Spatio-Temporal Segmentation with Depth-Inferred Videos of Static Scenes

Hanqing Jiang, Guofeng Zhang, and Hujun Bao

September, 2010
Technical Report

State Key Lab of CAD&CG, Zhejiang University
{jianghq, zhangguofeng, bao}@cad.zju.edu.cn

Abstract. Extracting spatio-temporally consistent segments from a video sequence is a challenging problem due to the complexity of color, motion and occlusions. Most existing spatio-temporal segmentation approaches rely on pairwise motion estimation, which have inherent difficulties in handling large displacement with significant occlusions. This paper presents a novel spatio-temporal segmentation method for depth-inferred videos. The depth data of input videos can be estimated beforehand using a multiview stereo technique. Our framework consists of two steps. In the first step, in order to make the extracted 2D segments temporally consistent, we introduce a spatial segmentation based on the probabilistic boundary maps, by collecting the boundary statistics in a video. In the second step, the consistent 2D segments in different frames are matched to initialize the volume segments. Then we compute the segment probability for each pixel by projecting it to other frames to collect the statistics, and incorporate it into the spatio-temporal segmentation energy to explicitly enforce temporal coherence constraint. The spatio-temporal segmentation results are iteratively refined in a video, so that a set of spatio-temporally consistent volume segments are finally achieved. The effectiveness of our automatic method is demonstrated using a variety of challenging video examples.

1 Introduction

Image and video segmentation has long been a fundamental problem in computer vision. It is useful in many applications, such as object recognition, image/video annotation, video stylization, and video editing. However, unsupervised segmentation is an inherently ill-posed problem due to the large number of unknowns and the possible geometric and motion ambiguities in the computation.

Recent advances in structure-from-motion [1] and stereoscopic vision [2–4] have made it possible to create high-quality depth maps with a handheld camera. The increasing prevalence of range sensors also implies that achieving high-quality depth images is becoming more and more convenient and flexible. For instance, time-of-flight senors that provide realtime range estimates have been available at commodity prices. Therefore, it is not difficult to imagine that most

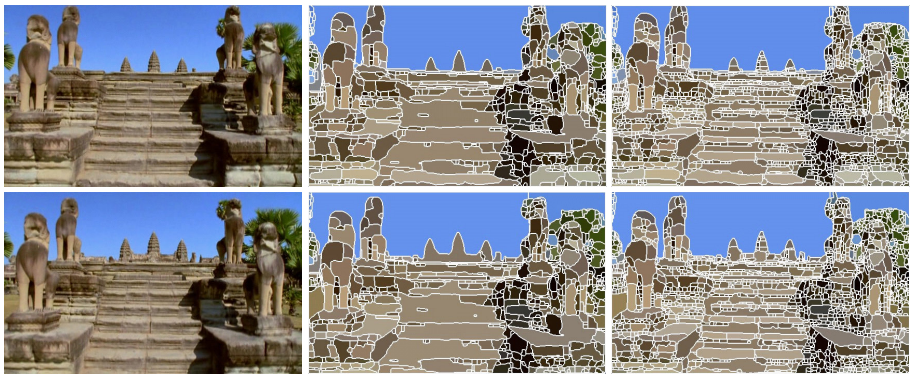

Fig. 1. Spatio-temporal segmentation of "Stair" sequence. Left column: the selected images from the input sequence. Middle and right columns: our segmentation results with different segment numbers.

of captured images will have depth data in the future. How to appropriately utilize depth information for segmentation is becoming an important issue. To the best of our knowledge, it has not yet been thoroughly discussed in literatures, especially for video segmentation.

In this paper, we propose a novel depth-based video segmentation method with the objective that the extracted segments not only preserve object boundaries but also maintain the temporal consistency in different images. The spatio-temporal segmentation is crucial for many video applications, such as content-based video editing, which typically requires that the modified video content should maintain high temporal consistency over frames.

We assume that the scene is static and there is a depth map for each frame of the input video. The depth data could be achieved by using a depth camera or multiview stereo techniques [5, 3, 4]. In our experiments, we capture videos with a handheld camera, and compute the depth maps by the multiview stereo method proposed in [4]. Our method contributes the following two aspects. First, we introduce "probabilistic boundary" which can be computed by collecting the boundary statistics in a sequence. The experimental results demonstrate that our spatial segmentation with probabilistic boundary can preserve object boundaries well and obtain much more consistent segments than those of directly using the state-of-the-art image segmentation methods. Second, we introduce a novel spatio-temporal segmentation method which iteratively refines the spatio-temporal segmentation by associating multiple frames in a video. By projecting each pixel to other frames, we can reliably compute the segment probability and incorporate it into our data term definition. The modified data term can significantly improve the spatio-temporal segmentation results, so that a set of spatio-temporally coherent volume segments can be achieved.

Our method is very robust against occlusions. We have conducted experiments on a variety of challenging examples. One challenging example is shown

in Fig. 1, in which the scene contains complex color and structure. Besides maintaining the temporal coherence, our method also allows controlling the granularity of segmentation result. Readers are referred to our supplementary video for inspecting the complete segmentation results.

2 Related Work

During the past decades, many state-of-the-art image segmentation methods have been proposed, such as mean shift [6], normalized cuts [7], watershed algorithm [8], and segmentation by weighted aggregation (SWA) [9]. Mean shift based image segmentation is often adopted in practice for its promising performance. However, for a video sequence, if we directly use these image-based segmentation methods to segment each frame independently, the segmentation results will be inconsistent for different images due to the lack of necessary temporal coherence constraints.

Some spatio-temporal segmentation methods [10] have been proposed to extend segmentation from single image to videos. Two main types of segmentation criteria (i.e. motion and color/texture) are generally used alone or in combination for video segmentation. Motion-based segmentation methods [11] aim to group pixels that undergo similar motion, and separate them into multiple layers. Many of them [12–14] need to estimate optical flow first, and then segment the pixels based on the learned motion models. Some of them [15, 16, 11] combine motion estimation and segmentation together, and iteratively refine them. However, pure motion-based methods are difficult to achieve high-quality segmentation results and usually produce inaccurate object boundaries due to the motion ambiguity and the difficulties of accurate optical flow estimation. Some works that combine color and motion cues for spatio-temporal segmentation are proposed. Khan and Shah [14] proposed a MAP framework for video segmentation combining multiple cues including spatial location, color and motion.

