# Video Segmentation with Superpixels

Fabio Galasso[1], Roberto Cipolla[2], and Bernt Schiele[1]

[1] Max Planck Institute for Informatics, Saarbrücken, Germany
[2] University of Cambridge, Cambridge, UK

**Abstract.** Due to its importance, video segmentation has regained interest recently. However, there is no common agreement about the necessary ingredients for best performance. This work contributes a thorough analysis of various within- and between-frame affinities suitable for video segmentation. Our results show that a frame-based superpixel segmentation combined with a few motion and appearance-based affinities are sufficient to obtain good video segmentation performance. A second contribution of the paper is the extension of [1] to include motion-cues, which makes the algorithm globally aware of motion, thus improving its performance for video sequences. Finally, we contribute an extension of an established image segmentation benchmark [1] to videos, allowing coarse-to-fine video segmentations and multiple human annotations. Our results are tested on BMDS [2], and compared to existing methods.

## 1 Introduction

Segmentation is a fundamental problem in computer vision with many applications such as action recognition, 3D reconstruction, or video indexing. Many powerful image segmentation (IS) methods exist (e.g. [3–9]) and there is common agreement to use multiple similarities based on brightness, color and texture over local image patches to achieve best image segmentation performance.

Video segmentation (VS) is far less researched due to its computational complexity and the inherent difficulties such as camera-motion, occlusions, changes in scale, perspective, illumination and contrast, or non-rigid deformations. Intuitively, besides *within-frame* similarities used for image segmentation, VS should also use *between-frame* similarities to connect and thus segment corresponding regions across multiple frames. While recent work on VS proposes a variety of such between-frame similarities [2, 10–13] there is no common agreement yet on which similarities are necessary for best performance.

The main contribution of the present work is thus a systematic analysis of different between- and within-frame similarities in a unified framework. Similarities are novel terms or derived from other VS methods. The major result of the analysis is to identify the most powerful similarities that in combination achieve best performance. We further contribute an extension to a hierarchical image segmentation (HIS) [1] including motion cues, which improves significantly its performance for video-segmentation. Finally, we extend an established IS benchmark [1] to evaluate coarse-to-fine VS results on multiple human annotations.

K.M. Lee et al. (Eds.): ACCV 2012, Part I, LNCS 7724, pp. 760–774, 2013.

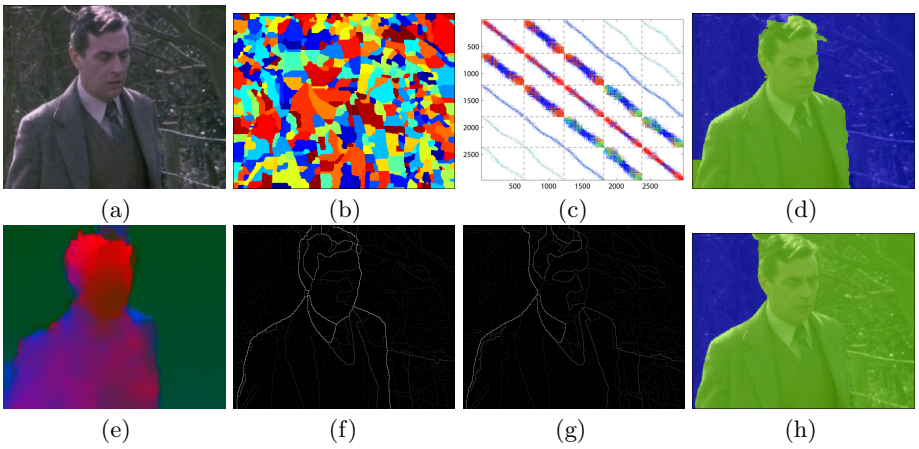



**Fig. 1.** (*Top row*) Proposed video segmentation model: we extract superpixels (b) with a novel motion-aware hierarchical image segmentation algorithm (MAHIS); we systematically analyze affinity matrices (c) based on novel and existing within- and between-frame superpixel similarities, and employ a spectral clustering framework to provide the final video segmentation (d). (*Second row*) The proposed MAHIS includes motion (e) and generates Ultrametric Contour Maps (f) which outperform the state-of-the-art (g) [1]. Note the ability of our video segmentation algorithm (d) to overcome the problems that standard image segmentation has for the right part of the image (h).

## 2 Related Work

A large body of literature on VS exists leveraging on appearance [11, 14–16], motion [2, 3, 12], or a combination of cues [10, 13, 17–23]. A variety of techniques is used, e.g. generative layered models [19, 20], graph-based models [15], mean-shift [17, 21], and techniques based on manifold-embedding and eigendecomposition, such as ISOMAP [12] and spectral clustering [2, 3]. Layered models [19, 20] have shown potential in learning general object motion and appearance, but are limited by their computational load. On the other hand graph-based [15] and meanshift techniques [21] may generalize to video sequences of arbitrary sizes, as they are based on local properties. We have chosen spectral clustering for our framework as it provides globally optimal solutions.

Recent works on VS have employed point trajectories [2], improving on the corresponding point track clustering literature [24, 25], and dense solutions are obtained with densification by non-linear diffusion [23] and graph-based methods [16]. Here we consider the dense video volume, arguing that sparse tracks do not capture the spatial cohesiveness of objects, as also maintained in [14].

