# Face Detection Coupling Texture, Color and Depth Data

**Loris Nanni, Alessandra Lumini, Ludovico Minto, and Pietro Zanuttigh**

**Abstract** In this chapter, we propose an ensemble of face detectors for maximizing the number of true positives found by the system. Unfortunately, combining different face detectors increases both the number of true positives and false positives. To overcome this difficulty, several methods for reducing false positives are tested and proposed. The different filtering steps are based on the characteristics of the depth map related to the subwindows of the whole image that contain the candidate faces. The most simple and easiest criteria to use, for instance, is to filter the candidate face region by considering its size in metric units.

The experimental section demonstrates that the proposed set of filtering steps greatly reduces the number of false positives without decreasing the detection rate. The proposed approach has been validated on a dataset of 549 images (each including both 2D and depth data) representing 614 upright frontal faces. The images were acquired both outdoors and indoors, with both first and second generation Kinect sensors. This was done in order to simulate a real application scenario. Moreover, for further validation and comparison with the state-of-the-art, our ensemble of face detectors is tested on the widely used BioID dataset where it obtains 100 % detection rate with an acceptable number of false positives.

A MATLAB version of the filtering steps and the dataset used in this paper will be freely available from http://www.dei.unipd.it/node/2357.

## 1 Introduction

The goal of face detection is to determine the location of faces in an image. It is one of the most studied problems in computer vision, due partly to the large number of applications requiring the detection and recognition of human beings and the availability of low-cost hardware. Face detection has also attracted a lot of

L. Nanni (✉) • L. Minto • P. Zanuttigh
DEI, University of Padova, Via Gradenigo 6, 35131 Padova, Italy
e-mail: nanni@dei.unipd.it; mintolud@dei.unipd.it; zanuttigh@dei.unipd.it

A. Lumini
DISI, Università di Bologna, Via Sacchi 3, 47521 Cesena, Italy
e-mail: alessandra.lumini@unibo.it



M. Kawulok et al. (eds.), *Advances in Face Detection and Facial Image Analysis*,
DOI 10.1007/978-3-319-25958-1_2

attention in the research community because it is a very hard problem, certainly more challenging than face localization in which a single face is assumed to be located inside an image [1]. Although human faces have generally the same appearance, several personal variations (like gender, race, individual distinctions, and facial expression) and environment conditions (like pose, illumination, and complex background) can dramatically alter the appearance of human faces. A robust face detection system must overcome all these variations and be able to perform detection in almost any lighting condition. Moreover, it must manage to do all this in real-time.

Over the past 25 years, many different face detection techniques have been proposed [2], motivated by the increasing number of real world applications requiring recognition of human beings. Indeed, face detection is a crucial first step for several applications ranging from surveillance and security systems to human–computer interface interaction, face tagging, behavioral analysis, as well as many other applications [3].

The majority of existing techniques address face detection from a monocular image or a video-centric perspective. Most algorithms are designed to detect faces using one or more camera images, without additional sensor information or context. The problem is often formulated as a two-class pattern recognition problem aimed at classifying each subwindow of the input image as either containing or not containing a face [4].

The most famous approach for frontal 2D detection is the Viola–Jones algorithm [5], which introduced the idea of performing an exhaustive search of an image using Haar-like rectangle features and then of using Adaboost and Cascade algorithm for classification. The importance of this detector, which provides high speed, can be measured by the number of approaches it has inspired, such as [6–9]. Amongst them, SURF cascades, a framework recently introduced by Intel labs [10] that adopts multi-dimensional SURF features instead of single dimensional Haar features to describe local patches, is one of the top performers. Another recent work [11] that compared many commercial face detectors (Google Picasa, Face.com acquired by Facebook, Intel Olaworks, and the Chinese start-up Face++) showed that a simple vanilla deformable part model (a general purpose object detection approach which combines the estimation of latent variables for alignment and clustering at training time and the use of multiple components and deformable parts to handle intra-class variance) was able to outperform all the other methods in face detection.

The Viola–Jones algorithm and its variants are capable of detecting faces in images in real-time, but these algorithms are definitely affected by changeable factors such as pose, illumination, facial expression, glasses, makeup, and factors related to age. In order to overcome problems related to these factors, 3D face detection methods have been proposed. These new methods take advantage of the fact that the 3D structure of the human face provides highly discriminatory information and is more insensitive to environmental conditions.

The recent introduction of several consumer depth cameras has made 3D acquisition available to the mass market. Among the various consumer depth cameras, Microsoft Kinect is the first and the most successful device. It is a depth sensing

device that couples the 2D RGB image with a depth map (RGB-D) computed using the structured light principle which can be used to determine the depth of every object in the scene. A second generation of the sensor exploiting the Time-of-flight principle has been recently introduced. The depth information is not enough precise to differentiate among different individuals, but can be useful to improve the robustness of a face detector.

Because each pixel in Kinect's depth map indicates the distance of that pixel from the sensor, this information can be used both to differentiate among different individuals at different distances and to reduce the sensitivities of the face detector to illumination, occlusions, changes in facial expression, and pose. Several recent approaches have used depth maps or other 3D information for face detection and several have been tested on the first benchmark datasets collected by Kinetic devices for 3D face recognition [12] and detection [13]. Most exiting 3D approaches use depth images combined with gray-level images to improve detection rates. In [14] Haar wavelets on 2D images are first used to detect the human face, and then face position is refined by structured light analysis. In [15] depth-comparison features are defined as pixel pairs in depth images to quickly and accurately classify body joints and parts from single depth images. In [16] a similar method for robust and accurate face detection based on square regions comparison is coupled with Viola Jones face detector. In [1, 17] depth information is used to reduce the number of false positive and improve the percentage of correct detection. In [18] biologically inspired integrated representation of texture and stereo disparity information are used to reduce the number of locations to be evaluated during the face search. In [19] the additional information obtained by the depth map improves face recognition rates. This latter method involves texture descriptors extracted both from color and depth information and classification based on random forest. Recently, 3D information is used by DeepFace [20] to perform a 3D model-based alignment that is coupled with large capacity feedforward models for effectively obtaining a high detection rate.

