Automatic video segmentation using spatiotemporal T-junctions

Nicholas Apostoloff
University of Oxford
http://www.robots.ox.ac.uk/~vgg

Andrew Fitzgibbon
Microsoft Research Ltd.
http://research.microsoft.com/mlp

Abstract

The problem of figure–ground segmentation is of great importance in both video editing and visual perception tasks. Classical video segmentation algorithms approach the problem from one of two perspectives. At one extreme, *global* approaches constrain the camera motion to simplify the image structure. At the other extreme, *local* approaches estimate motion in small image regions over a small number of frames and tend to produce noisy signals that are difficult to segment. With recent advances in image segmentation showing that sparse information is often sufficient for figure–ground segmentation it seems surprising then that with the extra temporal information of video, an unconstrained automatic figure–ground segmentation algorithm still eludes the research community. In this paper we present an automatic video segmentation algorithm that is intermediate between these two extremes and uses spatiotemporal features to regularize the segmentation. Detecting spatiotemporal T-junctions that indicate occlusion edges, we learn an occlusion edge model that is used within a colour contrast sensitive MRF to segment individual frames of a video sequence. T-junctions are learnt and classified using a support vector machine and a Gaussian mixture model is fitted to the (foreground, background) pixel pairs sampled from the detected T-junctions. Graph cut is then used to segment each frame of the video showing that sparse occlusion edge information can automatically initialize the video segmentation problem.

1 Introduction

Video segmentation, or layer extraction, is a classic inverse problem in computer vision that involves the extraction of foreground objects from a set of images [4, 17, 33]. In image segmentation the goal is to segment an image into *spatially* coherent regions, whereas in video segmentation the goal is segment the image into *temporally* coherent regions. In both situations, the coherence is typically broken by occlusion edges and accurate detection of these occlusion boundaries is essential.

Traditionally, video segmentation algorithms are problematic because they generally lie at the extreme of possible approaches. At one extreme, *global* approaches constrain the camera motion to simplify the spatiotemporal image structure and generally work well because the object motion and occlusion detection are mathematically well defined. For example, epipolar plane image (EPI) analysis assumes constant horizontal camera motion such that the spatiotemporal image structure consists of a set of straight lines whose gradients depend on the depth of objects in the scene [5, 8]. Junctions between these lines

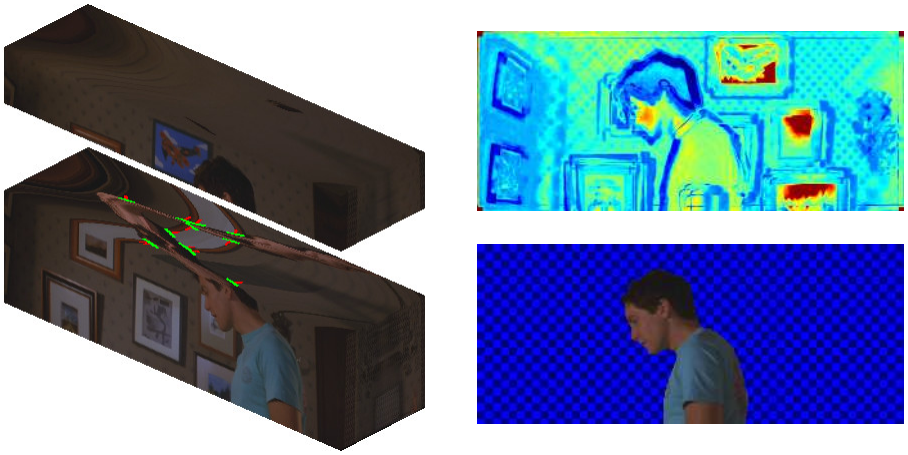
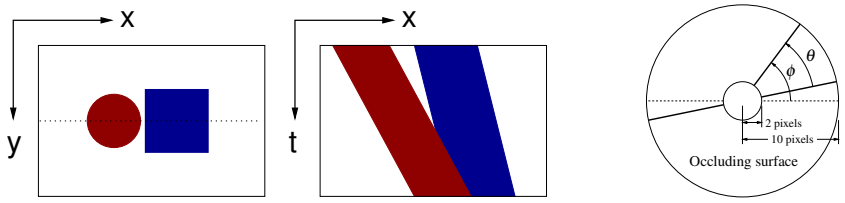