Similarly to [11–13], we employ superpixels, which provide a desirable computational reduction and powerful within-frame representation. [11–13] extract region trajectories and provide video segmentations by respectively labelling them, in a CRF and ISOMAP framework. By contrast, we encompass an analysis of the

between-frame affinities which they use. Our results are useful to improve the quality of their region trajectories.

Other work exists which extends the HIS of [1] to include motion cues. Most notably [26] evaluates frame differences and optical flow among multiple frames and applies twice the machinery of [1], outperforming [1] for occlusion boundary detection. Our extension to [1] is straightforward and provides significant improvements for VS, which motivates further research on the topic.

## 3 Framework

Many segmentation approaches exist that could serve as a general framework for our analysis of different within- and between-frame similarities for video segmentation. For the purpose of the paper we have opted to use spectral clustering [3, 27] given its long tradition and state-of-art performance in a number of areas including image [1] and video object segmentation [2, 12]. The mathematical theory is well understood, and the general framework is well established since the pioneering work on normalized cuts [3].

The algorithm is based on an affinity matrix $W$ formed of pair-wise similarity scores $w_{i,j}$ between data elements $i$ and $j$. For image segmentation these data elements are often the image pixels themselves [1, 3] or for video object segmentation these elements might be point trajectories [2, 12]. While $W$ is quadratic in the number of elements, it is typically sparse as only a local spatial(-temporal) neighborhood is considered, making the approach computationally viable.

The affinity matrix $W$ is employed to embed the data elements onto a manifold, by eigendecomposing the normalized graph Laplacian $L$:

$$L = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}} = V^T \Lambda V \tag{1}$$

where $D$ is a diagonal matrix with entries $d_{i,i} = \sum_j w_{i,j}$, matrix $V$ contains the eigenvectors, and the diagonal matrix $\Lambda$ contains the eigenvalues. The mapping into the manifold is given by taking the eigenvectors $\{v_0, v_1, \ldots, v_m\}$ corresponding to the $m + 1$ smallest eigenvalues $0 = \lambda_0 < \lambda_1 < \ldots, < \lambda_{m+1}$ (as $\lambda_0 = 0$, $v_0$ is constant and can be discarded). Partitioning or segmentation of the data elements can be obtained with standard clustering schemes.

This method is well suited to analyze the contribution of different within- and between-frame affinities by simply defining entries of the affinity matrix $W$ using single or a combination of affinities. Here we employ the above approach twice in a two-step framework. The first step results in a motion-aware hierarchical image segmentation (MAHIS) from which we obtain superpixels for the second step. This step uses pixel-based affinities that are calculated both from brightness, color, and texture [1], as well as from optical flow (see sec. 4).

The second step is superpixel-based and uses a variety of within- and between-frame affinities (see sec. 5) to obtain the video segmentation result. The motivation to use superpixels for video-segmentation is two-fold. First, a drastic reduction of the computational complexity is achieved since the number of data-elements to be considered is lowered by two orders of magnitude. And second,

(a) (b) (c) (d) (e)

**Fig. 2.** (a,b) Two foreground objects move on a background with the same texture. None of the brightness, color and texture cues from the HIS algorithm of [1] detects the objects (c), which our motion cue (d) clearly identifies. Our proposed MAHIS provides the correct boundaries for the objects (e).

we can define richer and more powerful between-frame affinities than would be possible with pixel-based affinities alone.

Sec. 4 describes our motion-aware hierarchical image segmentation approach, sec. 5 the different between-frame affinities, and sec. 6 evaluated different combinations of those affinities in the context of video segmentation.

## 4 Motion-Aware Superpixels

Image segmentation is inherently ambiguous. In the frame shown in fig. 1(a), the jacket has a similar color as the background, clearly distinct from the skin and the shirt. Illumination and contrast help recover the true contours on the left contour of the body, but not the right contour. In fig. 1(g) the output of the image segmentation algorithm of [1] illustrates this expected result, also clear in the corresponding image segmentation in figure 1(h). On the other hand, the rich texture of the wood allows to accurately estimate optical flow, fig. 1(e), especially on the right side of the person, which perfectly complements appearance. By integrating our proposed motion cue into the algorithm of [1], we recover more respondent boundaries, fig. 1(f), and therefore superpixels.

Please note that not only the boundaries are better weighted, but the motion cue detects further boundaries in the image, as depicted in fig. 2 for a toy example. This further ensures that superpixels are conservatively representing the video without merging objects. On the other hand, the motion cue is zero for the static parts of the scene and the output is the same as from [1]. Here we describe our proposed MAHIS and validate it experimentally against [1].