An improved face detection approach based on information of the 2D image and the depth obtained by Microsoft Kinect 1 and Kinect 2 is proposed in this paper.

The proposed method in this chapter is based on an ensemble of face detectors. One advantage of using an ensemble to detect faces is that it maximizes the number of true positives; a major disadvantage, however, is that it increases both the number of false positives and the computation time. The main aim of this work, an update of our previous paper [1], is to propose a set of filtering step for reducing the number of false positives while preserving the true positive rate. To achieve this goal, the following approaches:

- SIZE: the size of the candidate face region is calculated according to the depth data, removing faces that are the too small or too large.
- STD: images of flat objects (e.g. candidate face found in a wall) or uneven objects (e.g. candidate face found in the leaves of a tree) are removed using the depth map and a segmentation approach based on the depth map.

- SEG: a segmentation step based on the depth map is used to segment a candidate image into homogenous regions; images whose main region is smaller than a threshold (with respect to the candidate image dimension) are then filtered out.
- ELL: using the segmented depth-image, an ellipse fitting approach is employed to evaluate whether the larger region can be modeled as an ellipse. The fitting cost is evaluated to decide whether or not to remove the candidate face.
- EYE: an eye detection step is used to find eyes in the candidate image and reject regions having a very low eye detection score.
- SEC: a neighborhood of the candidate face region is considered in the depth map. Neighbor pixels whose depth value is close to the face mean depth are assigned to a number of radial sectors. The lower sectors should contain a higher number of pixels.

The proposed approach has been validated on a dataset composed by 549 samples (containing 614 upright frontal faces) that include both 2D and depth images. The experimental results prove that the proposed filtering steps greatly reduce the number of false positives without decreasing the detection rate. For a further validation, the ensemble of face detectors is also tested on the widely used BioID dataset [21], where it obtains 100 % detection rate with an acceptable number of false positives. We want to stress that our approach outperforms the approach proposed in [22], which works better than such powerful commercial face detectors as Google Picasa, Face.com—Facebook, Intel Olaworks, and the Chinese start-up Face++.

The arrangement of this chapter is as follows. In Sect. 2 the whole detection approach is described, starting from the base detectors and moving on to explain all the filtering steps. In Sect. 3 the experiments on the above mentioned benchmark datasets are presented, including a description of the self-collected datasets, the definition of the testing protocol, and a discussion of the experimental results. Finally, in Sect. 4 the chapter is concluded and some future research directions are presented. The MATLAB code developed for this chapter and the datasets will be freely available.

## 2 The Proposed Approach

The proposed method is based on an ensemble of several well-known face detectors. As a first step we perform face detection on the color images using a low acceptance threshold in order to have high recall. This results in low precision since many false positive occur in the search. As a second step the depth map is aligned to the color image, and both are used to filter out false positives by means of several criteria designed to remove non-face images from the final list. In order to better handle non-upright faces, the input color images are also rotated $\{20°, -20°\}$ before detection. In the experiments the use of rotated images for adding poses is denoted by a *.

We perform experiments on the fusion of four face detectors:

- ViolaJones(VJ) [5], which is probably the most diffused face detector due to its simplicity and very fast classification time. VJ uses simple image descriptors, based on Haar wavelets extracted in a low computational time from the integral image. Classification is performed using an ensemble of AdaBoost classifiers for selecting a small number of relevant descriptors, along with a cascade combination of weak learners for classification. This approach requires a long training time but it is very fast for testing. The precision of VJ strictly relies on the threshold σ used to classify a face an input subwindow.
- SN [23] is a face detector[1] based on local descriptors and Successive Mean Quantization Transform (SMQT) features that is applied to a Split up sparse Network of Winnows (SN) classifier. The face detector extracts SMQT features by a moving a patch of $32 \times 32$ pixels that is repeatedly downscaled and resized in order to find faces of different sizes. SMQT is a transform for automatic enhancement of gray-level images that reveals the structure of the data and removes properties such as gain and bias. As a result SMQT features overcome most of the illumination and noise problems. The detection task is performed by a Split up sparse Network of Winnows as the classifier, which is a sparse network of linear units over a feature space that can be used to create lookup-tables. SN precision in face detection can be adjusted by a sensitivity parameter σ that can be tuned to obtain low to high sensitivity values. In the original implementation $\sigma_{min} = 1$ and $\sigma_{max} = 10$.
- FL [22] is a method that combines an approach for face detection that is a modification of the standard Viola–Jones detection framework with a module for the localization of salient facial landmark points. The basic idea of this approach is to scan the image with a cascade of binary classifiers (a multi-scale sequence of regression tree-based estimators) at all reasonable positions and scales. An image region is classified as containing a face if it successfully passes all the classifiers. Then a similar ensemble is used to infer the position of each facial landmark point within a given face region. Each binary classifier consists of an ensemble of decision trees with pixel intensity comparisons as binary tests in their internal nodes. The learning process consists of a greedy regression tree construction procedure and a boosting algorithm. The reported results show performance improvement with respect to several recent commercial approaches.
- RF [24] is a face detector based on face fitting, which is the problem of modeling a face shape by inferring a set of parameters that control a facial deformable model. The method, named Discriminative Response Map Fitting (DRMF), is a novel discriminative regression approach for the Constrained Local Models (CLMs) framework which shows impressive performance in the generic face

[1] http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId = 13701&objectType = FILE.