Figure 1: Automatic video segmentation using spatiotemporal T-junctions. The spatiotemporal T-junctions detected on the space-time slices (left) are used to learn the occlusion edge energy (top-right), where blue is low and red is high energy, and segment each frame using graph cuts (bottom-right).

are then indicators of occlusion as they result from objects at different depths moving past each other (figure 2). In particular, a T-junction is formed where the occluding object is above the hat of the T and the background forms the stem of the T. At the other extreme, *local* approaches estimate motion in small image regions over two or three frames for typically arbitrary camera motion (e.g. optical flow [15]). While less constrained, local approaches produce noisy signals that can be difficult to segment, particularly at object boundaries. In this paper we combine local and global paradigms and assume locally linear camera motion such that the resulting spatiotemporal image structure is EPI-like. We use a variant of the spatiotemporal T-junction detector of Apostoloff and Fitzgibbon [3] to learn the appearance of occlusion edges that can be used to automatically initialize the segmentation problem. This work is motivated by recent advances in single image alpha matting where it has been shown that sparse information in the form of user brush strokes can be sufficient for figure–ground segmentation in many situations [30]. These results have been extended to video sequences, but to date have required considerable user interaction [32]. We show in this paper that sparse local information in the form of spatiotemporal T-junctions can remove the requirement for user interaction in many cases.

Key features of this paper are threefold: first, we approach the problem of motion segmentation from a viewpoint intermediate between the local and global extremes. Our assumption of *locally linear* camera motion allows EPI-like images to be searched for spatiotemporal T-junctions. The spatiotemporal T-junction detector of Apostoloff and Fitzgibbon [3] is improved through the use of a support vector machine. Second, in contrast to previous efforts in this area, we learn prior distributions on pairs of (foreground, background) colours spanning occlusion edges instead of modelling the foreground and background distributions separately. This means that strong occlusion edge terms can be used within a Markov random field (MRF) to regularize the problem. Finally, we solve the segmentation using a novel graph cut MRF that combines contrast, colour and occlusion edge terms to give a global solution for the segmentation of each frame in a video sequence.



(a) Spatial image with the circle in front of the square

(b) EPI image where the camera is moving to the left

(c) The T-junction template is defined by θ and ϕ, while the central region is ignored.

Figure 2: The epipolar plane image (a and b) and the T-junction template (c) used for modelling edge occlusion events.

2 Previous approaches to video segmentation

Global approaches to motion analysis based on the epipolar plane image were pioneered by Bolles and Baker who used it to build a 3D description of a static scene [9]. They constrained the problem to constant horizontal camera motion, which ensures that image features stay in the same scanline and move continuously over the image sequence. Hence, an XT slice[1] of a complex scene reduces to an image with a set of straight edges whose gradient is relative to their depth in the scene (figure 2). In this case, the termination of an edge by another edge is an indicator of occlusion and presents a similar profile to the T-junction in the spatial domain. This technique was shown to be quite powerful at determining visible 3D structure within a scene, but was inherently limited to constant horizontal camera motion. They extended this work to include arbitrary camera *rotation* by working in the dual-space of cylindrical epipolar plane coordinates [5]. Feldmann *et al*. relax the constraints of EPI analysis to circular camera movements by defining a set of trajectories that define the depth of a point within an image volume [13]. Niyogi and Adelson observed that walkers generate a helix-like signature in space-time and exploit this characteristic to detect and model a persons gait from a stationary camera [26, 27]. Later Niyogi analyzed kinetic occlusion in space-time using a layering framework and a motion energy scheme adapted from models of biological vision [24]. In a slightly less restrictive approach Criminisi *et al*. exploit the structure within epipolar plane to detect occluding edges for a dense reconstruction of the scene [12].