### 4.1 Motion-Aware Hierarchical Image Segmentation

Optical flow has reached satisfactory maturity and suits the task of detecting motion. We use the dense optical flow algorithm of [28]. For each frame, horizontal $U(x, y)$ and vertical $V(x, y)$ optical flow components are composed by averaging the respective forward $U^+$ $V^+$ and backward $U^-$ $V^-$ estimates. The single-frame-averages smooth the flow and reduce the effect of outliers. A gradient is then computed for $U(x, y)$ and $V(x, y)$ with the histogram-based gradient operator, employed in [1] for the brightness BG, color CG and texture TG gradients. In particular, for both $U(x, y)$ and $V(x, y)$, we compute

**Table 1.** Our proposed image and video benchmark is used to compare our proposed VS and MAHIS against the HIS of [1] and the VS of [15] on BMDS [2]. All measures range $[0, 1]$, higher is better; only VI ranges $[0, \infty)$, lower is better. (*First part*) Our proposed MAHIS outperforms HIS on most metrics, most notably on boundary scores. This clearly identifies MAHIS as the better candidate to extract superpixels for VS. (*Second part*) Our VS outperforms HIS and MAHIS. STT+LTT+STM+STA is identified as the minimal best set of affinities, in agreement with fig. 4. (*Third part*) Our proposed VS outperforms [15] on all fronts by large margins.

| | Boundary | | | Region | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | SC | | | PRI | | VI | |
| | ODS | OSS | AP | ODS | OSS | Best | ODS | OSS | ODS | OSS |
| HIS of [1] | 0.30 | 0.37 | 0.18 | **0.75** | 0.79 | 0.82 | **0.70** | 0.79 | **0.75** | 0.72 |
| Proposed MAHIS | **0.35** | **0.43** | **0.23** | 0.74 | **0.81** | **0.84** | 0.69 | **0.82** | 0.76 | **0.67** |
| VS: All | **0.35** | 0.41 | 0.22 | **0.80** | **0.85** | **0.87** | **0.78** | **0.86** | **0.71** | **0.56** |
| VS: ABA+LTT | 0.23 | 0.28 | 0.13 | 0.74 | 0.77 | 0.81 | 0.72 | 0.78 | 0.71 | 0.71 |
| VS: ABM+LTT | 0.21 | 0.25 | 0.11 | 0.74 | 0.76 | 0.80 | 0.72 | 0.77 | 0.71 | 0.71 |
| VS: STT | 0.16 | 0.20 | 0.08 | 0.74 | 0.75 | 0.78 | 0.72 | 0.75 | 0.71 | 0.70 |
| VS: STT+LTT | 0.20 | 0.24 | 0.12 | 0.74 | 0.76 | 0.79 | 0.72 | 0.77 | 0.71 | 0.71 |
| VS: STM | 0.32 | 0.33 | 0.22 | 0.74 | 0.76 | 0.78 | 0.72 | 0.78 | 0.71 | 0.69 |
| VS: STA | 0.19 | 0.25 | 0.11 | 0.74 | 0.75 | 0.76 | 0.72 | 0.75 | 0.71 | 0.71 |
| VS: STT+LTT+STM | 0.30 | 0.35 | 0.19 | 0.76 | 0.82 | 0.85 | 0.75 | 0.83 | 0.71 | 0.60 |
| VS: STT+LTT+STM+STA | **0.35** | 0.40 | 0.22 | **0.80** | **0.85** | **0.87** | **0.78** | **0.86** | **0.71** | **0.56** |
| VS: All-ABA | **0.35** | 0.41 | 0.22 | **0.80** | **0.85** | **0.87** | **0.78** | **0.86** | **0.71** | **0.56** |
| VS: All-ABM | **0.35** | 0.40 | 0.22 | **0.80** | **0.85** | **0.87** | 0.77 | **0.86** | **0.71** | 0.57 |
| VS: All-STT | 0.32 | 0.39 | 0.21 | 0.74 | 0.82 | 0.85 | 0.73 | 0.83 | 0.71 | 0.65 |
| VS: All-LTT | 0.30 | 0.33 | 0.19 | 0.74 | 0.78 | 0.79 | 0.72 | 0.80 | 0.71 | 0.67 |
| VS: All-STM | 0.24 | 0.29 | 0.13 | 0.74 | 0.76 | 0.80 | 0.72 | 0.77 | 0.71 | 0.70 |
| VS: All-STA | 0.31 | 0.36 | 0.19 | 0.76 | 0.82 | 0.86 | 0.75 | 0.83 | 0.71 | 0.60 |
| Our VS | **0.35** | 0.41 | 0.22 | **0.80** | **0.85** | **0.87** | **0.78** | **0.86** | **0.71** | **0.56** |
| VS of [15] | 0.20 | 0.23 | 0.10 | 0.74 | 0.76 | 0.78 | 0.72 | 0.76 | **0.71** | 0.71 |

the gradients $\Delta_\rho U(x, y, \theta)$ and $\Delta_\rho V(x, y, \theta)$ along 8 orientations $\theta \in (0, \pi]$ and 3 scale octaves $\rho$ (same parameters as [1]). Finally the flow gradient FG, for each orientation $\theta$ and scale $\rho$ sample, is given by the respective squared sums $\mathrm{FG}_\rho(x, y, \theta) = \left[\Delta^2_\rho U(x, y, \theta) + \Delta^2_\rho V(x, y, \theta)\right]^{-\frac{1}{2}}$.

The new FG is then considered an additional channel, alongside BG, CG and TG, passed by the boundary detector to the spectral partitioning process, and finally to the OWT-UCM machinery which produces the hierarchical segmentation, following the pipeline and setup of [1]. The superpixels are conservatively extracted from the finest (over-)segmentation provided.

### 4.2 Experimental Evaluation

Our proposed MAHIS directly addresses the hierarchical image segmentation of video frames. For single images, or static frames, MAHIS provides identical

result as the HIS of [1]. We thus test MAHIS on a video dataset including camera and object motion. Recently, [2] has provided the Berkeley motion segmentation dataset (BMDS), with 26 video sequences. The dataset includes persons, cars and other objects, and various degrees of motion. Our proposed MAHIS is evaluated against HIS of [1] at the provided ground truth frames.

