Graph cuts with shape priors for segmentation

Mayuresh Kulkarni
University of Cape Town
Cape Town, South Africa
Email: mayuresh.kulkarni@uct.ac.za

Fred Nicolls
University of Cape Town
Cape Town, South Africa
Email: fred.nicolls@uct.ac.za

***Abstract*—This paper investigates segmentation of images and videos using graph cuts and shape priors. Graph cuts is used to find the global optimum of a cost function based on the region and boundary properties of the image or video. The region and boundary properties are estimated using certain pixels marked by the user. A shape prior term is added to this cost function to bias the solution towards a known shape. In this work, a circular shape prior defined by center and radius parameters is used. Powell's minimization algorithm is used to align the shape prior with the object to be segmented. The average location of the user-marked pixels is used as a starting point to initialize Powell's method. Accurate image and video segmentations are achieved with minimal user input. The results obtained when including shape priors are compared to those using just the region and boundary properties in the graph cut. Although only a circular prior is used in this work, the concepts can be extended to any parametric shape prior that determines the shape of the desired object. In this paper, graph cuts and shape priors are used to segment faces from images and videos.**

I. Introduction

Segmentation is the extraction of regions of interest from images. Fully automatic segmentation has inherent problems associated with it. This paper focuses on interactive image and video segmentation into 'foreground' and 'background'.

In images, the user marks certain pixels as 'foreground' and 'background', also known as *seeds*. Seeds are used as hard constraints for the segmentation. Hard constraints provide the clues to the desired segmentation. A graph is set up using each pixel as a node. Each pixel or node is connected to adjacent pixels in all directions to define the edges. A cost function based on region and boundary properties is defined. Region weights are estimated using the properties of the hard constraints using Gaussian Mixture Models (GMMs). Colour and texture features are used as components of the GMMs. The probability of each pixel being either 'foreground' or 'background' can be estimated using the logarithmic likelihood ratio. Edge detection methods are used to find the evidence of a boundary in each pixel in the image. A globally optimal solution is calculated using soft and hard constraints. The segmentation process can be made iterative to get the desired result. A globally optimal segmentation can be efficiently recalculated when the user adds or removes hard constraints at each iteration.

Intensity, colour and texture properties are used as features in GMMs to assign soft constraints on pixels. Different colour schemes like RGB and Luv are used to overcome the drawbacks of any single scheme. Combinations of colour and texture are used to analyse the best features for region weights. Edge detection methods like Canny edge detector, gradient methods and a GMM-based edge model are used to set edge weights.

Shape priors are added to the region and boundary properties in the cost function to improve segmentation. A circular shape prior defined using the center and the radius is used. The shape prior is aligned to the object in the image using Powell's [7] minimization algorithm to get the minimum over minimum cuts of the graph. The average location of the seeds is used as an initial guess for Powell's method. A weighted distance transform from the shape is used to weigh the edges in the graph. The pixels closer to shape prior are assigned a lower cost which increases the probability of classifying them as foreground. Shape priors and graph cuts are also used for video segmentation using a 26-voxel neighbourhood.

Section II provides a detailed literature review of image and video segmentation related to this paper. The details of the implementation of the algorithm are discussed in Section III. The results for images and videos are discussed in Section IV. Segmentations resulting from different methods are compared to the methods in [2]. Section V derives conclusions from the work done and provides suggestions for future research.

II. Related Work

A. Segmentation using graph cuts

The graph cut method is a popular and powerful technique for image segmentation. It can be modified to fit certain problems where there is specific knowledge about the object to be segmented. For example, if the shape of the object to be segmented is known, then this information can be used to direct graph cuts to segment images accordingly.

Boykov and Jolly [6] use interactive graph cuts for region- and boundary-based image segmentation. Globally optimal segmentation is achieved using the cost function with hard constraints imposed by the user. The segmentation process is made interactive so that the segmentation desired by the user can be obtained. Applications of graph cuts for video and medical image segmentation are given. Assuming that $\mathcal{O}$ and $\mathcal{B}$ denote pixels marked by the user as object ("OBJ") and background ("BKG") the weights of the edges are assigned as follows:

TABLE I
ASSIGNMENT OF EDGE WEIGHTS IN BOYKOV AND JOLLY [6].