At the other extreme, local approaches make no assumptions on the camera path and the motion is estimated locally over a small number of frames. A classic example of this is optical flow where spatiotemporal image derivatives are calculated to estimate a velocity field that adheres to the brightness consistency constraint [15]. Because these methods are local and based on small regions that carry little information, they are often inaccurate and noisy. To compensate for this, constraints to smooth the velocity fields spatially [2, 15] and temporally [7] for segmentation are used. These global constraints either limit the range of camera motion like EPI or introduce artificial smoothing that is highly inaccurate at motion boundaries.

Early work by Wang and Adelson [33] and Irani *et al*. [17] showed how optic flow information could be used to automatically extract layers from an image sequence. Irani *et al*. use temporal integration to localize and track moving objects by registering frames by the

[1] An XT or spatiotemporal image is a slice through the volume of images at a constant scanline.

dominant motion, but are limited to tracking non-articulated objects. Later, Irani showed that flow fields of a rigid scene reside in a low-dimensional subspace and constrained the flow field to reduce the noise in the estimate [16]. The flexible sprites approach of Jojic and Frey automatically learns multiple layers using probabilistic 2D appearance maps in an expectation maximization (EM) framework [19], but is limited to stationary camera scenarios. Niyogi, Adelson and Bergen [1, 25] also present methods to detect motion boundaries using oriented spatiotemporal energy models that detect surface texture accretion and deletion.

More recently, Xiao and Shah employ graph cut over spatiotemporal volumes to obtain a motion-based segmentation and derive the General Occlusion Constraint to solve for foreground, background and occlusion segmentations [35]. Further advances in video matting and segmentation have attempted to reduce user interaction, not remove it. Interactive video cutout by Wang and Cohen extends a frame-wise 2D colour over-segmentation over time using graph cut and user interaction through a novel volumetric painting interface [34]. The video cut and paste algorithm by Li *et al*. over-segments using colour in 2D and then propagates the segmentation through key-frames with graph cut [22].

The main contribution of this paper is an automatic video segmentation algorithm that learns the appearance of occlusion edges from spatiotemporal T-junctions and proceeds in three main steps (figure 1). First, spatiotemporal T-junctions are detected in every scanline slice of the video cube. Second, (foreground, background) pixel pairs are sampled from each T-junction and an occlusion edge Gaussian mixture model (GMM) is learnt that models the transition from foreground to background. Finally, each frame is segmented separately using graph cut with an MRF defined by both a contrast sensitive smoothing term and the learnt occlusion edge term. This paper is structured as follows. First, we overview the spatiotemporal T-junction detector used to learn the appearance of occlusion edges. We then present the segmentation framework with a description of the occlusion edge model and conclude with a discussion of the results.

3 Occlusion detection using spatiotemporal T-junctions

A natural indicator of occlusion is the T-junction—a photometric profile shaped like a "T", which is formed where the edge of an object occludes a change in intensity in the background. Until recently, there have been two predominant approaches to T-junction detection: gradient or filter based approaches, and model-based template matching. Gradient based methods assume that there is a distinct gradient profile in a region close to a junction [6, 29]. Model based methods approach the problem from a top-down perspective by fitting an explicit junction model at hypothesized junction locations [28]. Unfortunately, single images can produce many false T-junctions that do not lie on an occlusion edge; however, it is also known that T-junctions in the *spatiotemporal* domain are strong indicators of occlusion events [8], and we use the recent spatiotemporal T-junction detector of Apostoloff and Fitzgibbon [3]. They learn the appearance of spatiotemporal T-junctions using oriented SIFT [23] descriptors and a relevance vector machine (RVM) with a linear kernel [31].