We use the established evaluation metrics for HIS, for which benchmark code is publicly available [1]. The evaluation considers both boundaries and regions. The former are benchmarked using precision-recall. The latter are evaluated with three metrics: segmentation covering (SC) the degree of overlap between the ground truth and the machine segmentation; probabilistic rand index (PRI) the fraction of pairs of pixels consistently labelled; variation of information (VI) the distance between segmentations in term of mutual information and conditional entropy. For all metrics the optimal dataset scale (ODS) and optimal segmentation scale (OSS - namely OIS in [1]) are reported: best aggregated performance over the dataset for a fixed scale and for the best scale for each segmentation. Additionally the benchmark reports average precision (AP) for boundaries and Best for region SC, i.e. best selection of segments across scales.

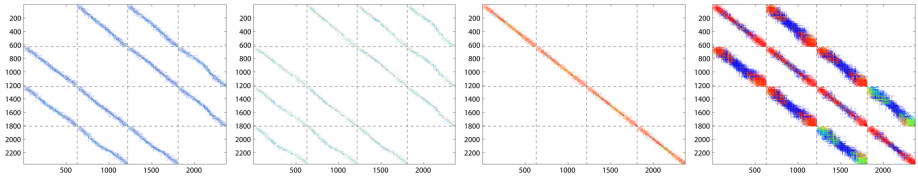
Table 1(first part) illustrates the results: our proposed MAHIS outperforms [1] on most metrics. An improvement on the region metrics is desirable when extracting superpixels so as not to span multiple objects. An improvement in the boundary metric is also desirable so that superpixels on different objects are better separated. Most notably, the boundary ODS and OIS score outperform [1] by about 17%, our AP improves by 28%. These results clearly shows the potential of our proposed MAHIS for the extraction of superpixels for VS.

## 5 Superpixel Affinities for Video Segmentation

As discussed in sec. 3 the first step of our method extracts superpixels (sec. 4) and the second step uses within- and between-frame superpixel affinities to derive the final video segmentation result. This section motivates and introduces various affinities that are analyzed in sec. 6 alone and in combination.

There are two major dimensions that we explore in this paper to define affinities. The first dimension is the type of information used to calculate affinities. We use appearance (based e.g. on color, brightness, and texture) as well as motion and spatial overlap of superpixels in successive frames. The second dimension is time or the number of frames considered to calculate affinities. Besides within a single frame, one can also use affinities calculated across neighboring frames or even across a potentially large number of frames. Intuitively, affinities connecting superpixels across many frames may enable good video segmentation performance. However, in general, affinity matrices should be sparse to allow computationally viable eigendecompositions of the graph Laplacian.

Fig. 3 illustrates samples for four of the six affinity matrices introduced below. Superpixels are ordered with an increasing index, according to their top-down left-right position in the frame and to their frame. Dashed lines delineate the frame partitioning. Terms on the block diagonal correspond to within-frame



(a) STT (b) LTT (c) ABM (d) STM

**Fig. 3.** Affinity matrices for 5 frames of sequence Marple1. Non-zero terms are colored blue to red. Note the different structure and sparsity of the affinity matrices. Affinity ABA has a similar structure as ABM and STA has a similar structure as STM.

affinities (e.g. ABM, fig. 3(c)) and terms off the block-diagonal correspond to between-frame affinities (e.g. STT and LTT, fig. 3(a,b)). In the following, affinity terms are grouped into three categories: between-frame, within-frame, and combined between- and within-frame affinities.

### 5.1 Between-Frame Affinities

The first two affinities measure the spatial overlap of superpixels in different frames. The first measure – STT – is taken from [12] that used it in the context of video-object segmentation. As STT is restricted to neighboring frames we define a new overlap term building on long-term point-trajectories [29].

**Short-Term-Temporal Affinity – STT.** [12] measures the similarity of superpixels by propagating the binary support mask of a superpixel with optical flow to neighboring frames and measuring their overlap by the Dice measure. Given superpixel $p_f^i$ at frame $f$ with binary mask $m_{p_f^i}$, and superpixel $p_{f'}^j$ at frame $f' = f \pm 1, 2$ with mask $m_{p_{f'}^j}$, the STT affinity score $w^{stt}_{p_f^i, p_{f'}^j}$ is given by:

$$w^{stt}_{p_f^i, p_{f'}^j} = \frac{2|m^{f'}_{p_f^i} \cap m_{p_{f'}^j}|}{|m^{f'}_{p_f^i}| + |m_{p_{f'}^j}|} \tag{2}$$

where $m^{f'}_{p_f^i}$ indicates the propagated mask of superpixel $p_f^i$ to frame $f'$.