edge	weight (cost)	condition
$\{p,q\}$	$B_{\{p,q\}}$	$\{p,q\} \in \mathcal{N}$
$\{p,S\}$	$\lambda \cdot R_p(\text{“}bkg\text{”})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
	K	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
$\{p,T\}$	$\lambda \cdot R_p(\text{“}obj\text{”})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
	0	$p \in \mathcal{O}$
	K	$p \in \mathcal{B}$

where

$$K = 1 + \max_{p \epsilon \mathcal{P}} \sum_{q:\{p,q\} \epsilon \mathcal{N}} B_{\{p,q\}} \tag{1}$$

and λ is the weighting factor between regions and boundaries in the cost function. The source and sink nodes are represented using S and T respectively. The cost function is described as

$$E(A) = \lambda \cdot R(A) + B(A) \tag{2}$$

where

$$R(A) = \sum_{p \epsilon \mathcal{P}} R_p(A_p), \tag{3}$$

$$B(A) = \sum_{\{p,q\} \epsilon \mathcal{N}} B_{\{p,q\}} \cdot \delta(A_p, A_q), \tag{4}$$

and

$$\delta(A_p, A_q) = \begin{cases} 1 & \text{if } A_p \neq A_q, \\ 0 & \text{otherwise.} \end{cases}$$

The pixels marked as object or background by the user are hard constraints on the segmentation. Region and boundary properties are determined based on these hard constraints to assign soft constraints.

The region term $R(A)$ reflects how well a pixel p fits into object or background model based on region properties like colour, intensity or texture. $B(A)$ term describes the boundary properties of the image. $B_{\{p,q\}}$ can be interpreted as the evidence of a boundary between two neighbouring pixels p and q. In equation (2), λ is a coefficient that shows the weight given to region properties $R(A)$ with respect to the boundary properties $B(A)$. A similar graph structure is used in this paper, but different methods are used to estimate edge weights in this paper. A fast implementation of this algorithm is described by Boykov and Kolmogorov [8].

The problem of effective, interactive foreground/background segmentation is also investigated in GrabCut [10]. Colour data is modeled using GMMs to estimate foreground and background probabilities of each pixel. The main aim of GrabCut [10] is to reduce user interaction by using techniques called “iterative estimation” and “incomplete labeling”. GrabCut begins with the user drawing a rectangle around the desired object. Foreground statistics are estimated using the pixel data in the rectangle. A segmentation using graph cuts is done and the user is allowed to add background, foreground or matting information to improve the segmentation. Matting information is border information that is used to recover foreground colour information, free of colour bleeding from the background. “Incomplete labeling” enables the user to only mark background pixels. There is no need to mark foreground pixels explicitly because of the rectangular bounding box provided by the user. “Iterative estimation” assigns provisional labels to some pixels (in the foreground) that can be retracted subsequently. Border matting is used to overcome the problem of blur and mixed pixels in the segmentation. Although a formal evaluation of the results is not performed, a visual inspection shows better results than other methods.

B. Segmentation using graph cuts and shape priors

Vicente [1] uses a natural assumption about the connectivity of objects to overcome the shortcomings of graph cuts in segmenting elongated objects. An explicit connectivity prior is imposed on the segmentation. The user marks certain pixels that must be connected to the object being segmented, in addition to the pixels required to be foreground or background. The algorithm imposes this connectivity to get a detailed segmentation of elongated objects or thin parts of objects.

Lempitsky et al. [3] use a technique where the user draws a bounding box around the object to be segmented. This is an intuitive first step for the user. The bounding box not only excludes its exterior from consideration but also imposes a strong topological prior. This prevents the solution from shrinking, as discussed in [12]. The algorithm is driven towards a sufficiently ‘tight’ segmentation, which means that the segmented object should have parts sufficiently close to the edges of the bounding box. This work also defines the ‘tightness’ of shapes and globally optimizes a cost function similar to that given in Equation 2. Experiments are conducted and compared to the images used in GrabCut [10]. The algorithm is slower than GrabCut but it is more accurate.

PoseCut [4, 5] uses dynamic graph cuts to optimize a cost function based on Conditional Random Fields (CRFs) to simultaneously segment and estimate the pose of humans. A simply-articulated stickman model is used to ensure human-like segmentations. The distance transform of this stickman is used as a shape prior for segmentation. Region and boundary properties are represented by GMMs of pixel intensities and pose-specific stickman models respectively.

PoseCut is based on ObjCut [11]. ObjCut is based on a probabilistic approach which can deal with object deformation. Layered pictorial structures (LPS) are used as shape priors for segmentation. Pictorial structures are a combination of 2D patterns based on their shape, appearance and spatial layout. ObjCut combines graph cut segmentation and object recognition techniques discussed in Felzenszwalb and Huttenlocher [13, 14]. The parameters of pictorial structures have to be estimated from the data and graph cuts are used to segment images. Likelihoods for parts are estimated using features and spatial locations of the parts. The desired configuration of parts of the object is given a lower cost than other unlikely configurations. Accurate object specific segmentations are achieved by combining LPS and MRFs.