For video segmentation, both spatial and temporal dimensions should be considered. Most approaches handle these two types of dimensions separately. For example, many approaches [17, 18] first perform spatial segmentation of each frame, and then perform temporal grouping to obtain spatio-temporal volumes. Due to the complexity of color, motion and occlusions in a video, it is challenging for spatial segmentation to produce very consistent segments in different images, so that the obtained spatio-temporal segments by temporal grouping will easily contain obvious artifacts. Some methods [19, 20] employ a progressive scheme to obtain consistent segments across frames, that each frame is segmented according to the segmentation information propagated from previous frames. Zitnick et al. [16] proposed to combine segmentation and optical flow estimation together to produce consistent segments for a pair of images. However, all these methods are difficult to handle significant occlusions, where large groups of segments appear or disappear.

Some space-time segmentation methods [12, 13] are proposed to combine spatial and temporal grouping together, by treating the image sequence as a 3D vol-

ume and attempting a segmentation of pixel volumes. These methods typically construct a weighted graph by taking each pixel as a node and connecting the pixels that are in the spatiotemporal neighborhood of each other. Normalized cuts are typically used to partition the spatiotemporal volume. Some space-time segmentation methods define a high dimensional feature vector for each pixel by integrating multiple cues (such as color, space, motion, and time), and cluster these feature points via mean shift analysis [21, 22] or GMM [23]. However, all these methods are sensitive to large displacement with significant occlusions. Especially, if an object temporarily disappear due to occlusion or out-of-view, it is quite challenging for these methods to cluster the corresponding regions into the same segment.

Our work is also closely related to joint segmentation techniques [24, 25], which simultaneously segment the reconstructed 3D points and the registered 2D images. Given the multiple view images, they aim to semantically organize the recovered 3D points and obtain semantic object segmentation, which requires user assistance. In contrast, our method can automatically obtain a set of spatio-temporally consistent volume segments from a video sequence.

In summary, spatio-temporal segmentation is still a very challenging problem. Previous approaches generally have difficulties for handling large displacement with significant occlusions. In this paper, we show that by associating multiple frames on the inferred dense depth maps, surprisingly spatio-temporal consistent segments can be obtained from video sequences. The high-quality segmentation results can benefit many other applications, such as 3D modeling, video editing, and non-photorealistic rendering.

3 Our Approach

Given a video sequence with n frames, our objective is to estimate a set of spatio-temporal volume segments $S = \{S^k | k = 1, 2, ..., K\}$, where K is the volume segment number. For a pixel $\mathbf{x}_t$ in frame t, we denote $S(\mathbf{x}_t) = S^k$ if $\mathbf{x}_t \in S^k$. The color of pixel $\mathbf{x}_t$ is denoted as $I(\mathbf{x}_t)$, defined in Luv color space. Denoting by $z_{\mathbf{x}_t}$ the depth value of pixel $\mathbf{x}_t$, the disparity $D(\mathbf{x}_t)$ is defined as $D(\mathbf{x}_t) = 1/z_{\mathbf{x}_t}$ by convention.

We start by using the structure-from-motion (SFM) method proposed in [26] to recover the camera motion parameters from the input video sequence. The set of camera parameters for frame t is denoted as $\mathbf{C}_t = \{\mathbf{K}_t, \mathbf{R}_t, \mathbf{T}_t\}$, where $\mathbf{K}_t$ is the intrinsic matrix, $\mathbf{R}_t$ is the rotation matrix, and $\mathbf{T}_t$ is the translation vector. With the recovered camera poses, we then employ the multi-view stereo method of Zhang et al. [4] to recover a set of consistent depth maps. The computed depth maps will be used in the following segmentation process.

4 Spatial Segmentation with Probabilistic Boundary

Directly obtaining spatio-temporal volume segments in a video is difficult due to the large number of unknowns and the possible geometric and motion ambigui-

ties in the segmentation. Therefore, we design an iterative optimization scheme to achieve spatio-temporal video segmentation. For initialization, instead of directly segmenting each frame independently, we first compute the probabilistic boundary map by collecting the statistics of segment boundaries among multiple frames. Then we perform spatial segmentation for each frame independently with the computed probabilistic boundary maps. Our experimental results demonstrate that much more consistent segmentation results can be obtained than those of directly using mean shift algorithm.

4.1 Probabilistic Boundary

We first use mean shift algorithm [6] to segment each frame independently with the same parameters. The 2D segments in frame t are denoted as $s_t = \{s_t^k | k = 1, ..., K_t'\}$. For a pixel $\mathbf{x}_t$ in frame t, we denote $s(\mathbf{x}_t) = s_t^k$ if $\mathbf{x}_t \in s_t^k$. Fig. 2(b) shows the segmentation results of the selected frames, which are not consistent in different images. The segmented boundaries are quite flickering, and a segment may span over multiple layers, which is obviously not good enough as a starting point for spatio-temporal segmentation.

With the computed depths, we can project each pixel to other frames to find the correspondences. Considering a pixel $\mathbf{x}_t$ in frame t, with the estimated depth value $z_{\mathbf{x}_t}$, its projection $\mathbf{x}_{t'}$ in frame t' can be computed as follows:

$$\mathbf{x}_{t'}^h \sim z_{\mathbf{x}_t} \mathbf{K}_{t'} \mathbf{R}_{t'}^\top \mathbf{R}_t \mathbf{K}_t^{-1} \mathbf{x}^h + \mathbf{K}_{t'} \mathbf{R}_{t'}^\top (\mathbf{T}_t - \mathbf{T}_{t'}), \tag{1}$$

where the superscript h denotes the vector in the homogeneous coordinate system. The 2D point $\mathbf{x}_{t'}$ is computed by dividing $\mathbf{x}_{t'}^h$ by the third homogeneous coordinate. Then we compute the probabilistic boundary as follows:

$$p_b(\mathbf{x}_t, \mathbf{y}_t) = \frac{1}{n_v} \sum_{t'} [s(\mathbf{x}_{t'}) \neq s(\mathbf{y}_{t'})], \tag{2}$$

where $\mathbf{y}_t$ is a neighboring pixel of $\mathbf{x}_t$ in frame t, and n_v denotes the number of valid mapping. A mapping is defined to be valid, if the projection points $\mathbf{x}_{t'}$ and $\mathbf{y}_{t'}$ in frame t' are neither occluded nor out-of-view. If $p_b(\mathbf{x}_t, \mathbf{y}_t)$ is large, it is very likely that there is a boundary across pixels $\mathbf{x}_t$ and $\mathbf{y}_t$. Compared to the traditional segmentation boundaries in single image, our probabilistic boundary map is computed with multiple frames, which is robust to image noise and occasional segmentation errors. The computed probabilistic boundary maps are shown in Fig. 2(c), which are surprisingly consistent among different frames. The reason is that mean shift segmentation can preserve object boundaries well. Although the generated segment boundaries by mean shift may be occasionally inaccurate in one frame, it still has large chance to be accurate in other frames. By collecting the boundary statistics in multiple frames, the computed probabilistic boundaries can naturally preserve the object boundaries and maintain consistent in neighboring frames.