**Long-Term-Temporal Affinity – LTT.** In order to calculate superpixel affinities across many frames that are potentially hundreds of frames apart we leverage on the recently introduced long-term point trajectories of [29]. Let $\Phi_{p_f^i} \subseteq \Upsilon = \{T^i\}_{i=1}^{Q}$ be a subset of all trajectories, containing those trajectories $T^i$ intersecting superpixel $p_f^i$. We define the LTT affinity score $w^{ltt}$ between superpixels $p_f^i$ and $p_{f'}^j$, $f' = f + N, N \neq 0$ to be the Dice coefficient between the intersection sets $\Phi_{p_f^i}$ and $\Phi_{p_{f'}^j}$ of the superpixels:

$$w^{ltt}_{p_f^i, p_{f'}^j} = \frac{2|\Phi_{p_f^i} \cap \Phi_{p_{f'}^j}|}{|\Phi_{p_f^i}| + |\Phi_{p_{f'}^j}|} \tag{3}$$

Figs. 3(a,b) illustrate the LTT and STT affinities. While both measure spatial overlap of superpixels there are two major differences. By design, LTT allows to calculate affinities between frames that can be hundreds of frames apart, due to the long-term nature of the point trajectories. However, since not all superpixels contain point trajectories, the LTT affinity matrix is much sparser than the STT matrix, although this only calculates affinities between superpixels in neighboring frames (in practice ±2 frames are used).

### 5.2 Within- and Between-Frame Affinities

One way to measure similarities between pixels across frames are spatiotemporal affinities based on appearance and/or motion [3, 17]. In order to overcome the computational complexity related to measuring pixel-affinities, [11] proposed to measure spatiotemporal affinities between superpixels instead. The first measure defined below – STA – is directly related to [11] and measures the appearance affinity. The second term – STM – focuses on the motion affinity of superpixels.

**Spatio-Temporal-Appearance Affinity – STA.** To score the appearance affinity we use the median brightness and color $\overline{Lab}_{p_f^i}$ of a superpixel $p_f^i$ using CIE Lab color space. The STA affinity between pairs of superpixels $p_f^i$ and $p_{f'}^j$ in a spatiotemporal neighborhood (±1 frame, 2-layered neighborhood) is therefore:

$$w^{sta}_{p_f^i, p_{f'}^j} = \exp\left\{-\lambda_{sta} || \overline{Lab}_{p_f^i} - \overline{Lab}_{p_{f'}^j} ||\right\} \tag{4}$$

The affinity is inspired by [11]. More elaborate extensions use the $\chi^2$ distance between appearance histograms for video segmentation [15, 16].

**Spatio-Temporal-Motion Affinity – STM.** This term calculates affinities based on motion to allow grouping of superpixels of the same moving objects. Given the median optical flow $\overline{\mathbf{u}}_{p_f^i}$ of a superpixel $p_f^i$ the STM affinities $w^{stm}$ are calculated between superpixels $p_f^i$ and $p_{f'}^j$ in a spatio-temporal neighborhood (±1 frames, 2-layered neighborhood) as:

$$w^{stm}_{p_f^i, p_{f'}^j} = \exp\left\{-\lambda_{stm} || \overline{\mathbf{u}}_{p_f^i} - \overline{\mathbf{u}}_{p_{f'}^j} ||^2\right\} \tag{5}$$

The STM affinity has been employed in numerous works for video segmentation [3], or in combination with an STA affinity [17]. These works use STM affinities between pixels, here we use it for superpixels.

### 5.3 Within-Frame Affinities

The terms defined here are complementary to STA and STM in the sense that they focus on the local similarities near the common boundary between superpixels rather than the median appearance and motion of the superpixels. The appearance based term – ABA – directly uses the contour maps of our MAHIS

(sec. 4). The motion based term – ABM – is one of two terms used for occlusion boundary detection in [26].

**Across-Boundary-Appearance Affinity – ABA.** Motivated by the success of the HIS-algorithm [1] for hierarchical segmentation we propose to measure appearance affinity using our improved MAHIS algorithm. We define the affinity $w^{aba}$ between pairs of neighboring superpixels $p_f^i$ and $p_f^j$ as the average value $\overline{v}_f^{ij}$ of the ultrametric contour map (see fig. 1 for a ultrametric contour map) along the common boundary of the superpixels:

$$w^{aba}_{p_f^i, p_f^j} = \overline{v}_f^{ij} \tag{6}$$

**Across-Boundary-Motion Affinity – ABM.** While STM measures the similarity of the median motion of two superpixels, ABM measures the local similarity of motion along the common boundary of two superpixels. This measure allows to connect superpixels e.g. in the case of non-rigid motions where the median motion of the superpixels can be quite different but the motion on both sides of the common boundary between superpixels might be similar. We consider $\overline{\mathbf{u}}^f(\mathbf{x})$, a dense optical flow field [28], locally median filtered (first temporally ($\pm 2$ frames), then spatially (3 px radius within the superpixel)). Given $\Psi_f^{ij}$ the set of pixel pairs on opposite sides across the common boundary between $p_f^i$ and $p_f^j$, the ABM affinity $w^{abm}$ is defined as:

$$w^{abm}_{p_f^i, p_f^j} = \exp\left\{ -\lambda_{abm} \frac{\sum_{(\mathbf{x}_i^m, \mathbf{x}_j^m) \in \Psi_f^{ij}} ||\overline{\mathbf{u}}^f(\mathbf{x}_i^m) - \overline{\mathbf{u}}^f(\mathbf{x}_j^m)||^2}{|\Psi_f^{ij}|} \right\} \tag{7}$$

The ABM affinity has been proposed for occlusion boundary detection in [26], in combination with an affinity similar to ABA.