T-junction detection proceeds as follows. First, the search is seeded with Harris corners [14] in each spatiotemporal slice of the video sequence. A T-junction template is then fitted to each Harris corner that aligns the dominate gradients with the T-junction model (figure 2). A SIFT descriptor is calculated at the oriented T-junction which captures localized gradient histogram responses and is then classified using the RVM. The RVM is a subset of sparse Bayesian learning methods [31] that uses linear kernel mod-

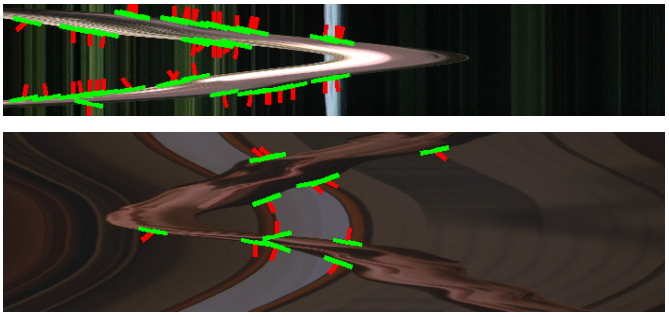


Figure 3: Results: two frames of spatiotemporal T-junction detection. These spatiotemporal slices are from two image sequences where the actor moves into and then out of the shot with a stationary camera (top) and a moving camera (bottom). Green lines are the occlusion edge while red lines point towards the background.

els similar to the support vector machine (SVM) for classification [18]. It is during the learning phase that an RVM differs significantly from an SVM and in this case where the performance of the RVM suffers. As the learning phase of the RVM requires inverting a dense MxM matrix (M being the number of training examples) it is inherently limited to a small training set. The SVM on the other hand has no such limitation and all the training examples can be used to learn the SVM. Using the SVM with over 5000 hand-labelled training examples, we are able to select a lower classification threshold and obtain significantly more T-junctions at little cost to the speed or accuracy of the detector. Figure 3 shows the detection of T-junctions in two different spatiotemporal images.

4 Image segmentation using graph cuts

First, we provide an overview of the image segmentation algorithm of Boykov and Jolly as it is the basis upon which we segment each frame [10].

Their algorithm segments a greyscale image C given a trimap T, which separates pixels into foreground, background and unknown values. Given the array of N composite pixels $\mathbf{c} = (c_1, ..., c_N)$, the segmentation of the image is expressed as the array $\underline{\alpha} = (\alpha_1, ..., \alpha_N)$ of "opacity" values at each pixel, where generally $0 \leq \alpha \leq 1$; however, in the case of "hard" segmentation $\alpha \in \{0, 1\}$ with 0 for background and 1 for foreground.

Segmentation proceeds by first defining an energy equation that contains data terms modelling the potential for a pixel to be foreground or background, U, and pairwise potentials between pixels that reflect the tendency for neighbouring pixels to be alike, V. This is given in the form of the Gibbs energy:

$$\mathbf{E}(\underline{\alpha}, \Theta, \mathbf{c}) = U(\underline{\alpha}, \Theta, \mathbf{c}) + V(\underline{\alpha}, \mathbf{c}) \tag{1}$$

where Θ defines a parametric model of the foreground and background pixel distributions. In the work of Boykov and Jolly, Θ represents two normalized histograms that describe the foreground and background greyscale pixel distributions in the trimap image and are defined as $\Theta = \{h(c; \alpha), \alpha \in \{0, 1\}\}$. The data term U thus becomes

$$U(\underline{\alpha}, \Theta, \mathbf{c}) = \sum_{n=1}^{N} D(\alpha_n, \Theta, c_n) \tag{2}$$

where $D(\alpha_n, \Theta, c_n) = -\log(h(c_n, \alpha_n))$.

The smoothness term V defines an 8-connected MRF with contrast sensitive edges:

$$V(\underline{\alpha},\mathbf{c}) = \gamma_1 \sum_{(i,j)\in\mathbf{P}} \text{euc}(i,j)^{-1}[\alpha_i \neq \alpha_j]\exp\{-(c_i-c_j)^2/2\sigma^2\} \tag{3}$$

where $\mathbf{P}$ is the subset of all 8-connected pixel edges in the image, $\text{euc}(i,j)$ is the Euclidean distance between pixels i and j, the weight γ_1 was set to 50 and σ is derived from the mean of the image gradient: $\sigma^2 = < ||c_i - c_j||^2 >$.