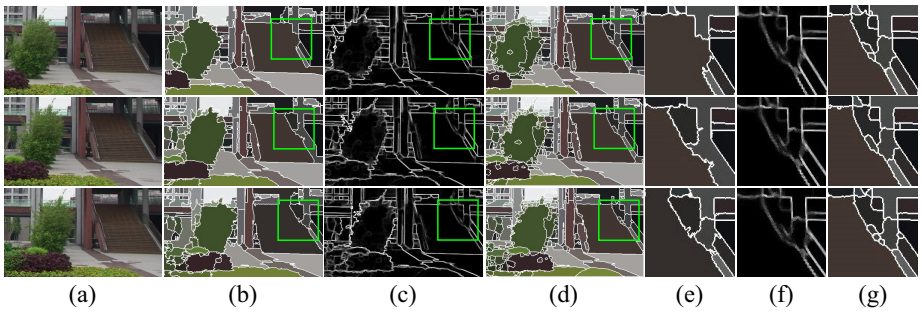

Fig. 2. Spatial segmentation with probabilistic boundary. (a) Three selected frames. (b) The segmentation results with mean shift. (c) The computed probabilistic boundary maps. (d) Our spatial segmentation results. (e-g) The magnified regions of (b-d). Compared to the results of mean shift, our segmentation results better preserve object boundaries and are much more consistent in different images.

4.2 Spatial Segmentation

With the computed probabilistic boundary map, we use the watershed algorithm [27] to segment the image. We compute a topographic surface $\mathcal{T}(\mathbf{x}_t) = \max_{\mathbf{y}_t \in N(\mathbf{x}_t)} p_b(\mathbf{x}_t, \mathbf{y}_t)$, the maximal probabilistic boundary over the 4 connected probabilistic edges for each pixel, and apply watershed transformation on the surface. The topological map is clipped with a threshold value δ to avoid over segmentation. Fig. 3(c) shows the segmentation result. We notice that some quite small segments appear around the areas with strong probabilistic boundaries, most of which are segmentation noise and do not consistently appear in neighboring frames. So, we eliminate too small segments (with less than 30 pixels), and set the pixels in these segments as unlabeled ones. The remaining 2D segments in frame t are denoted as $s_t = \{s_t^k | k = 1, ..., K_t\}$. The set of unlabeled pixels is denoted as Φ_t, which will be assigned to these K_t segments. We use $s(\mathbf{x}_t)$ to denote the assigned 2D segment for pixel $\mathbf{x}_t$.

For each frame t, we define the following energy for spatial segmentation:

$$E(s_t) = \sum_{\mathbf{x}_t \in \Phi_t} \big(E_d(s(\mathbf{x}_t)) + \sum_{\mathbf{y}_t \in N(\mathbf{x}_t)} E_s(s(\mathbf{x}_t), s(\mathbf{y}_t)) \big), \tag{3}$$

where $N(\mathbf{x}_t)$ denotes the set of neighbors of pixel $\mathbf{x}_t$. Data term E_d measures how well the pixels fit the assigned clusters, and the spatial smoothness term E_s encodes the segmentation continuity.

The data term E_d is defined using the Gaussian models of color, disparity and spatial distributions:

$$\begin{aligned} E_d(s(\mathbf{x}_t)) = -w_c \log \mathcal{N}(I(\mathbf{x}_t) | \mu^c_{s(\mathbf{x}_t)}, \Sigma^c_{s(\mathbf{x}_t)}) \\ -w_d \log \mathcal{N}(D(\mathbf{x}_t) | \mu^d_{s(\mathbf{x}_t)}, \Sigma^d_{s(\mathbf{x}_t)}) - w_s \log \mathcal{N}(\mathbf{x}_t | \eta_{s(\mathbf{x}_t)}, \Delta_{s(\mathbf{x}_t)}), \end{aligned}$$

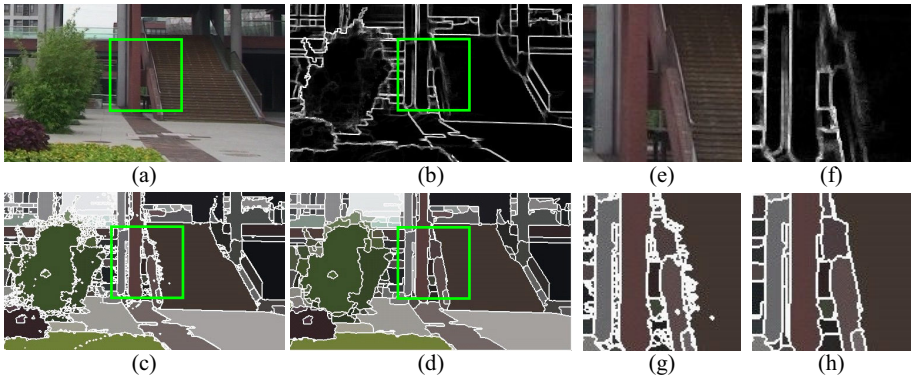

Fig. 3. Flow illustration of spatial segmentation. (a) One original image. (b) The computed probabilistic map. (c) The segmentation results by watershed algorithm based on the computed probabilistic map. (d) After solving (3), the unlabeled pixels are fused into nearby segments. (e-h) The magnified regions of (a-d), respectively.

where w_c, w_d and w_s are the weights. $\mathcal{N}(I(\mathbf{x}_t)|\mu^c_{s(\mathbf{x}_t)}, \Sigma^c_{s(\mathbf{x}_t)})$ describes the color distribution of segment $s(\mathbf{x}_t)$, where $\mu^c_{s(\mathbf{x}_t)}$ and $\Sigma^c_{s(\mathbf{x}_t)}$ are the mean color and covariance matrix, respectively. $\mathcal{N}(D(\mathbf{x}_t)|\mu^d_{s(\mathbf{x}_t)}, \Sigma^d_{s(\mathbf{x}_t)})$ describes the disparity distribution, which is similarly defined. $\mathcal{N}(\mathbf{x}_t|\eta_{s(\mathbf{x}_t)}, \Delta_{s(\mathbf{x}_t)})$ describes the spatial distribution of the segment $s(\mathbf{x}_t)$, where $\eta_{s(\mathbf{x}_t)}$ is the mean position coordinate, and $\Delta_{s(\mathbf{x}_t)}$ is the covariance matrix.