## 6 Experimental Validation

The VS literature does not yet provide a common benchmark or evaluation metric that is agreed upon and widely used such as the Berkeley image segmentation benchmark. Some work [15] only provide a qualitative evaluation, others introduce datasets and metrics [2, 11, 12], but few compare on a common dataset [2, 12, 23]. Here we use BMDS [2] (see also sec. 4.2), because it is a publicly available, of reasonable complexity and various papers show results for this dataset [2, 12, 23]. Following [11, 12], we perform dense clustering of the video sequences, and restrict the sequences to the first 100 frames. For each setting, we vary the number of clusters, in the range [1,600]. We assign each video segment to a ground truth label based on maximal region overlap, and score performance by global and average (over each ground truth frame) per-pixel labeling error, i.e. fraction of misclassified pixels. Note that the global per-pixel error is dominated by the large segments in the scene (often the background) and that the average error weights all ground-truth segments equally.

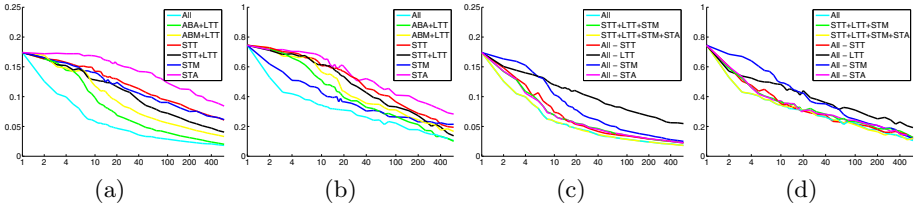



**Fig. 4.** (a) Global and (b) average error curves comparing All combined affinities against the individual terms; (c) global and (d) average error curves comparing All against All minus one term. Please note the different scaling of the $y$-axis for (a,c). Curves are evaluated on BMDS [2]. We conclude from (c,d) that LTT and STM are most contributing to the VS performance and that the combination STT+LTT+STM+STA (green curve) equals the performance of All (cyan).

Fig. 4 presents the results: first we evaluate the performance of individual affinities to identify those contributing the most to VS, and then we aim to determine the minimal set providing best overall performance. The first two plots illustrate the performances of individual affinities and compare them to the overall best performance when using all affinities. Since ABA and ABM are within-frame affinities only we pair them with LTT to analyze their 'individual' performance, since LTT relates some superpixels only, it is paired with STT. Overall, the lowest error is obtained by all combined affinities (All, cyan, fig. 4(a,b)). As for the average error (fig. 4(b)), STM (blue) is the single best, followed by ABA+LTT (green) and ABM+LTT (yellow). STT+LTT (black) and STT (red) are slightly worse and the weakest affinity is STA (magenta). The ordering is nearly identical for the global error (fig. 4(a)) with the exception of ABA+LTT being best (green) and STM being one of the weaker overall (blue). While it can be concluded that STA does not perform well, none of the other affinities stands out to be better than any other.

The second two plots in fig. 4 reveal more insights into which of the affinities are essential to obtain the overall best performance when combined. For this we compare the performance of All (cyan) when taking out individual affinities, namely STM (All-STM, blue), LTT (All-LTT, black), STT (All-STT, red) and STA (All-STA, magenta). Note the significant drop in performance for the first two, All-STM(blue) and All-LTT(black), both for the global (fig. 4(c)) and the average error (fig. 4(d)). A less significant drop is observed for All-STT(red) and All-STA(magenta), while the performance is not altered when taking out the boundary affinities ABM and ABA (not reported due to space constraints). The performance of STT+LTT+STM+STA(yellow) nearly superposes All(cyan), therefore barely visible, while the performance drops slightly for STT+LTT+STM(green). Boundary terms surprisingly do not improve performance for VS, while they turned out useful to improve HIS [26].

These quantitative results are supported by qualitative results. Fig. 5 illustrates segmentation results when extracting 10 objects (i.e. clusters) from the video sequences Cars6 and People1. It shows (*column-wise*) All terms, the minimal best set STT+LTT+STM+STA, the temporal terms STT+LTT, the individual best