Having defined the energy model, they then use the minimum cut algorithm [10, 20] to minimize the energy function and obtain a segmentation. This energy minimization forms the basis of our hard segmentation framework; however, it differs in three ways. First, like the method of Rother *et al*. we extend the algorithm to colour images and model the foreground and background pixel distributions with Gaussian mixture models (GMM). Second, we remove user interaction completely by learning the colour models from the spatiotemporal T-junctions. Third, we add an additional term to the energy equation that encapsulates the occlusion edge statistics that are learnt from the spatiotemporal T-junctions.

5 Video segmentation using learnt occlusion edges

5.1 Modelling colour

In a similar fashion to Rother *et al*., we extend the segmentation algorithm of Boykov and Jolly to colour images and model the foreground and background pixel distributions with Gaussian mixture models instead of greyscale histograms.

Now, $\mathbf{c}$ is the array of colour pixels, and the foreground and background parametric models become RGB GMMs such that $D(\alpha_n,\Theta,c_n) = -\log(p(c_n|\alpha_n,\Theta))$ and $p(\bullet|\alpha_n,\Theta)$ is the foreground or background GMM when α_n is 1 or 0 respectively (unlike Rother *et al*. who model each pixel as being drawn from a single foreground and background Gaussian):

$$p(c_n|\alpha_n,\Theta) = \sum_k \pi(k,\alpha_n)\frac{\exp(-(c_n-\mu(k,\alpha_n))^\top \Sigma(k,\alpha_n)^{-1}(c_n-\mu(k,\alpha_n))/2)}{\sqrt{(2\pi)^3|\Sigma(k,\alpha_n)|}} \tag{4}$$

where $\pi(k,\alpha_n)$, $\mu(k,\alpha_n)$ and $\Sigma(k,\alpha_n)$ are the mixing coefficient, mean and covariance matrix of cluster k from the foreground or background GMM (depending on the value of α_n) and are contained in the parameterization Θ.

The smoothness term V remains essentially unchanged as

$$V(\underline{\alpha},\mathbf{c}) = \gamma_1 \sum_{(i,j)\in\mathbf{P}} \text{euc}(i,j)^{-1}[\alpha_i \neq \alpha_j]\exp\{-||c_i-c_j||^2/2\sigma^2\}. \tag{5}$$

5.2 Modelling occlusion edges

As figure 4 shows, the contrast sensitive smoothness term of equation 5 is responds to all image edges and can cause unwanted noise in the final segmentation. To mitigate this we include an additional pairwise potential between neighbouring pixels that discourages transitions from foreground to background or vice versa if node colours do not match the learnt occlusion edge model. The energy now becomes

$$\mathbf{E}(\alpha,\Theta,\mathbf{c}) = U(\underline{\alpha},\Theta,\mathbf{c}) + V(\underline{\alpha},\mathbf{c}) + W(\underline{\alpha},\Theta,\mathbf{c}) \tag{6}$$

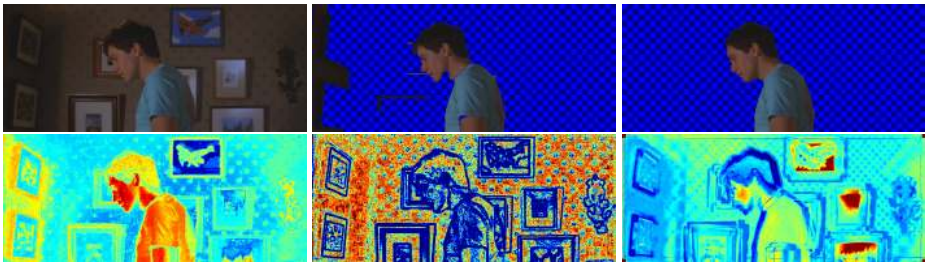


Figure 4: Modelling occlusion edges. From left to right the first row contains the original image, the segmentation using only the data and contrast terms and the segmentation with all terms. The second row contains the data term (red corresponds to high foreground probability and blue to high background probability), the contrast term and the occlusion edge term. Notice that the errors in the first segmentation are corrected with the addition of the occlusion edge term since edges that do not match the occlusion edge colour profiles are discouraged.