In order to preserve discontinuity, our spatial smoothness term is defined in an anisotropic way, encouraging the segment discontinuity to be coincident with the probabilistic boundary, color contrast and depth discontinuity. It is defined as

$$E_s(s(\mathbf{x}_t), s_t(\mathbf{y}_t)) = [s(\mathbf{x}_t) \neq s(\mathbf{y}_t)] \cdot \\ (\lambda_b \frac{\varepsilon_b}{p_b(\mathbf{x}_t, \mathbf{y}_t) + \varepsilon_b} + \lambda_c \frac{\varepsilon_c}{\|I(\mathbf{x}_t) - I(\mathbf{y}_t)\| + \varepsilon_c} + \lambda_d \frac{\varepsilon_d}{\|D(\mathbf{x}_t) - D(\mathbf{y}_t)\| + \varepsilon_d}), \tag{4}$$

where λ_b, λ_c and λ_d are the smoothness weights. ε_b, ε_c, and ε_d control the contrast sensitivity.

Since it is a labeling problem, we can use belief propagation algorithm to solve (3) for spatial segmentation. We only need to solve the segment labeling of the pixels in Φ_t, and the segment labels of other pixels are all fixed. In our experiments, the 2D segment number for each frame is around $300 \sim 2000$. So it will be very time-consuming and requires a very large memory space if we use a standard belief propagation algorithm like [28] to solve (3). In order to speed up and break through the limitation of memory space, we perform label pruning. In fact, only a small number of labels need to be considered for each pixel, since the cost of most labels are very large. Therefore, for each pixel, we only consider a few closest segments (70 segments in our experiments) with similar colors and depths. This strategy can well address the limitation of memory space and

dramatically accelerate BP optimization. The spatial segmentation results are shown in Fig. 2(d) and 3(d). The segmentation results in different images are rather consistent, which provide a good starting point for the following spatio-temporal segmentation.

5 Spatio-Temporal Segmentation

Due to the lack of explicit temporal coherence constraint, the spatial segmentation results may contain inconsistent segments. In addition, the segments in different images are not matched. In the following stage, we will perform spatio-temporal segmentation to achieve a set of pixel volumes. First, we need to match the segments in different images and link them to initialize volume segments.

5.1 Initializing Spatio-temporal Volumes

Without loss of generality, we consider two 2D segments s_t^k in frame t and $s_{t'}^{k'}$ in frame t'. With the depths, we can project s_t^k from frame t to t', and $s_{t'}^{k'}$ from frame t' to t, respectively. The projection mask of s_t^k from frame t to t' is denoted as $s_{t\to t'}^k$, and the projection mask of $s_{t'}^{k'}$ from frame t' to t is denoted as $s_{t'\to t}^{k'}$. An illustration is shown in Fig. 4. We can use their overlapping rate to define the matching confidence. If $\min(|s_{t\to t'}^k \cap s_{t'}^{k'}|/|s_{t'}^{k'}|, |s_{t'\to t}^{k'} \cap s_t^k|/|s_t^k|) > \delta_v$, where δ_v is a threshold, we think s_t^k and $s_{t'}^{k'}$ are matched.

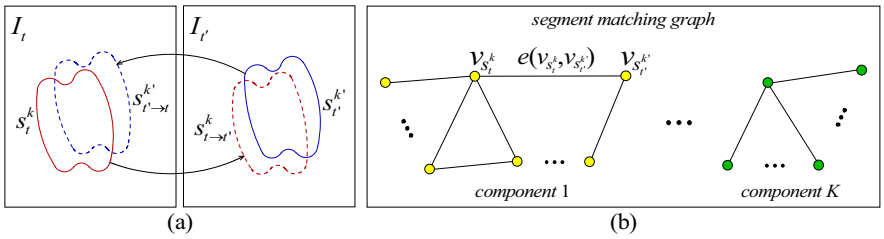

Fig. 4. Segment matching and Linking. (a) Segments s_t^k and $s_{t'}^{k'}$ are projected to frame t' and t, respectively, for segment matching. (b) The connected segment components. Each component represents a volume segment.

Each 2D segment can be projected to other frames, to find its matched segments in other frames. With these correspondences, we can build a matching graph. It is an undirected graph $G = (\mathcal{V}, \mathcal{E})$. Each 2D segment s_t^k corresponds to a vertex $v_{s_t^k} \in \mathcal{V}$, and every pair of matched segments (s_t^k and $s_{t'}^{k'}$) has an edge $e(v_{s_t^k}, v_{s_{t'}^{k'}})$ connecting them, as illustrated in Fig. 4(b). Each connected component represents a volume segment. The initialized volume segments are denoted as $S = \{S^k | k = 1, 2, ..., K\}$. One example is shown in Figs. 5(b) and (d). Most segments are already quite consistent. Then we perform an iterative optimization to further improve the results.

5.2 Iterative Optimization

For a pixel $\mathbf{x}$ in frame t, its corresponding pixel $\mathbf{x}_{t'}$ in frame t' can be computed by (1). Due to segmentation error, the segment labels of pixels $\mathbf{x}$ and $\mathbf{x}_{t'}$ may be different, i.e. $S(\mathbf{x}_{t'}) \neq S(\mathbf{x}_t)$. If there is no occlusion or out-of-view, each projection should correspond to a valid segment. In our experiments, we found that most of these projected segments are the same, which indicates that our initialized volume segments are already quite good. We use $P(\mathbf{x}_t)$ to denote the set of segment candidates for pixel $\mathbf{x}_t$, which includes these projected volume segments and $S(\mathbf{x}_t)$. Then, we define the segment probability of pixel $\mathbf{x}_t$ as:

$$L_h(l, \mathbf{x}_t) = \frac{1}{|P(\mathbf{x}_t)|} \sum_{t'} [S(\mathbf{x}_{t'}) = l], \tag{5}$$

where $\mathbf{x}_{t'}$ is the projected pixel in frame t' of pixel $\mathbf{x}_t$. $L_h(l, \mathbf{x}_t)$ denotes the probability of each segment label l for pixel $\mathbf{x}_t$. Obviously, $L_h(l, \mathbf{x}_t)$ will be a large value if the assigned segment label l is consistent with most of the projected segments. For each pixel $\mathbf{x}_t$, we only need to consider the segment candidates in $P(\mathbf{x}_t)$, because the probabilities of other labels are all zeros.

We define the spatio-temporal segmentation energy in a video as follows:

$$E(S) = \sum_{t=1}^{n} \sum_{\mathbf{x}_t} \big(E'_d(S(\mathbf{x}_t)) + \sum_{\mathbf{y}_t \in N(\mathbf{x}_t)} E_s(S(\mathbf{x}_t), S(\mathbf{y}_t)) \big), \tag{6}$$

where $N(\mathbf{x}_t)$ denotes the set of spatial neighbors of pixel $\mathbf{x}_t$ in frame t. The energy contains two components, i.e. data term E'_d and smoothness term E_s. E_s is the same as that of (4), and only E'_d is largely modified by incorporating the temporal coherence constraint in a statistical way.