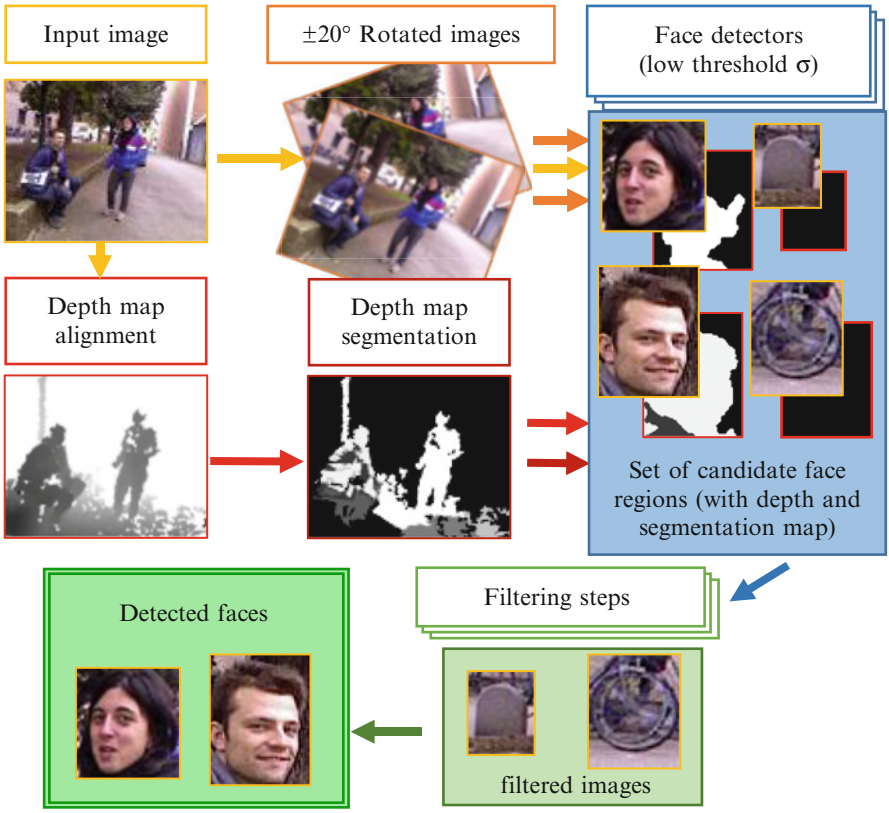


**Fig. 1** Outline of our complete system

fitting scenario. RF precision can be adjusted by a sensitivity parameter σ, which can be tuned to obtain a lower or a higher sensitivity value.

In Fig. 1 a schema of our complete system is outlined. In the first step one or more face detectors (the final configuration is a result of the experimental section) are employed for an "imprecise" detection using a low acceptance threshold, then in the second step all the candidate face regions are filtered out according to several criteria (detailed below in the following subsections) that take advantage of the presence of the depth map.

The second step exploits the information contained in the depth data to improve face detection. First calibration between color and depth data is computed according to the method proposed in [25]: the positions of the depth samples in the 3D space are first computed using the intrinsic parameters of the depth camera and then reprojected in the 2D space using both the color camera intrinsic parameters and the extrinsic ones between the two cameras. Then a color and a depth value are associated with each sample (to speed-up the approach, this operation can be

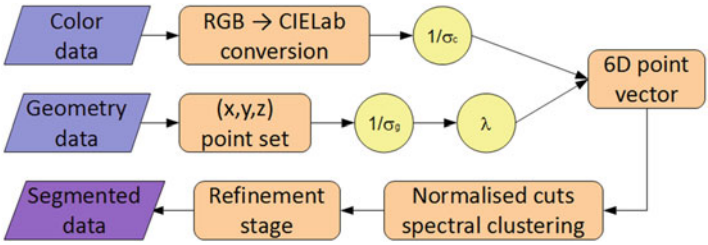


**Fig. 2** Architecture of the proposed segmentation scheme

performed only for the regions containing a candidate face). Finally, filtering is applied in order to remove false positives from the set of candidate faces. In the next subsections, the depth map alignment and segmentation approach and all the filtering rules used in this work are detailed. In Fig. 4 some candidate images properly filtered out by the different filtering rules are shown.

## *2.1 Depth Map Alignment and Segmentation*

The color image and the depth map are jointly segmented using the approach shown in Fig. 2. The employed approach associates to each sample a multi-dimensional vector and then clusters the set of vectors associated to the image using the Mean Shift algorithm [26] following an approach similar to [27].

As shown in Fig. 2 the procedure has two main stages: first a six-dimensional representation of the points in the scene is built from the geometry and color data and then second the obtained point set is segmented using Mean Shift clustering.

Each sample in the acquired depth map correspond to a 3D point of the scene $p_i$, $i = 1, \ldots, N$. After the joint calibration of the depth and color cameras, it is possible to reproject the depth samples over the corresponding pixels in the color image and to associate to each point the spatial coordinates *x*, *y,* and *z* of $p_i$ and its R, G, and B color components. Notice that these two representations lie in completely different spaces and cannot be directly compared.

In order to obtain multi-dimensional vectors suited for the clustering algorithm, the various components need to be comparable. All color values are converted to the CIELAB perceptually uniform space. This provides a perceptual significance to the Euclidean distance between the color vectors that will be used in the clustering algorithm. We can denote the color information of each scene point in the CIELAB space with the 3-D vector:

$$p_i^c = \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix}, \; i = 1, \ldots, N$$

The geometry is instead simply represented by the 3-D coordinates of each point, i.e., by:

$$p_i^g = \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix}, \; i = 1, \ldots, N$$

As previously noted the scene segmentation algorithm should be insensitive to the relative scaling of the point-cloud geometry and should bring geometry and color distances into a consistent framework. Therefore, all components of $p_i^g$ are normalized with respect to the average of the standard deviations of the point coordinates in the three dimensions$\sigma_g = \left(\sigma_x + \sigma_y + \sigma_z\right)/3$, thus obtaining the vector:

$$\begin{bmatrix} \overline{x}(p_i) \\ \overline{y}(p_i) \\ \overline{z}(p_i) \end{bmatrix} = \frac{3}{\sigma_x + \sigma_y + \sigma_z} \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix} = \frac{1}{\sigma_g} \begin{bmatrix} x(p_i) \\ y(p_i) \\ z(p_i) \end{bmatrix}$$