where W is similar in form to V in that it operates over all 8-connected edges in the image; however, it is no longer undirected and is learnt from the detected T-junctions

$$W(\underline{\alpha},\Theta,\mathbf{c}) = \sum_{(i,j)\in\mathbf{P}} \text{euc}(i,j)^{-1}\{-[\alpha_i == 1\,\&\,\alpha_j == 0]\log\left(p_{FB}\left(c_i,c_j|\Theta\right)\right) \\ -[\alpha_i == 0\,\&\,\alpha_j == 1]\log\left(p_{FB}\left(c_j,c_i|\Theta\right)\right)\} + K$$

where K is a constant that ensures all the summed terms are positive (≈ 15 in most cases) while the first term models foreground to background transitions and the second term accounts for background to foreground transitions. This effectively discourages edges that do not match the learnt model but allows the contrast term to determine the precise location of the edge.[2]

The distribution $p_{FB}(c_i,c_j|\Theta)$ is modelled as a 6D GMM and is learnt from (foreground, background) RGB pixel pairs sampled from the spatiotemporal T-junctions.

6 Learning occlusion edges

From each spatiotemporal T-junction, we extract all pixels in a 10 pixel radius of the T-junction that are a minimum of 2 pixels away from the occluding edge (to mitigate alignment errors and pixel blending). The pixels on the occlusion surface are labelled as foreground F and are paired with the background pixels B opposite them over the occluding edge to form a 6D vector $\begin{bmatrix}F\\B\end{bmatrix}$. We then learn a 6D Gaussian mixture model using variational Bayesian model selection [11] initializing the algorithm with 20 Gaussians, each parameterized by the covariance Σ and mean μ

$$\Sigma = \begin{bmatrix}\Sigma_F & \Sigma_{FB}\\ \Sigma_{BF} & \Sigma_B\end{bmatrix};\ \mu = \begin{bmatrix}\mu_F\\ \mu_B\end{bmatrix}. \tag{7}$$

[2]To mitigate the effect of pixel blending at edges we initially used long range edges within the graph that stretched over 2 and 3 pixels instead of the 8-connected neighbours; however, this led to a chequerboard pattern appearing in the result. As a result, we break the strict independence of the MRF and model the colour at pixels i and j by the colours at 3 pixels beyond them in the same direction as the edge joining them.

Figure 5: Results: learnt occlusion edge models. Sampling pixels from either side of the T-junctions detected in the input sequence (left and middle) produces the GMM occlusion edge model shown on the right. Each column is a single 6D Gaussian with the first row sampled from the foreground marginal and the second row sampled from the background marginal. The width of each column is proportional to the mixing coefficient of that Gaussian.

The parameters μ_F, Σ_F, μ_B and Σ_B are also used as the mean and covariances of the foreground and background colour models respectively. Furthermore, the covariances between F and B (Σ_{FB} and Σ_{BF}) are set to zero to ensure that the occlusion edge energy function W is graph representable [21].[3] For the sequence in figure 5 we can see the learnt (foreground, background) pairings. For example, column 2 in the right images shows the learnt occlusion edge transition from the actor's dark brown hair to the blue of the picture in the background, while column 8 shows the transition from his light blue shirt to the brown background.

7 Results and conclusions

Figure 6 shows two simple scenarios with a stationary camera and an actor moving into and then out of the field of view, and a third scenario with a moving camera and a moving actor. Almost perfect segmentation is achieved in the stationary camera scenarios even though the background contains many sharp edges that can degrade segmentation performance. The third scenario is particularly difficult given that the background contains many objects of a similar colour to the foreground; however, the algorithm obtains a satisfactory segmentation for most frames.

We have shown that the colour information learnt by detecting spatiotemporal T-junctions can segment a number of simple video sequences and that by modelling explicitly the occlusion edge transitions, erroneous objects can be removed from the final segmentation. Of course, the current implementation is limited to sequences where the dominant motion is horizontal; however, we have implemented a straightforward extension that first computes the camera trajectory over short sub-sequences, approximates it by a translation, and then rectifies the images such that it is horizontal.