The data term E'_d contains four components:

$$\begin{aligned} E'_d(S(\mathbf{x}_t), \mathbf{x}_t) = &-w'_h \log L_h(S(\mathbf{x}_t), \mathbf{x}_t) - w'_c \log L_c(S(\mathbf{x}_t), \mathbf{x}_t) \\ &-w'_d \log L_d(S(\mathbf{x}_t), \mathbf{x}_t) - w'_s \log L_s(S(\mathbf{x}_t), \mathbf{x}_t), \end{aligned}$$

where w'_h, w'_c, w'_d and w'_s are the cost weights. $L_c(S(\mathbf{x}_t), \mathbf{x}_t)$ describes the Gaussian distribution of color, and is simply defined as:

$$L_c(S(\mathbf{x}_t), \mathbf{x}_t) = \mathcal{N}(I(\mathbf{x}_t) | \mu^c_{S(\mathbf{x}_t)}, \Sigma^c_{S(\mathbf{x}_t)}), \tag{7}$$

where $\mu^c_{S(\mathbf{x}_t)}$ and $\Sigma^c_{S(\mathbf{x}_t)}$ are the mean color and covariance matrix of volume segment $S(\mathbf{x}_t)$, respectively. $L_d(S(\mathbf{x}_t), \mathbf{x}_t)$ describes the Gaussian distribution of disparity, and is similarly defined as:

$$L_d(S(\mathbf{x}_t), \mathbf{x}_t) = \mathcal{N}(D(\mathbf{x}_t) | \mu^d_{S(\mathbf{x}_t)}, \Sigma^d_{S(\mathbf{x}_t)}), \tag{8}$$

where $\mu^d_{S(\mathbf{x}_t)}$ and $\Sigma^d_{S(\mathbf{x}_t)}$ are the mean disparity and covariance matrix of volume segment $S(\mathbf{x}_t)$, respectively.

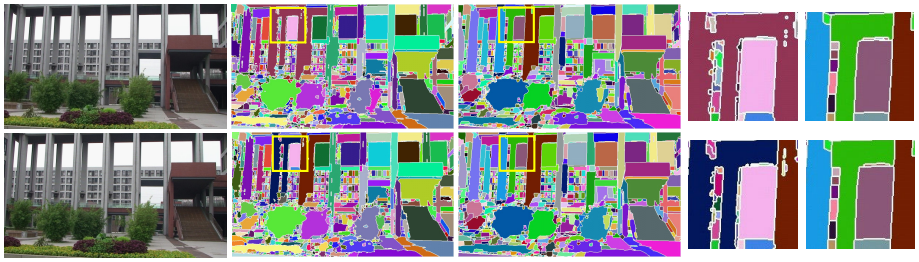

Fig. 5. Spatio-temporal segmentation results of "Campus" sequence. (a) Two selected original images. (b) The initialized volume segments. The pixels in the same volume segment are represented with the same color. (c) The final volume segments after iterative optimization, which become more consistent and better preserve object boundaries. (d) The magnified regions of (b), highlighted with yellow rectangles. (e) The magnified regions of (c).

$L_s(S(\mathbf{x}_t), \mathbf{x}_t)$ describes the shape distribution by mixture of Gaussians, which is defined as follows:

$$L_s(S(\mathbf{x}_t), \mathbf{x}_t) = \frac{1}{|f(S(\mathbf{x}_t))|} \sum_{t' \in f(S(\mathbf{x}_t))} \mathcal{N}(\mathbf{x}_{t'} | \eta_{s(\mathbf{x}_{t'})}, \Delta_{s(\mathbf{x}_{t'})}), \tag{9}$$

where $f(S(\mathbf{x}_t))$ denotes the frames spanned by $S(\mathbf{x}_t)$, and $\mathbf{x}_{t'}$ is the corresponding pixel in frame t' for pixel $\mathbf{x}_t$. $s(\mathbf{x}_{t'})$ is the subset of $S(\mathbf{x}_t)$ in frame t'. $\eta_{s(\mathbf{x}_{t'})}$ and $\Delta_{s(\mathbf{x}_{t'})}$ are the mean coordinate and covariance matrix of $s(\mathbf{x}_{t'})$, respectively.

With the above energy definition, we iteratively refine the segmentation results using belief propagation. Each pass starts from frame 1. While solving the segmentation for frame t, the segment labels of other frames are fixed. After solving the segmentation of frame t, the related volume segments are immediately updated. One pass completes when the segmentation of frame n is optimized. In our experiments, three passes are sufficient to produce spatially and temporally coherent volume segments. One example is shown in Fig. 5. Compared to the initialized volume segments (Fig. 5(b)), the refined volume segments (Fig. 5(c)) become more consistent and better preserve object boundaries.

6 Experimental Results

We experimented with several challenging examples where the videos are taken by a moving camera. Table 1 lists the statistics of the test sequences. The configuration of the parameters in our system is easy. Most parameters are just fixed in our experiments. Specifically, $\delta_v = 0.8$, $\lambda_b = 1.33$, $\varepsilon_b = 0.6$, $\lambda_c = 0.16$, $\varepsilon_c = 0.1$, $\lambda_d = 0.16$, $\varepsilon_d = 0.1(D_{\max} - D_{\min})$. Here, $[D_{\min}, D_{\max}]$ is the disparity range of the scene. For spatial segmentation, we set $w_c = 0.54$, $w_d = 0.1$, $w_s = 0.36$. For spatio-temporal segmentation, we set $w'_h = 0.9$, $w'_c = 0.054$, $w'_d = 0.01$,

Table 1. The statistics of the test sequences.

Sequences	Building	Campus	Road	Stair	Great Wall	Garden	Cones	Teddy
Frames	151	151	141	125	156	135	9	3

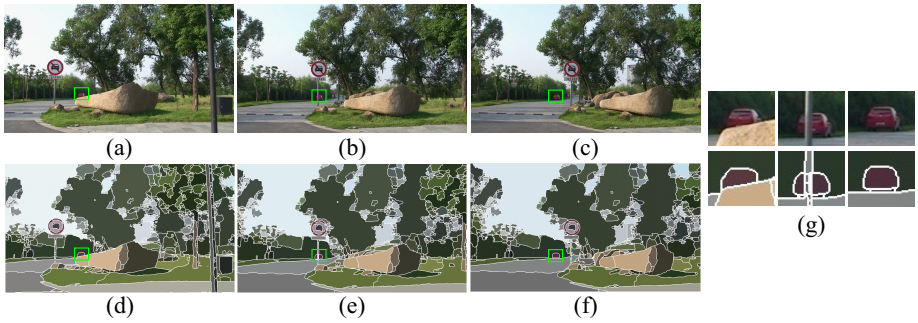

Fig. 6. The segmentation results of "Road" sequence. (a-c) Three selected frames. (d-f) The extracted volume segments. (g) The magnified regions of (a-f).