In order to balance the relevance of color and geometry in the merging process, the color information vectors are also normalized by the average of the standard deviations of the L, a, and b components. The final color representation, therefore, is:

$$\begin{bmatrix} \overline{L}(p_i) \\ \overline{a}(p_i) \\ \overline{b}(p_i) \end{bmatrix} = \frac{3}{\sigma_L + \sigma_a + \sigma_b} \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix} = \frac{1}{\sigma_c} \begin{bmatrix} L(p_i) \\ a(p_i) \\ b(p_i) \end{bmatrix}$$

From the above normalized geometry and color information vectors, each point is finally represented as

$$p_i^f = \begin{bmatrix} \overline{L}(p_i) \\ \overline{a}(p_i) \\ \overline{b}(p_i) \\ \lambda\overline{x}(p_i) \\ \lambda\overline{y}(p_i) \\ \lambda\overline{z}(p_i) \end{bmatrix}$$

The parameter $\lambda$ controls the contribution of color and geometry to the final segmentation. High values of $\lambda$ increase the relevance of geometry, while low values of $\lambda$ increase the relevance of color information. Notice that at the two extrema the algorithm can be reduced to a color-based segmentation ($\lambda = 0$) or to a geometry (depth) only segmentation ($\lambda \to \infty$). A complete discussion on the effect of this parameter and a method to automatically tune it to the optimal value is presented in [27].

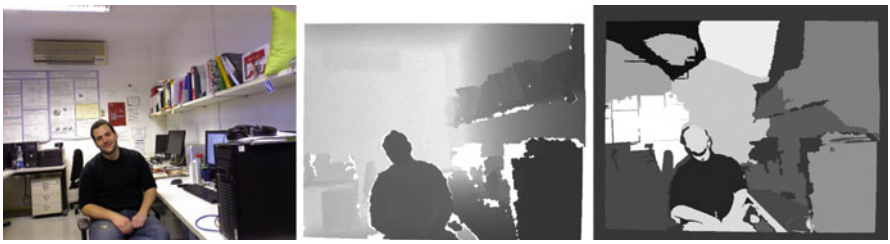

**Fig. 3** Color image, depth map, and segmentation map

The computed vectors $p_i^f$ are then clustered in order to segment the acquired scene. Mean shift clustering [26] has been used since it obtains an excellent trade-off between the segmentation accuracy and the computation and memory resources. A final refinement stage is also applied in order to remove regions smaller than a predefined threshold typically due to noise. In Fig. 3 an example of segmented image is reported.

## 2.2 Image Size Filter

The image size filter (SIZE) rejects candidate images according to their size. The size of a candidate face region is extracted from the depth map. Assuming that the face detection algorithm returns the 2D position and dimension in pixels ($w_{2D}$, $h_{2D}$) of a candidate face region, its 3D physical dimension in mm ($w_{3D}$, $h_{3D}$) can be estimated as:

$$w_{3D} = w_{2D}\frac{\bar{d}}{f_x}$$
$$h_{3D} = h_{2D}\frac{\bar{d}}{f_y}$$

where $f_x$ and $f_y$ are the Kinect camera focal lengths computed by the calibration algorithm in [25] and $\bar{d}$ is the average depth of the samples within the face candidate bounding box. Note how $\bar{d}$is defined as the median of the depth samples; this is done in order to reduce the impact of noisy samples in the average computation.

The candidate regions out of a fixed range [0.075, 0.35] centimeters are rejected.

## 2.3 Flatness\Unevenness Filter

Another source of significant information that can be obtained from the depth map is the flatness\unevenness of the candidate face regions. For this filter a segmentation procedure is applied. Then from each face candidate region the standard deviation of

the pixels of the depth map that belong to the larger segment is calculated. Regions having a standard deviation (STD) out of a fixed range [0.01, 2] are removed.

This method is slightly different to the one proposed in our previous work [1] (called STD° in the experimental section), where the flatness\unevenness was calculated using the whole candidate window.

## 2.4 Segmentation Based Filtering

From the segmented version of the depth image, it is possible to evaluate some characteristics of the candidate face based on its dimension with respect to its bounding box and its shape (which should be elliptical). According to these considerations, we define two simple filtering rules. The first evaluates the relative dimension of the larger region with respect to the whole candidate image (SEG). We have rejected the candidate regions where the area of the larger region is less than 40 % of the whole area.

The second considers the fitting score of an ellipse fitting approach[2] to evaluate whether the larger region can be modeled as an ellipse (ELL). The candidate regions with a cost higher than 100 are rejected.

## 2.5 Eye Based Filtering

The presence of two eyes is another good indicator of a face. In this work two efficient eye detectors are applied to candidate face regions [28]. The score of the eye detector is used to filter out regions having a very low probability of containing two eyes (EYE).

The first approach is a variant of the PS model proposed in [28]. PS is a computationally efficient framework for representing a face in terms of an undirected graph G = (V, E), where the vertices V correspond to its parts (i.e., two eyes, one nose, and one mouth) and the edge set E characterizes the local pairwise spatial relationship between the different parts. The PS approach proposed in [28] enhances the traditional PS model to handle the complicated appearance and structural changes of eyes under uncontrolled conditions.

The latter approach is proposed in [29] where the color information is used to build an eye map for emphasizing the iris area. Then, a radial symmetry transform is applied both to the eye map and the original image. Finally, the cumulative result of the transforms indicates the positions of the eye.

[2] http://it.mathworks.com/matlabcentral/fileexchange/3215-fit-ellipse.