Although this paper presented a simple model of occlusion edges, they were shown to be powerful enough to regularize the segmentation in many cases. Further extensions to improve performance could include more complicated models of edge transitions such as the patch-based approaches that are becoming popular in image-based prior methods.

The authors would like to acknowledge Carsten Rother for his helpful insights on this subject.

[3] A pairwise energy function E of two binary variables is regular and hence graph representable if $E(0,0) + E(1,1) \leq E(0,1) + E(1,0)$. This is a necessary condition to ensure that the global energy function can be minimized *exactly*.

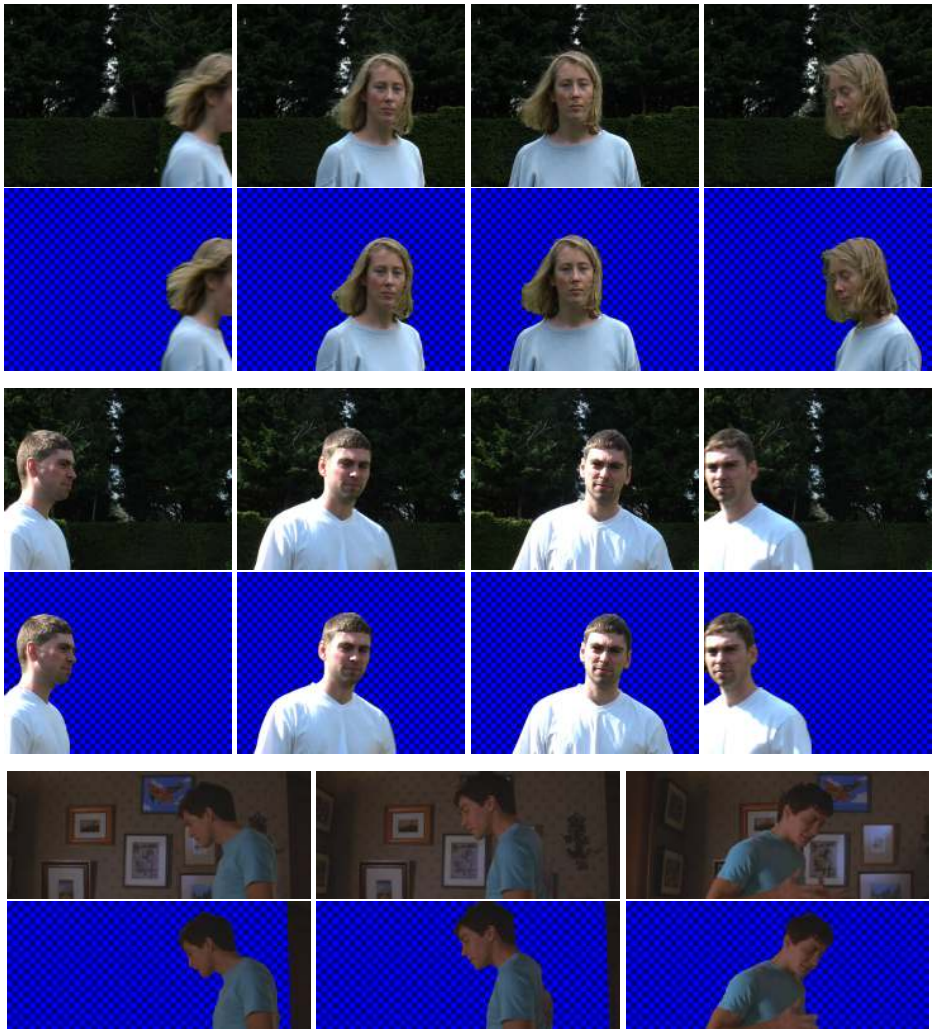


Figure 6: Results: fully automatic video segmentation. The first two sequences have a stationary camera and moving actors while the third has a moving camera and a moving actor.

References

[1] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am. A*, 2(2):284–299, 1985.

[2] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *IJCV*, 2:283–310, 1989.

[3] N. Apostoloff and A. W. Fitzgibbon. Learning spatiotemporal T-junctions for occlusion detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 553–559, San Diego, 2005.