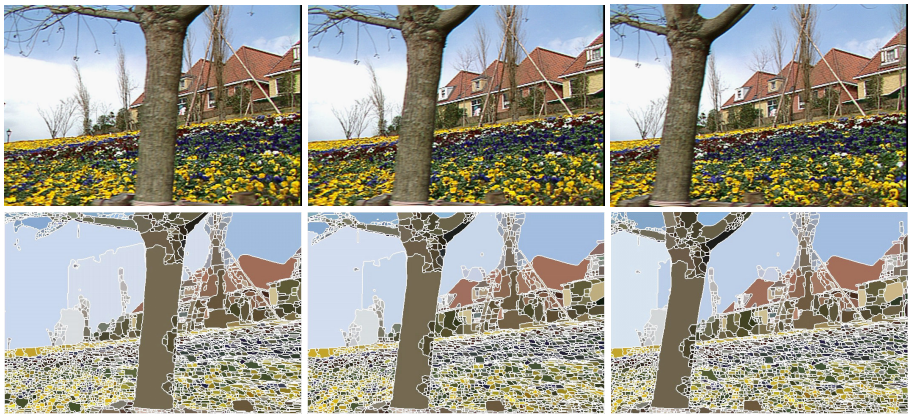

Fig. 7. The segmentation results of "Garden" sequence.

$w_s' = 0.036$. Since mean shift allows the control of segmentation granularity, we can obtain different numbers of volume segments by adjusting the parameters of mean shift in the initialization stage, as shown in Fig. 1.

6.1 Results of Ordinary Video Sequences

Besides the example shown in Fig. 5, we also have experimented with the publicly available 3D video data [4]. One example is shown in Fig. 6. This sequence is very challenging for spatio-temporal segmentation since it contains complex occlusions, where different objects occlude each other and the trees have quite

Fig. 8. The segmentation results of "Building" sequence.

Fig. 9. The segmentation results of a low-frame-rate sequence. Top: three consecutive frames. Bottom: the extracted volume segments represented with unique color.

fractional structures. Our segmentation results faithfully preserve all these structures, and the obtained volume segments are quite consistent in the whole sequence. Our method is also very robust to occlusions. As highlighted in green rectangles, the red car is temporarily occluded by the rocks and thin post of the traffic sign, and even splitted into two separated regions in some frames. Our method successfully recognizes the separated regions and clusters them into the same volume segment.

Figures 7 and 8 show the segmentation results of "Garden" and "Building" sequences, respectively. These two sequences both contain strong image noise,

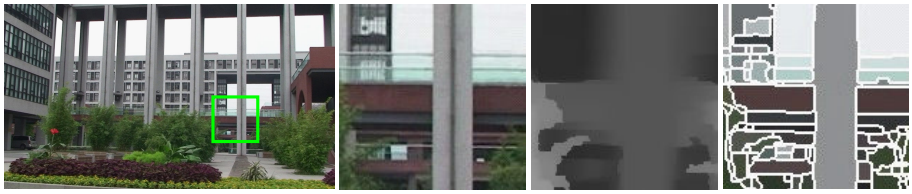

Fig. 10. The segmentation result with imperfect depth data. (a) One selected frame. (b) The magnified region of (a). (c) The estimated depths of (b). (d) The segmentation result.

large textureless sky regions and complex occlusions. The segmentation results demonstrate the robustness of the proposed method. Please refer to our supplementary video for the complete frames and more video examples.

6.2 Results of Low-Frame-Rate Sequences

Though our method is developed to solve the video segmentation problem, it can also handle low-frame-rate sequences that contain a relatively small number of frames with moderately wide baselines between consecutive frames. Although the "Cones" dataset [1] shown in Fig. 9 contains only 9 images, our segmentation results still preserve fine structures and faithfully maintain the coherence in different images. Please refer to our supplementary video for the complete frames.

6.3 Segmentation Results with Imperfect Depth Data

Our segmentation method has moderate tolerance to depth estimation error, as shown in Fig. 10. Although the estimated depths contain noticeable artifacts in this example as shown in Fig. 10(c), the segmentation results still preserve accurate object boundaries and are quite temporally consistent in the whole sequence. The reason is that our method mainly uses the depth information to connect the correspondences among multiple frames and collect the statistics information (such as probabilistic boundaries and the segment probability) for spatio-temporal segmentation, which is more robust than directly using depth information as an additional color channel.

6.4 Results of Challenging Wide-baseline Images

Given an extremely small number of wide-baseline images, the collected statistics may be degraded and handling the problems of large occlusions or out-of-view

[1] This dataset is downloaded from the Middlebury stereo evaluation website [2, 29]: http://vision.middlebury.edu/stereo/data/