**Fig. 4** Some samples of images filtered out by the different filtering rules

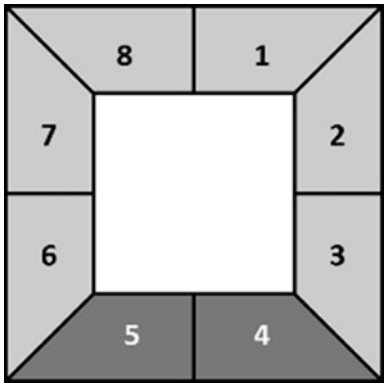


**Fig. 5** Partitioning of a neighborhood of the candidate face region into 8 sectors (*gray area*). Lower sectors 4, 5 are depicted in *dark gray*

Only the face candidates where a pair of eyes is found with a score higher than a prefixed threshold (1 for the first approach and 750 for the latter) are retained (Fig. 4).

## *2.6 Filtering Based on the Analysis of the Depth Values*

Excluding some critical cases like people lying on the floor, it is reasonable to assume the person body is present in the lower volume right under the face, while the remaining surrounding volume is likely to be empty. We exploit this observation in order to reject candidate faces whose neighborhood shows a different pattern from the expected one.

In particular, we enlarge the rectangular region associated to the candidate face in order to analyze a neighborhood of the face in the depth map, i.e., all the pixels which belong to the extended region but are not part of the smaller face box. The region is then partitioned into a number of radial sectors centered to the center of the candidate face. We used eight sectors in our experiments, see Fig. 5. For each sector $S_i$we count the number of pixels $n_i$ whose depth value $d_p$ is close to the average depth value of the face $\overline{d}$, *i.e.*, .

$$n_i = \left| \{ p \ : \ \left| d_p - \overline{d} \right| < t_d \wedge p \in S_i \} \right|$$

where we used $t_d = 50cm$. Finally, the number of pixels per sector is averaged on the two lower sectors ($S_4$ and $S_5$) and on the remaining ones, obtaining the two values $n_u$ and $n_l$ respectively. The ratio between $n_l$ and $n_u$ is then computed:

$$\frac{n_l}{n_u} = \frac{\frac{1}{2}(n_4 + n_5)}{\frac{1}{6}(n_1 + n_2 + n_3 + n_6 + n_7 + n_8)}$$

If the ratio drops below a certain threshold $t_r$, then the candidate face is removed. We set $t_r = 0.8$ in our experiments.

This approach is named SEC in the experimental section.

## 3 Experimental Results

For our experiments we use four datasets of faces acquired in an unconstrained setup ("in the wild") for purposes other than face detection. For preliminary experiments and parameter tuning, we used a separate set of images appositely collected for this aim. The images will be publicly available as a part of the Padua FaceDec dataset. All four datasets contain upright frontal images possibly with a limited degree of rotation, and all are composed of color images and their corresponding depth map:

- Microsoft hand gesture [30] is a dataset collected for gesture recognition composed of images of ten different people performing gestures; most of the images in the datasets are quite similar to each other, and each image contains only one face. Only a subset of the whole dataset (42 images) has been selected for our experiments and manually labeled with the face position.
- Padua Hand gesture [31] is a dataset similar to the previous one that was collected for gesture recognition purposes and composed of images from ten different people; each image contains only one face. A subset of 59 images has been manually labeled and selected for our experiments.
- Padua FaceDec [1], is a dataset collected and labeled for the purpose of face detection. It contains 132 images acquired both outdoors and indoors with the Kinect sensor at the University campus in Padua. Some images contain more than one face and some contain no faces. The images capture one or more people performing various daily activities (e.g., working, studying, walking, and chatting) in an unconstrained setup. Images have been acquired different hours of the day in order to account for varying lighting conditions, and some faces are partially occluded by objects and other people. For these reasons, this dataset is more challenging than previous datasets.
- Padua FaceDec2, is a new dataset acquired for the purpose of this chapter by a second generation Kinect sensor. For each scene a $512 \times 424$ depth map and a $1920 \times 1080$ color image have been acquired. The dataset includes 316 images of

**Table 1** Datasets characteristics

| Dataset | No. images | Color resolution | Depth resolution | No. faces | Difficulty |
|---|---|---|---|---|---|
| Microsoft hand gesture | 42 | 640 × 480 | 640 × 480 | 42 | Low |
| Padua hand gesture | 59 | 1280 × 1024 | 640 × 480 | 59 | Low |
| Padua FaceDec | 132 | 1280 × 1024 | 640 × 480 | 150 | High |
| Padua FaceDec2 | 316 | 1920 × 1080 | 512 × 424 | 363 | High |
| MERGED | 549 | – | – | 614 | High |

both indoor and outdoor scenes with people in various positions and challenging situations, e.g., with the head tilted with respect to the camera or very close to other objects. Some images contain more than one face and some contain no faces. Note that, even if the second generation Kinect, differently from the first version, is able to work outdoor, the outdoor depth data is noise, thus making the recognition problem more challenging. The depth data has been retro-projected over the color frame and interpolated to the same resolution thus obtaining two aligned depth and color fields.

A summary of the datasets characteristics is reported in Table 1.

The four datasets have been merged to form a single larger dataset consisting of 549 images containing 614 faces (only upright frontal faces with a maximum rotation of $\pm 30°$ have been considered). The parameter optimization of the face detectors has been performed manually and, despite the different origin and characteristics of the images included in the final set, the selected parameter optimizations have been fixed for all the images. The MERGED dataset is not an easy dataset to classify, as illustrated in Fig. 6, which presents some samples the face detectors could not accurately detect (even when executed with a very low recognition threshold). The collected dataset contains images with various lighting conditions, see Fig. 7.

Moreover, for comparing the face detectors and the proposed ensemble with other approaches proposed in the literature, we perform comparisons on the well-known BioID dataset, the foremost benchmark for upright frontal face detection. It is composed by 1521 images of 23 different people acquired during several identification sessions. The amount of rotation in the facial images is small. All the images are gray-scale and do not have a depth map. As a result, most of the filters proposed in this work are not applicable to the BioID dataset. Nonetheless, this dataset provides a means of comparing the approach proposed in this chapter with other state-of-the-art methods and is one way of showing the effectiveness of our ensembles.