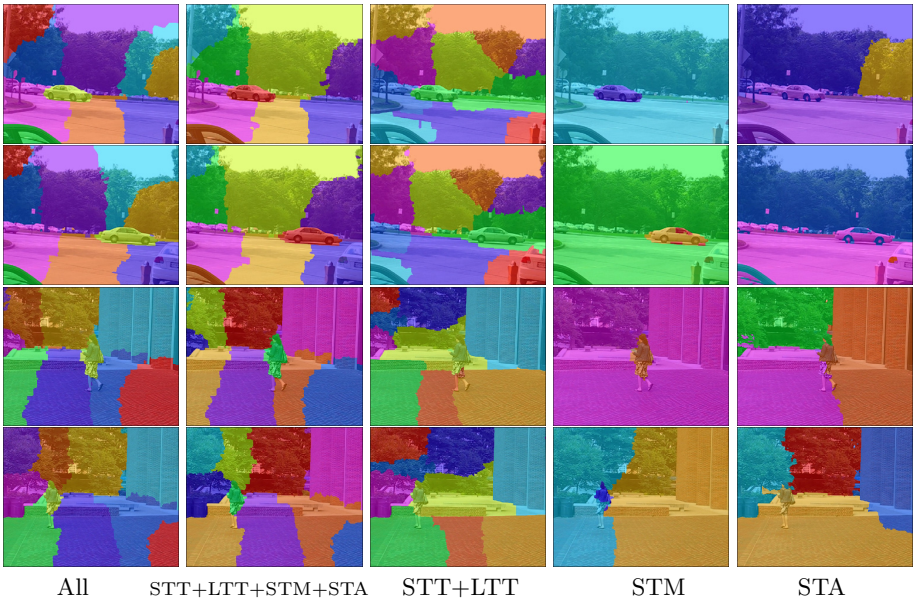



**Fig. 5.** 10-cluster video segmentations extracted from the video sequences Cars6 (*top two rows*) and People1 (*bottom two rows*). STT+LTT+STM+STA provides the same qualitative result as All affinities. STT+LTT: please note the "drag" effects generated by the imprecise optical flow at the contours of the moving objects. STM: the affinity gets more effective where the motion is larger, which makes STM the perfect complement to STT+LTT. STA: clusters are denoted by strong color differences. Although performing poorly, the term supplements the three motion affinities effectively.

performer STM, and the appearance term STA. As expected from the error metric All and STT+LTT+STM+STA provide the same qualitative results (while the random colors are different, the segmentation results are nearly identical). Temporal terms STT+LTT successfully track the image parts but suffer from "drag" effects in cases of large motion and imprecise optical flow. By contrast, STM addresses the moving objects, and employs the 10 object "allowance" to neatly segment them from the background. The appearance term STA segments the scene according to the strongest color differences. As from the numerical results, STT+LTT and STM are complementary motion terms, which are supplemented by STA, although the latter alone has a poor segmentation performance.

Finally, fig. 6 illustrates the use of our algorithm to extract a minimal number of clusters, as for obtaining object cut-outs. The figure also discusses some typical failure cases.

Next we discuss how we might obtain a commonly agreed upon evaluation metric. We believe that a cause for the lack of an established evaluation metric is mainly twofold: i) no standard format is given to write a segmentation output, i.e. the benchmark of [2] requires a conversion of the labelled video into point trajectories for evaluation; ii) the aspects of coarse-to-fine and over-segmentation

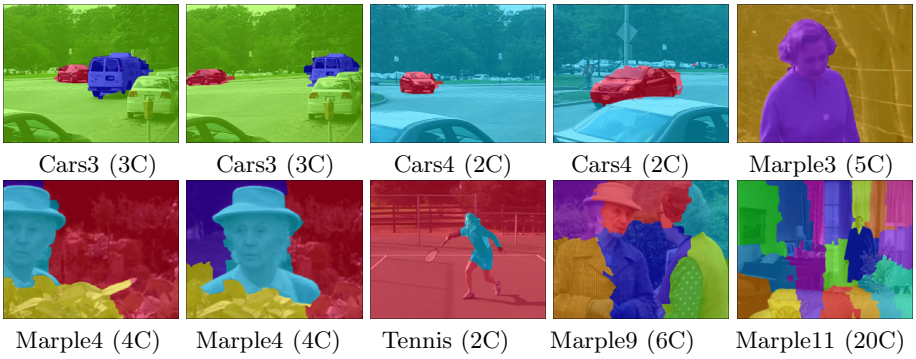



**Fig. 6.** Video segmentations provided with a minimal number of clusters (C). The algorithm successfully segments objects in the first five examples. Marple9 and Marple11 are failure cases. Marple9: the right actress moves little, so it is wrongly segmented due to the misleading appearance differences; the left actress, also minimally moving, rotates the head but the motion is too small. Marple11: the person is not segmented even with 20 clusters, as he does not move and the scene boundaries are prevalent.

are not clearly addressed, i.e. the "right" video segmentation and number of clusters usually depends on the task, and may vary for the same video sequence.

These problems however have already been addressed for HIS [1], so that we propose to extend those established metrics to VS. We employ the same metrics as described in sec. 4.2, namely precision-recall for boundaries and SC, PRI and VI for regions, aggregating performance optimally for a fixed dataset scale (ODS) and for the best (video) segmentation scale (OSS). The extended benchmark uses video (spatio-temporal) segments, which it evaluates against all the ground truth frames altogether over the video sequence, thus addressing temporal consistency. The benchmark allows for evaluating coarse-to-fine VS on multiple ground truths, as for evaluating more general VS, without a specific defined task. We define the coarse-to-fine VS levels by varying the number of clusters in the range [1,600].

We evaluate the extended benchmark metrics on all BMDS videos, and report the results in table 1(second part) for the same setups of affinities as in fig. 4. These results confirm most findings of fig. 4: ABA+LTT and ABM+LTT are single best performers for the region evaluation, while the single best boundaries are provided by STM. On both boundary and region metrics, STT+LTT+STM+STA provides the same performance as All, while removing ABA and ABM from All does not alter performance. Interestingly, the best VS results (VS:All and VS:STT+LTT+STM+STA) are comparable to the best IS (MAHIS) on boundary metrics but neatly superior on region metrics, notwithstanding the additional temporal consistency constraint. This confirms the importance to consider segmentation as a spatio-temporal problem.

In table 1(third part), we also compare our VS with the algorithm of [15] (as implemented by [30]), which we outperform by ∼12% on regions and ∼80% on boundaries.