[4] S. Ayer and H. S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Proc. ICCV*, pages 777–783, 1995.

[5] H. H. Baker and R. C. Bolles. Generalizing epipolar-plane image analysis on the spatiotemporal surface. *IJCV*, 3(1):33–49, May 1989.

[6] D. Beymer. Finding junctions using the image gradient. In *Proc. IEEE Computer Vision and Pattern Recognition*, 1991.

[7] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. CVPR*, pages 296–302, 1991.

[8] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *IJCV*, 1(1):7–56, 1987.

[9] R. C. Bolles and R. Horaud. 3DPO: A three-dimensional part orientation system. In T. Kanade, editor, *Three Dimensional Vision*, pages 399–450. Kluwer Academic Publishers, 1987.

[10] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proc. IEEE Int. Conf. on Computer Vision*, 2001.

[11] A. Corduneanu and C. M. Bishop. Variational bayesian model selection for mixture distributions. In *Proceedings Eighth International Conference on Artificial Intelligence and Statistics*, pages 27–34, 2001.

[12] A. Criminisi, S. B. Kang, R. Swaminathan, R. Szeliski, and R. Anandan. Extracting layers and analyzing their specular properties using epipolar-plane-image analysis. Technical Report 44, Microsoft Research, 2002.

[13] I. Feldmann, P. Eisert, and P. Kauff. Extension of epipolar image analysis to circular camera movement. In *Proc. of Int. Conf. on Image Processing (ICIP 2003)*, Barcelona, Spain, September 2003.

[14] C. J. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Alvey Vision Conf.*, pages 147–151, 1988.

[15] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[16] M. Irani. Multi-frame optical flow estimation using subspace constraints. In *Proc. ICCV*, pages 626–633, 1999.

[17] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *IJCV*, 12(1):5–16, 1994.

[18] T. Joachims. Making large-scale support vector machine learning practical. In A. Smola B. Schölkopf, C. Burges, editor, *Advances in Kernel Methods: Support Vector Machines*. MIT Press, Cambridge, MA, 1998.

[19] N. Jojic and B. Frey. Learning flexible sprites in video layers. In *Proc. CVPR*, 2001.

[20] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *Proc. ECCV*, 2002.

[21] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. PAMI*, 26:147–159, 2004.

[22] Y. Li, J. Sun, and H. Shum. Video cut and paste. In *Proc. ACM SIGGRAPH*, 2005.

[23] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, pages 1150–1157, 1999.

[24] S. Niyogi. Detecting kinetic occlusion. In *Proc. ICCV*, pages 1044–1049, 1995.

[25] S. Niyogi. Spatiotemporal junction analysis for motion boundary detection. In *ICIP*, pages 468–471, 1995.

[26] S. Niyogi and E. H. Adelson. Analyzing and recognizing walking figures in XYT. Technical Report 223, M.I.T. Media Lab, Vision and Modeling Group, 1993.

[27] S. Niyogi and E. H. Adelson. Analyzing gait with spatiotemporal surfaces. In *orkshop on Non-Rigid Motion and Articulated Objects*, Austin, TX, November 1994.

[28] L. Parida and D. Geiger. Junctions: detection, classification and reconstruction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(7):687–698, 1998.

[29] P. Perona. Steerable-scalable kernels for edge detection and junction analysis. In *European Conference on Computer Vision*, pages 3–18, 1992.

[30] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004.

[31] M. E. Tipping. The relevance vector machine. *Advances in Neural Information Processing Systems*, 12:652–658, 2000.

[32] J Wang, P. Bhat, A. Colburn, M. Agrawala, and M. F. Cohen. Interactive video cutout. *ACM Trans. Graph.*, 24(3):585–594, 2005.

[33] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *The IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638, September 1994.

[34] Jue Wang and Michael F. Cohen. An iterative optimization approach for unified image segmentation and matting. In *ICCV*, pages 936–943, 2005.

[35] J. Xiao and M Shah. Accurate motion layer segmentation and matting. In *Proc. CVPR*, San Diego, June 2005.