The performance of the proposed approach is evaluated according the following well-known performance indicators:


**Fig. 6** Some samples where faces are not correctly detected

- **Detection rate (DR)**: the detection rate, also known as *recall* and *sensitivity*, is the fraction of relevant instances that are retrieved, i.e. the ratio between the number of faces correctly detected and the total number of faces (manually labelled) in the dataset. Let $d_l$ ($d_r$) be the Euclidean distance between the manually extracted $C_l$ ($C_r$) and the detected $C'_l$ ($C'_r$) left (right) eye position,[3] the relative error of detection is defined as $\mathrm{ED} = \max(d_l, d_r)/d_{lr}$ where the normalization factor $d_{lr}$ is the Euclidean distance of the expected eye centers used to make the measure independent of the scale of the face in the image and of image size.
- **False positives (FP)**: it is the number of candidate faces not containing a face.

[3]The face detectors FL and RF give the positions of the eye centers as the output, while for VJ and SN the detected eye position is assumed to be a fixed position inside the face bounding box.

**Fig. 7** Some samples with various lighting conditions

- **False negatives (FN)**: it is the number of faces not retrieved, i.e. the candidate faces erroneously excluded by the system. This value is correlated to the detection rate, since it can be obtained as (1-Detection Rate) × N°faces.

**Table 2** Performance of the four face detectors and some ensembles (last five rows) on the MERGED dataset (* denotes the use of adding poses)

| Face detector(σ)/ensemble | +Poses | DR | FP | FN |
|---|---|---|---|---|
| VJ(2) | No | 55.37 | 2528 | 274 |
| RF(−1) | No | 47.39 | 4682 | 323 |
| RF(−0.8) | No | 47.07 | 3249 | 325 |
| RF(−0.65) | No | 46.42 | 1146 | 329 |
| SN(1) | No | 66.61 | 508 | 205 |
| SN(10) | No | 46.74 | 31 | 327 |
| FL | No | 78.18 | 344 | 134 |
| VJ(2)* | Yes | 65.31 | 6287 | 213 |
| RF(−1)* | Yes | 49.67 | 19475 | 309 |
| RF(−0.8)* | Yes | 49.67 | 14121 | 309 |
| RF(−0.65)* | Yes | 49.02 | 5895 | 313 |
| SN(1)* | Yes | 74.59 | 1635 | 156 |
| SN(10)* | Yes | 50.16 | 48 | 306 |
| FL* | Yes | 83.39 | 891 | 102 |
| FL+ RF(−0.65) | No | 83.06 | 1490 | 104 |
| FL+ RF(−0.65) + SN(1) | No | 86.16 | 1998 | 85 |
| FL+ RF(−0.65) + SN(1)* | Mixed | 88.44 | 3125 | 71 |
| FL* + SN(1)* | Yes | 87.79 | 2526 | 75 |
| FL* + RF(−0.65) + SN(1)* | Mixed | 90.39 | 3672 | 59 |

In this dataset (due to the low quality of several images) we considered a face detected in an image if ED < 0.35.

The first experiment is aimed at comparing the performance of the four face detectors and their combination by varying the sensitivity factor σ (when applicable) and the detection procedure (i.e., when using or not using additional poses with 20°/−20° rotation).

For each detector in Table 2, the value fixed for the sensitivity threshold is shown in parentheses. We also compare in Table 2 different ensembles of face detectors. To reduce the number of false positives, all the output images having a distance $md \leq 30$ pixels are merged together.

From the result in Table 2, it is clear that the adding poses is not so useful for the RF face detector. This probably is due to the fact that RF has already been trained on images containing rotated faces. Moreover, using added poses strongly increases the number of false positives, as might be expected. For the ensembles, we report only the most interesting results. As can be seen in Table 2, combining more high performing approaches clearly boosts the detection rate performance. Unfortunately, the ensembles based on the three face detectors have too many false negatives.

Another interesting result is that of the 3125 false positives of "FL+ RF(−0.65) + SN(1)*" 2282 are found where the depth map has valid values, the

**Table 3** Performance of the same four face detectors and ensembles used above on the BioID dataset

| Face detector(σ)/ensemble | +Poses | DR (ED < 0.15) | DR (ED < 0.25) | DR (ED < 0.35) | FP |
|---|---|---|---|---|---|
| VJ(2) | No | 13.08 | 86.46 | 99.15 | 517 |
| RF(−1) | No | 87.84 | 98.82 | 99.08 | 80 |
| RF(−0.8) | No | 87.84 | 98.82 | 99.08 | 32 |
| RF(−0.65) | No | 87.84 | 98.82 | 99.08 | 21 |
| SN(1) | No | 71.27 | 96.38 | 97.76 | 12 |
| SN(10) | No | 72.06 | 98.16 | 99.74 | 172 |
| FL | No | 92.57 | 94.61 | 94.67 | 67 |
| VJ(2)* | Yes | 13.08 | 86.46 | 99.15 | 1745 |
| RF(−1)* | Yes | 90.53 | 99.15 | 99.41 | 1316 |
| RF(−0.8)* | Yes | 90.53 | 99.15 | 99.41 | 589 |
| RF(−0.65)* | Yes | 90.53 | 99.15 | 99.41 | 331 |
| SN(1)* | Yes | 71.33 | 96.52 | 97.90 | 193 |
| SN(10)* | Yes | 72.12 | 98.36 | 99.87 | 1361 |
| FL* | Yes | 92.57 | 94.61 | 94.67 | 1210 |
| FL+ RF(−0.65) | No | 98.42 | 99.74 | 99.74 | 88 |
| FL+ RF(−0.65) + SN(10) | No | 99.15 | 99.93 | 99.93 | 100 |
| FL+ RF(−0.65) + SN(1)* | Mixed | 99.15 | 100 | 100 | 281 |
| FL* + SN(1)* | Yes | 98.03 | 99.87 | 99.93 | 260 |
| FL* + RF(−0.65) + SN(1)* | Mixed | 99.15 | 100 | 100 | 1424 |

others are found where the values of depth map is 0 (i.e., the Kinect has not been able to compute the depth value due to occlusion, low reflectivity, too high distance or other issues), while all the true positives are found where exists the depth map.