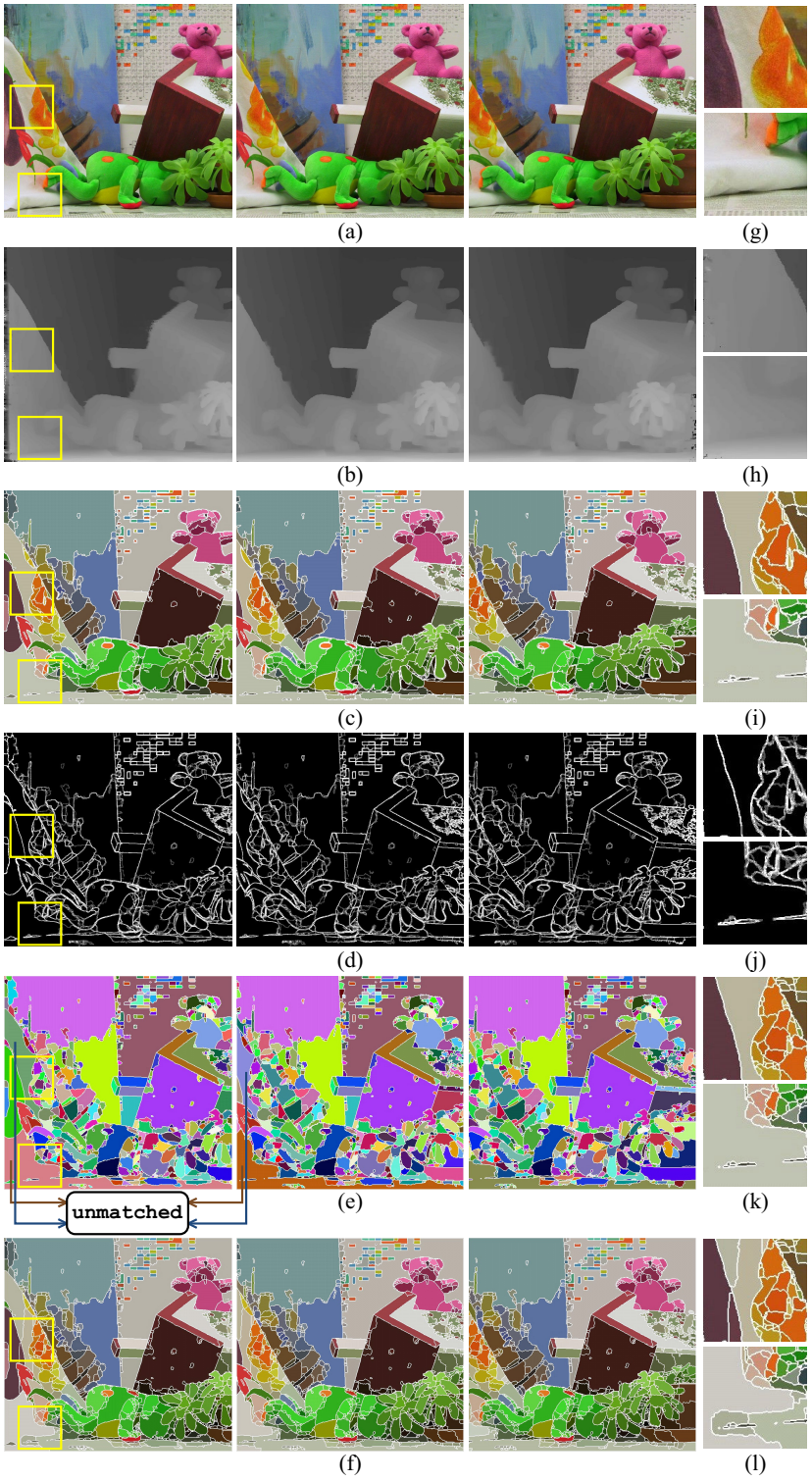

Fig. 11. The segmentation results of "Teddy" sequence. (a) The whole sequence. (b) The depth maps. (c) The mean shift segmentation results. (d) The computed probabilistic boundary maps. (e) Our spatial segmentation results. After segment matching, the matched segments are represented with unique color. (f) Our final spatio-temporal segmentation results. (g-l) The magnified regions of (a-f).

will become more difficult, which may cause our method to produce unsatisfactory segmentation results. Fig. 11 shows a challenging example [2], which only contains 3 wide-baseline images. Fig. 11(b-f) show the depth maps, mean-shift segmentation results, computed probabilistic boundary maps, the spatial segmentation results, and our final spatio-temporal segmentation results, respectively. Even in this extreme case, our segmentation results still preserve most accurate and temporally consistent object boundaries, except for the individual regions highlighted by yellow rectangles in Fig. 11(f). As shown in Fig. 11(e), some segments in different frames correspond to the same object but could not be matched due to the low projection overlapping rate caused by out-of-view, so that multiple volume segments will be produced for the same object. Therefore, the collected segment probability on these regions may have ambiguity, which eventually generates unsatisfactory segmentation results in Fig. 11(l): a uniform region is separated into multiple segments which, however, have similar colors and depths, as can be seen in Fig. 11(g-h).

6.5 Quantitative Evaluation

Our supplementary video already allows a perceptual judgment of the spatio-temporal segmentation results. To further demonstrate the effectiveness of the proposed method, we also use the metrics similar to [30] (i.e., intra-object homogeneity, depth uniformity and temporal stability) to objectively evaluate the quality of our segmentation results. We use the texture variance employed in [30] to measure intra-object homogeneity, and use the projection overlapping rate to measure the temporal stability. All the metrics are normalized to $[0, 1]$, and higher values indicate better results.

We first give the definitions of texture variance and depth uniformity metrics. Both kinds of metrics are normalized to the $[0, 1]$ range, using the following formula employed in [30]:

$$v_n = (\frac{1}{1 + v_m/v_t} - 0.5) \cdot 2, \tag{10}$$

where v_n denotes the normalized metric, v_m is the original metric value, and v_t is a truncation value determined empirically or by the nature of the metric. In our experiments, the truncation values are set to 256 and $0.2(D_{\max} - D_{\min})$ for the texture variance and depth uniformity metrics, respectively.

For video segmentation, v_m is computed by the weighted average metric of all the individual segments:

$$v_m = \sqrt{\sum_{t=1}^{n} \sum_{k=1}^{K_t} w(s_t^k) M(s_t^k)}, \tag{11}$$

[2] This dataset is downloaded from the Middlebury stereo evaluation website [29]: http://vision.middlebury.edu/stereo/data/

Table 2. Quantitative evaluation of different segmentation methods, i.e. Mean shift segmentation (MS), our spatial segmentation with probabilistic boundary (SS) and our iterative spatio-temporal segmentation (STS).

Sequences		Texture Variance	Depth Uniformity	Overlapping Rate
Building	MS	0.93	0.23	69.59%
	SS	0.90	0.64	81.74%
	STS	0.90	0.82	92.14%
Campus	MS	0.88	0.64	65.62%
	SS	0.89	0.78	79.23%
	STS	0.88	0.83	91.69%
Road	MS	0.85	0.84	67.70%
	SS	0.86	0.91	79.35%
	STS	0.84	0.92	92.43%
Stair	MS	0.88	0.78	64.64%
	SS	0.89	0.89	78.24%
	STS	0.88	0.90	91.66%
Great Wall	MS	0.90	0.80	58.69%
	SS	0.91	0.88	71.04%
	STS	0.90	0.89	88.08%
Garden	MS	0.82	0.76	58.96%
	SS	0.82	0.80	61.50%
	STS	0.80	0.81	88.73%
Cones	MS	0.90	0.89	72.95%
	SS	0.91	0.90	88.52%
	STS	0.91	0.90	94.61%

where $w(s_t^k)$ is the weight defined as: $w(s_t^k) = |s_t^k|/V$, where V denotes the number of pixels in the 3D video volume, and $M(s_t^k)$ is the metric value for segment s_t^k. Texture variance metric $M_t(s_t^k)$ is the same as the definition in [30], while depth uniformity metric $M_d(s_t^k)$ collects the statistics of depth boundaries (i.e., depth maps convolved with Sobel operator) contained inside the segment s_t^k, which is defined as:

$$M_d(s_t^k) = \frac{1}{|\Omega(s_t^k)|} \sum_{\mathbf{x}_t \in \Omega(s_t^k)} Sobel(D(\mathbf{x}_t))^2, \tag{12}$$

where $\Omega(s_t^k)$ denotes the interior of s_t^k excluding the boundary.

