easily added to a camera application in a commercial device.



WB 6100K ☑ Auto

Focus 10cm □ Auto

Exposure 1/200s □ Auto

Gain ISO100 □ Auto

Output Format JPEG Image

Flash Mode Off

Touch Action Focus (Global)

Viewfinder Mode Standard

Fig. 17 A non-photorealistic viewfinder on NVIDIA Tegra 3 tablet. An OpenGL ES 2.0 fragment shader filters the viewfinder frames in real time to give it a live video cartoon look. Image by Tony Hyun Kim and Irving Lin.

8 Conclusions

We have presented the FCam API and its applications to mobile computational photography. As we discussed, traditional camera APIs provide little control over the camera subsystem. By treating the camera subsystem as a pipeline in which its state is associated with a request, the FCam API proves powerful for computational photography applications because:

1. It provides deterministic and well defined control over image bursts,

2. It allows for novel imaging effects by providing tight synchronization with the flash, lens, and other devices, and
3. It enables the user to build her own auto control algorithms targeting specialized applications.

In our discussions we highlighted each of these qualities by showing relevant applications. We showed how to use the per-frame control to program an HDR imaging application, how to use the synchronization capabilities to implement a borrowed flash, and how to extend the traditional metering algorithms for efficient HDR capture and all-in-focus image capture. In addition, a complete camera application can be written using the FCam API and enhanced with the image processing capabilities of today's mobile phones and tablets. The API is simple enough for university students to tackle computational photography projects. The non-photorealistic preview application, developed by students in a Computational Photography course, highlights how we can integrate camera control with image processing on the GPU to produce new stylized images.

The example applications and code snippets we have presented use a single camera; however, the number of mobile devices that have two or more cameras is rapidly increasing, opening the door for new API extensions. In [12] we have started to address multiple camera enumeration and synchronization in FCam.

Acknowledgements. The main architects behind the Frankencamera architecture and FCam API were Eino-Ville Talvala and Andrew Adams. The project leaders were Marc Levoy, Mark Horowitz, and Kari Pulli.

There have been several FCam courses at various conferences, presented by the authors, together with Marius Tico, Timo Ahonen, and Andrew Adams.

The authors would like to thank Orazio Gallo, David Pajak, Jongmin Baek, and David Jacobs for their invaluable comments and suggestions that helped improve the quality of the manuscript.

References

1. Adams, A., Talvala, E.V., Park, S.H., Jacobs, D.E., Ajdin, B., Gelfand, N., Dolson, J., Vaquero, D., Baek, J., Tico, M., Lensch, H.P.A., Matusik, W., Pulli, K., Horowitz, M., Levoy, M.: The Frankencamera: An Experimental Platform for Computational Photography. ACM Transactions on Graphics 29(3) (2010)
2. ARM: Introducing NEON Development (2009), `http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dht0002a/ch01s04s02.html`
3. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision with OpenCV Library. O'Reilly Media (2008)
4. Cossairt, O., Zhou, C., Nayar, S.K.: Diffusion Coded Photography for Extended Depth of Field. ACM Transactions on Graphics 29(4) (2010)
5. Debevec, P.E., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: Proceedings of SIGGRAPH, pp. 369–378 (1997)
6. Eisemann, E., Durand, F.: Flash photography enhancement via intrinsic relighting. ACM Transactions on Graphics 23(3), 673–678 (2004)

7. Gallo, O., Tico, M., Manduchi, R., Gelfand, N., Pulli, K.: Metering for Exposure Stacks. In: Eurographics (2012)
8. Mertens, T., Kautz, J., Van Reeth, F.: Exposure fusion. In: Proceedings of the 15th Pacific Conference on Computer Graphics and Applications (2007)
9. Munshi, A., Ginsburg, D., Shreiner, D.: OpenGL ES 2.0 Programming Guide. Addison-Wesley Professional (2008)
10. Pulli, K., Baksheev, A., Kornyakov, K., Eruhimov, V.: Realtime Computer Vision with OpenCV. ACM Queue 10(4) (2012)
11. Talvala, E.V.: The Frankencamera: building a programmable camera for computational photography. Ph.D. thesis, Stanford University (2011)
12. Troccoli, A., Pajak, D., Pulli, K.: FCam for multiple cameras. In: Proc. SPIE, vol. 8304 (2012)
13. Vaquero, D., Gelfand, N., Tico, M., Pulli, K., Turk, M.: Generalized Autofocus. In: IEEE Workshop on Applications of Computer Vision, WACV (2011)

Author Index