**Table 2.** Comparison with [2] on BMDS according to the benchmark of [2]. Both our 10-cluster (k=10) and 20-cluster (k=20) video segmentations are comparable in error to [2]. Notably we provide 100% density.

| | Density | Overall error | Average error | Over-segmentation | Extracted objects |
|---|---|---|---|---|---|
| Our VS (k=10) | 100 | 9.92 | 16.52 | 6.77 | 17 |
| Our VS (k=20) | 100 | 5.84 | 15.20 | 16.27 | 18 |
| Method of [2] | 3.30 | 3.93 | 23.83 | 0.92 | 29 |

On a final note, we also evaluate our VS against [2] on BMDS employing their evaluation metric. Our average error is much lower although segmenting all pixels (density) rather than just a fraction; the number of extracted objects and overall errors are worse, although better for a larger number of clusters; the over-segmentation index is approximately fixed, given the number of clusters.

## 7 Conclusion and Future Work

We have proposed a model for unsupervised video segmentation based on clustering superpixels. We have analyzed a variety of affinities in isolation and in combination and have identified a minimal set that obtains best performance. While the use of superpixels necessarily results in an approximation we have shown that powerful affinity scores can be defined based on them and that good video segmentation performance can be obtained. A second contribution of the paper is the motion aware hierarchical image segmentation algorithm that is a direct extension of [1] to also include motion features improving their approach for image sequences. Finally, we have extended an established image segmentation benchmark to videos. We used it to evaluate our algorithm under different setups and to compare with a state-of-the-art algorithm [15]. The extended benchmark allows evaluating coarse-to-fine segmentations on multiple human ground truth annotations, although these are not yet provided by any video dataset.

## References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: From contours to regions: An empirical evaluation. In: CVPR (2009)
2. Brox, T., Malik, J.: Object Segmentation by Long Term Analysis of Point Trajectories. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 282–295. Springer, Heidelberg (2010)
3. Shi, J., Malik, J.: Normalized cuts and image segmentation. TPAMI (2000)
4. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. PAMI 24, 603–619 (2002)
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV 59 (2004)
6. Vese, L., Chan, T.: A multiphase level set framework for image segmentation using the mumford and shah model. IJCV 50, 271–293 (2002)

7. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the piecewise smooth mumford-shah functional. In: ICCV (2009)
8. Carreira, J., Sminchisescu, C.: Constrained parametric min-cuts for automatic object segmentation. In: CVPR (2010)
9. Endres, I., Hoiem, D.: Category Independent Object Proposals. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 575–588. Springer, Heidelberg (2010)
10. Levinshtein, A., Sminchisescu, C., Dickinson, S.: Spatiotemporal Closure. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 369–382. Springer, Heidelberg (2011)
11. Vazquez-Reina, A., Avidan, S., Pfister, H., Miller, E.: Multiple Hypothesis Video Segmentation from Superpixel Flows. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 268–281. Springer, Heidelberg (2010)
12. Galasso, F., Iwasaki, M., Nobori, K., Cipolla, R.: Spatio-temporal clustering of probabilistic region trajectories. In: ICCV (2011)
13. Cheng, H.T., Ahuja, N.: Exploiting nonlocal spatiotemporal structure for video segmentation. In: CVPR (2012)
14. Brendel, W., Todorovic, S.: Video object segmentation by tracking regions. In: ICCV (2009)
15. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. In: CVPR (2010)
16. Lezama, J., Alahari, K., Sivic, J., Laptev, I.: Track to the future: Spatio-temporal video segmentation with long-range motion cues. In: CVPR (2011)
17. DeMenthon, D.: Spatio-temporal segmentation of video by hierarchical mean shift analysis. In: Statistical Methods in Video Processing Workshop (2002)
18. Greenspan, H., Goldberger, J., Mayer, A.: A Probabilistic Framework for Spatio-Temporal Video Representation amp Indexing. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 461–475. Springer, Heidelberg (2002)
19. Kannan, A., Jojic, N., Frey, B.J.: Generative model for layers of appearance and deformation. In: AISTATS (2005)
20. Kumar, M.P., Torr, P., Zisserman, A.: Learning layered motion segmentations of video. IJCV 76, 301–319 (2008)
21. Paris, S.: Edge-Preserving Smoothing and Mean-Shift Segmentation of Video Streams. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 460–473. Springer, Heidelberg (2008)
22. Lee, Y.J., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: ICCV (2011)
23. Ochs, P., Brox, T.: Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In: ICCV (2011)
24. Brostow, G.J., Cipolla, R.: Unsupervised bayesian detection of independent motion in crowds. In: CVPR (2006)
25. Sugimura, D., Kitani, K.M., Okabe, T., Sato, Y., Sugimoto, A.: Using individuality to track individuals: clustering individual trajectories in crowds using local appearance and frequency trait. In: ICCV (2009)
26. Sundberg, P., Brox, T.: Maire, M., Arbelaez, P., Malik, J.: Occlusion boundary detection and figure/ground assignment from optical flow. In: CVPR (2011)
27. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: NIPS (2001)

28. Zach, C., Pock, T., Bischof, H.: A Duality Based Approach for Realtime TV-$L^1$ Optical Flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 214–223. Springer, Heidelberg (2007)
29. Sundaram, N., Brox, T., Keutzer, K.: Dense Point Trajectories by GPU-Accelerated Large Displacement Optical Flow. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 438–451. Springer, Heidelberg (2010)
30. Xu, C., Corso, J.J.: Evaluation of super-voxel methods for early video processing. In: CVPR (2012)