For measuring temporal stability, we can use the projection overlapping rate as introduced in Section 5.1. The overall projection overlapping rate is computed as follows:

$$\sum_{t=1}^{n} \sum_{k=1}^{K_t} \frac{w(s_t^k)}{|N(t)|} \sum_{t' \in N(t)} \frac{|s_{t \to t'}^k \cap s_{t'}^{k'}|}{\max(|s_{t \to t'}^k|, |s_{t'}^{k'}|)}, \tag{13}$$

where $N(t)$ denotes the neighboring frames of frame t (40 nearest neighboring frames in our experiment). The corresponding segment $s_{t'}^{k'}$ is the one that has

the largest overlapping rate with $s^k_{t\to t'}$, which is determined by the following formula:

$$s^{k'}_{t'} = \arg\max_{s^i_{t'}\in s_{t'}} \frac{|s^k_{t\to t'} \cap s^i_{t'}|}{\max(|s^k_{t\to t'}|, |s^i_{t'}|)}.$$

Table 2 shows the three kinds of metrics on all the test sequences in our paper. For each sequence, we evaluate the results of image-based mean shift segmentation (MS) [6], our spatial segmentation with probabilistic boundary (SS) and our iterative spatio-temporal segmentation (STS). As can be seen, the segmentation results by our spatial segmentation method have comparable texture variance with mean shift, and significantly improve the depth uniformity and temporal stability. After iterative optimization, the temporal stability is further significantly improved.

7 Conclusions and Discussion

In this paper, we have proposed a novel video segmentation method, which can extract a set of spatio-temporal volume segments from a depth-inferred video. Most previous approaches rely on pairwise motion estimation, which are sensitive to large displacement with occlusions. By utilizing depth information, we can connect the correspondences among multiple frames, so that the statistics information, such as probabilistic boundaries and the segment probability of each pixel, can be effectively collected. By incorporating these statistics information into segmentation energy function, our method can robustly handle significant occlusions, so that a set of spatio-temporally consistent segments can be achieved. In practice, the estimated depths are rarely perfect. Fortunately, we have found that our segmentation method has moderate tolerance to depth estimation error, as evidenced in Fig. 10.

Our method still has some limitations. First, we use a single handheld camera and multiview stereo method to recover the depth maps, which is restricted to videos of a static scene. We believe our method can be naturally extended to handle dynamic scenes, since the depths of the moving objects can be recovered by a depth camera or synchronized video cameras. Second, given an extremely small number of wide baseline images, the collected statistics may be degraded, so that extracting spatio-temporally consistent segments will become more difficult. This problem remains to be investigated in our future work.

Acknowledgements

This work is supported by the 973 program of China (No. 2009CB320802), NSF of China (No. 60633070 and 60903135), China Postdoctoral Science Foundation funded project (No. 20100470092), and a research grant from Microsoft Research Asia through the joint lab with Zhejiang University.

References

1. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
2. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision **47** (2002) 7–42
3. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: CVPR. Volume 1. (2006) 519–528
4. Zhang, G., Jia, J., Wong, T.T., Bao, H.: Consistent depth maps recovery from a video sequence. IEEE Transactions on Pattern Analysis and Machine Intelligence **31** (2009) 974–988
5. Kang, S.B., Szeliski, R.: Extracting view-dependent depth maps from a collection of images. International Journal of Computer Vision **58** (2004) 139–163
6. Comaniciu, D., Meer, P., Member, S.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 603–619
7. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 888–905
8. Vincent, L., Soille, P.: Watersheds in digital spaces: An efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991) 583–598
9. Sharon, E., Galun, M., Sharon, D., Basri, R., Brandt, A.: Hierarchy and adaptivity in segmenting visual scenes. Nature **442** (2006) 719–846
10. Megret, R., Dementhon, D.: A survey of spatio-temporal grouping techniques. Technical Report LAMP-TR-094/CS-TR-4403, Language and Media Processing, University of Maryland, College Park (2002)
11. Kumar, M.P., Torr, P.H.S., Zisserman, A.: Learning layered motion segmentations of video. International Journal of Computer Vision **76** (2008) 301–319
12. Shi, J., Malik, J.: Motion segmentation and tracking using normalized cuts. In: ICCV. (1998) 1154–1160
13. Fowlkes, C., Belongie, S., Malik, J.: Efficient spatiotemporal grouping using the Nystrom method. In: CVPR. Volume 1. (2001) 231–238
14. Khan, S., Shah, M.: Object based segmentation of video using color, motion and spatial information. In: CVPR. Volume 2. (2001) 746–750
15. Cremers, D., Soatto, S.: Variational space-time motion segmentation. In: ICCV. (2003) 886–893
16. Zitnick, C.L., Jojic, N., Kang, S.B.: Consistent segmentation for optical flow estimation. In: ICCV. Volume 2. (2005) 1308–1315
17. Deng, Y., Manjunath, B.S.: Unsupervised segmentation of color-texture regions in images and video. IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001) 800–810
18. Brendel, W., Todorovic, S.: Video object segmentation by tracking regions. In: ICCV. (2009)
19. Liu, S., Dong, G., Yan, C.H., Ong, S.H.: Video segmentation: Propagation, validation and aggregation of a preceding graph. In: CVPR. (2008) 1–7
20. Wang, Y., Ji, Q.: A dynamic conditional random field model for object segmentation in image sequences. In: CVPR. Volume 1. (2005) 264–270

21. Dementhon, D., Megret, R.: Spatio-temporal segmentation of video by hierarchical mean shift analysis. Technical Report LAMP-TR-090/CAR-TR-978/CS-TR-4388/UMIACS-TR-2002-68, University of Maryland, College Park (2002)
22. Wang, J., Thiesson, B., Xu, Y., Cohen, M.: Image and video segmentation by anisotropic kernel mean shift. In: ECCV. (2004) 238–249
23. Greenspan, H., Goldberger, J., Mayer, A.: Probabilistic space-time video modeling via piecewise GMM. IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (2004) 384–396
24. Quan, L., Wang, J., Tan, P., Yuan, L.: Image-based modeling by joint segmentation. International Journal of Computer Vision **75** (2007) 135–150
25. Xiao, J., Wang, J., Tan, P., Quan, L.: Joint affinity propagation for multiple view segmentation. In: ICCV. (2007) 1–7
26. Zhang, G., Qin, X., Hua, W., Wong, T.T., Heng, P.A., Bao, H.: Robust metric reconstruction from challenging video sequences. In: CVPR. (2007)
27. Smet, P.D., Pires, R.L.V.P.M.: Implementation and analysis of an optimized rainfalling watershed algorithm. In: IS&TSPIE's 12th Annual Symposium Electronic Imaging. (2000) 759–766
28. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. International Journal of Computer Vision **70** (2006) 41–54
29. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: CVPR. Volume 1. (2003) 195–202
30. Correia, P.L., Pereira, F.: Objective evaluation of video segmentation quality. IEEE Transactions on Image Processing **12** (2003) 186–200