In Table 3 we report the performance of the same face detectors on the BioID dataset. It is interesting to note that the creation of adding poses is not mandatory; if the acquisition is in a constrained environment, the performance is almost the same with or without the addition of artificial poses. Using artificial poses strongly increases the number of false positives. It is clear that different face detectors exhibit different behaviors as is evident by the fact that each is able to detect different faces. As a result of this diversity, the ensemble is able to improve the best stand-alone approaches. Another interesting result is the different behaviors exhibited by the same face detectors on the two different datasets. For instance, RF works very well on the BioID dataset but rather poorly on our dataset that contains several low quality faces. Regardless, in both datasets the same ensemble outperforms the other approaches.

The next experiment is aimed at evaluating the filtering steps detailed in Sect. 2. Since the first experiments proved that the best configuration (i.e., trade-off between performance and false positives) is the ensemble composed by FL+ RF(−0.65) + SN(1)*, the filtering steps are performed on the results of this detector. The performance after each filter or combination of filters is reported in Tables 4 and 5. The computation time reported in seconds is evaluated on a Xeon E5-1620 v2 –

**Table 4** Performance of different filtering steps on the MERGED dataset

| Filter | DR | FP | FN | Time |
|---|---|---|---|---|
| SIZE | 88.44 | 1247 | 71 | 0.000339 |
| STD | 88.44 | 2207 | 71 | 0.010865 |
| SEG | 88.44 | 2144 | 71 | 0.008088 |
| ELL | 88.44 | 1984 | 71 | 0.010248 |
| EYE | 88.44 | 1580 | 71 | 19.143445 |
| SEC | 88.11 | 1954 | 73 | 0.015302 |
| STD° | 87.79 | 2265 | 75 | 0.008280 |

**Table 5** Performance obtained combining different filtering steps on the MERGED dataset

| Filter combination | DR | FP | FN |
|---|---|---|---|
| SIZE | 88.44 | 1247 | 71 |
| SIZE + STD | 88.44 | 1219 | 71 |
| SIZE + STD + SEG | 88.27 | 1193 | 72 |
| SIZE + STD + SEG + ELL | 88.11 | 1153 | 73 |
| SIZE + STD + SEG + ELL + EYE | 88.11 | 1050 | 73 |
| SIZE + STD + SEG + ELL + SEC + EYE | 86.97 | 752 | 80 |

**Table 6** Performance obtained maximizing the reduction of the FP

| Filter | DR | FP | FN |
|---|---|---|---|
| SIZE | 87.79 | 944 | 75 |
| SIZE + STD | 87.79 | 908 | 75 |
| SIZE + STD + SEG | 87.79 | 877 | 75 |
| SIZE + STD + SEG + ELL | 87.30 | 852 | 78 |
| SIZE + STD + SEG + ELL+ EYE | 86.16 | 560 | 85 |
| SIZE + STD + SEG + ELL + SEC + EYE | 84.85 | 431 | 93 |

3.7 GHz – 64 GB Ram using Matlab R2014a on a candidate region of $78 \times 78$ pixels without parallelizing the code (note, however, that the different filters can be run in parallel).

It is clear that SIZE is the most useful criterion for removing the false positive candidates found by the ensemble of face detectors. The second best approach is the eye detector EYE. Although it works quite well, it has a high computational time. As a result, it cannot be used in all applications. The other approaches are less useful if taken individually; however, since they do not require a high computational cost, they can be useful in sequence to decrease the number of false positives. In applications where real-time detection is not mandatory (e.g. in face tagging), EYE filtering can be used in the ensemble to further reduce the number of false positives without decreasing the number of true positives.

As a final experiment, we report in Table 6 the performance of the different filters and combinations of filters by partially relaxing the thresholds, which greatly decreases the number of false positives even though some true positives are lost.

From the results of Table 6 it is evident that the proposed approach works even better than FL (which is known in the literature as one of the best performing face detectors). Even though these results have been obtained on a small dataset, we are confident that they are realistic and would perform comparatively in real-world conditions since the images contained in MERGE are very different from each other and include both images with a single frontal face and images acquired "in the wild" with multiple faces.

## 4 Conclusion

In this work a smart false positive-reduction method for face detection is proposed. An ensemble of state-of-the-art face detectors is combined with a set of filtering steps calculated both from the depth map and the color image. The main goal of this approach is to obtain accurate face detection with few false positives. This goal is accomplished using a set of filtering steps: the size of candidate face regions; the flatness or unevenness of the candidate face regions; the size of the larger cluster of the depth map of the candidate face regions; ellipse fitting to evaluate whether the region can be modeled as an ellipsis; and an eye detection step. The experimental section demonstrates that the proposed set of filtering steps reduces the number of false positives with little effect on the detection rate.

In conclusion, we show that the novel facial detection method proposed in this work is capable of taking advantage of a depth map to obtain increased effectiveness even under difficult environmental illumination conditions. Our experiments, which were performed on a merged dataset containing images with complex backgrounds acquired in an unconstrained setup, demonstrate the feasibility of the proposed system.

We are aware that the dimensions of the datasets used in our experiments are lower than most benchmark datasets containing only 2D data. It is in our intention to continue acquiring images to build a larger dataset with depth maps. In any case, the reported results obtained on this small dataset have statistical significance. As a result, we can confirm that the depth map provides criteria that can result in a significant reduction of the number of false positives.

## References

1. L. Nanni, A. Lumini, F. Dominio, P. Zanuttigh, Effective and precise face detection based on color and depth data. Appl. Comput. Inform. **10**(1), 1–13 (2014)
2. C. Zhang, Z. Zhang, A survey of recent advances in face detection. Microsoft Research Technical Report, MSR-TR-2010-66, June 2010
3. Z. Zeng, M. Pantic, G.I. Roisman, T.S. Huang, A survey of affect recognition methods: audio, visual, and spontaneous expressions. IEEE Trans. Pattern Anal. Mach. Intell. **31**(1), 39–58 (2009)

4. H.L. Jin, Q.S. Liu, H.Q. Lu, Face detection using one-class based support vectors, in *Proceedings 6th IEEE International Conference Automatic Face Gesture Recognition*, Hoboken, NJ 07030 USA, (2004), pp. 457–462
5. P. Viola, M.J. Jones, Rapid object detection using a boosted cascade of simple features, *Proceedings of the 2001 IEEE Computer Society conference on Computer Vision and Pattern Recognition*, **1**, 511–518 (2001)
6. C. Küblbeck, A. Ernst, Face detection and track in video sequences using the modified census transformation. Image Vis. Comput. **24**(6), 564–572 (2006)
7. C. Huang, H. Ai, Y. Li, S. Lao, High-performance rotation invariant multiview face detection. IEEE Trans. Pattern Anal. Mach. Intell. **29**(4), 671–686 (2007)
8. J. Wu, S. Charles Brubaker, M.D. Mullin, J.M. Rehg, Fast asymmetric learning for cascade face detection. IEEE Trans. Pattern Anal. Mach. Intell. **30**, 369–382 (2008)
9. M. Anisetti, Fast and robust face detection, in *Multimedia Techniques for Device and Ambient Intelligence*, Chapter 3 (Springer, US, 2009). ISBN: 978-0-387-88776-0
10. J. Li, Y. Zhang, Learning surf cascade for fast and accurate object detection, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2013) Hoboken, NJ 07030 USA
11. M. Mathias, et al. Face detection without bells and whistles, in *Computer Vision–ECCV 2014* (Springer International Publishing, 2014), Zurich, Switzerland, pp. 720–735
12. F. Tsalakanidou, D. Tzovaras, M.G. Strintzis, Use of depth and colour eigenfaces for face recognition. Pattern Recogn. Lett. **24**(910), 1427–1435 (2003)
13. R.I. Hg, P. Jasek, C. Rofidal, K. Nasrollahi, T.B. Moeslund, G. Tranchet, An RGB-D database using Microsoft's Kinect for windows for face detection, in *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems (SITIS)* (IEEE, 2012), Hoboken, NJ 07030 USA, pp. 42–46
14. M.-Y. Shieh, T.-M. Hsieh, Fast facial detection by depth map analysis. Math. Problems Eng. (2013), Article ID 694321, pp. 10, doi: 10.1155/2013/694321
15. J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-time human pose recognition in parts from single depth images. CVPR **2**, 3 (2011)
16. R. Mattheij, E. Postma, Y. Van den Hurk, P. Spronck, Depth-based detection using Haarlike features, in *Proceedings of the BNAIC 2012 Conference* (Maastricht University, The Netherlands, 2012), pp. 162–169
17. M. Anisetti, V. Bellandi, E. Damiani, L. Arnone, B. Rat, A3FD: Accurate 3D face detection, in *Signal Processing for Image Enhancement and Multimedia Processing* (Springer, US, 2008), pp. 155–165
18. F. Jiang, M. Fischer, H.K. Ekenel, B.E. Shi, Combining texture and stereo disparity cues for real-time face detection. Signal Process. Image Commun. **28**(9), 1100–1113 (2013)
19. G. Goswami, S. Bharadwaj, M. Vatsa, R. Singh, On RGB-D face recognition using Kinect, in *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (IEEE, 2013), Hoboken, NJ 07030 USA, pp. 1–6
20. Y. Taigman, M. Yang, M.A., Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2014), Hoboken, NJ 07030 USA, pp. 1701–1708
21. O. Jesorsky, K. Kirchberg, R. Frischholz, in *Face Detection Using the Hausdorff Distance*, eds. by J. Bigun, F. Smeraldi. Audio and Video based Person Authentication—AVBPA, (Springer, 2001), Berlin, Germany, pp. 90–95
22. N. Markuš, M. Frljak, I.S. Pandžić, J. Ahlberg, R. Forchheimer, Fast localization of facial landmark points, in *Proceedings of the Croatian Computer Vision Workshop*, Zagreb, Croatia, (2014)
23. M. Nilsson, J. Nordberg, I. Claesson, Face detection using local SMQT features and split up SNOW classifier. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP), **2**, Hoboken, NJ 07030 USA, 589–592 (2007)
24. A. Asthana, S. Zafeiriou, S. Cheng, M. Pantic, Robust discriminative response map fitting with constrained local models, in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2013), Hoboken, NJ 07030 USA, pp. 3444–3451

25. D. Herrera, J. Kannala, J. Heikkilä, Joint depth and color camera calibration with distortion correction. IEEE Trans. Pattern Anal. Mach. Intell. **34**, 2058–782 (2012)
26. D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
27. C. Dal Mutto, P. Zanuttigh, G.M. Cortelazzo, Fusion of geometry and color information for scene segmentation. IEEE J. Sel. Top. Signal Process. **6**(5), 505–521 (2012)
28. X. Tan, F. Song, Z.-H. Zhou, S. Chen, Enhanced pictorial structures for precise eye localization under incontrolled conditions, in *IEEE Conference on Computer Vision and Pattern Recognition 2009 (CVPR 2009)*, Hoboken, NJ 07030 USA, 20–25 June 2009, pp. 1621, 1628
29. E. Skodras, N. Fakotakis, Precise localization of eye centers in low resolution color images. Image Vision Comput. (2015). doi:10.1016/j.imavis.2015.01.006
30. Z. Ren, J. Meng, J. Yuan. Depth camera based hand gesture recognition and its applications in human-computer-interaction, in *Proceedings of ICICS* (2011), Hoboken, NJ 07030 USA, pp. 1–5
31. F. Dominio, M. Donadeo, P. Zanuttigh, Combining multiple depth-based descriptors for hand gesture recognition. Pattern Recogn. Lett. **50**, 101–111 (2014)