A star-shape segmentation prior is used for graph cut image segmentation in [15]. The star-shaped prior is used as a generic shape for all objects. In comparison to Equation 2, the cost function used in this work is

$$E(A) = \sum_{p \epsilon \mathcal{P}} R_p(A_p) + \sum_{\{p,q\} \epsilon \mathcal{N}} (B_{\{p,q\}} + S_{\{p,q\}}) \delta(A_p, A_q) \tag{5}$$

where $S_{\{p,q\}}$ is the shape prior. The shape prior is encoded using the distance transform of a learned shape. The shape prior tries to remove the shrinking bias of a graph cut segmentation and can be compared to other 'ballooning' terms. 'Ballooning' terms are used in [17] to inflate the segmented region. The inflation of the segmented region is used to accurately reconstruct thin protrusions and concavities in the 3D reconstruction problem. The value for the 'ballooning' term is set manually. The results using shape priors are promising but there are certain shortcomings. The major assumption in this work is that the center of the shape is known. The idea of using the star-shape prior for all objects gives rise to problems of shape alignment and of imposing the wrong shape prior.

Freedman and Zhang [16] incorporate level-set templates to introduce a shape energy into the overall cost function. The user is required to draw circles around the foreground and squares in the background, similar to the bounding box in [3]. The level-set templates are estimated by parameterizing the curve of the object boundary.

C. Video Segmentation

Criminisi et al. [18] present an algorithm for the real time foreground/background segmentation in monocular video sequences. The algorithm uses Hidden Markov Models (HMMs) to model temporal changes and a spatial MRF to favour colour coherence. Spatial and temporal priors and likelihoods of colour and motion are used to get accurate results. The fusion of colour and motion for segmentation ensures the foreground being segmented even if it is similar in colour to the background.

Kolmogorov et al. [20] segment binocular stereo video using Layered Graph Cuts (LGC) and Layered Dynamic Programming (LDP). An extended 6-state space for foreground/background separation, a colour-contrast model and the stereo-match likelihood are used to define the region and boundary measurements. The main contribution of their work is the fusion of stereo with colour and contrast, which results in good quality segmentation of temporal sequences without imposing any explicit temporal consistency between neighbouring frames.

Li et al. [19] present a system for cutting a moving object out of a video clip and inserting it into another video. It starts by performing a 3D graph cut, which pre-segments the video into foreground and background regions while preserving temporal coherence. The watershed transform is used for this pre-segmentation. The initial segmentation is refined locally by using a 2D graph cut on each frame, which utilizes the colour properties of the frame. Brush tools are provided to control the user boundary precisely, wherever needed. Coherent matting is used to smooth out the object boundary in a post-processing stage. Although this approach views the video as a 3D object, it requires a lot of interaction and can be cumbersome. The preprocessing, actual graph cut optimization and post-processing stages are slow. The approach of this paper is loosely based on this work, but with many improvements.

III. Implementation

In this paper, the work done in PoseCut [5] is extended to videos and 3D spatio-temporal graph cuts for videos are investigated. The results using shape priors are compared to those from methods discussed in our previous work [2]. The videos from the Microsoft i2i dataset [9] are used to test the methods.

A. Graph cut setup

A graph is set up by defining each pixel as a node and connections between pixels as edges. For images an 8-pixel neighbourhood is used, where each pixel is connected to pixels adjacent to it in all directions. A video is viewed as a 3D object and a 26-pixel neighbourhood is used. Thus each voxel is connected to 8 adjacent voxels in the same frame (intra-frame connections) and 9 pixels in the previous and next frame (inter-frame connections). The graph is constructed by assigning weights to each pixel or voxel based on region and boundary properties and information from the shape prior. Colour spaces like RGB and Luv are used to model the regions, and boundary properties like standard edge detection techniques are used. Gaussian Mixture Models (GMMs) are used to model region properties and estimate the probability of each pixel being 'foreground' or 'background' based on these models. This is discussed in detail in our previous work [2].

The main contribution of this paper is the use of a shape prior. A shape prior term is added to the cost function as shown in Equation 5. A circular shape prior is defined using its center and radius parameters. This circular shape prior is then aligned with the object in the image. The edge weights on all pixels are scaled using the distance transform values from the shape prior. This ensures that a pixel away from the shape prior will have a higher cost and will be more likely to be classified as background.

An undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is defined with a set of nodes, $\mathcal{V}$, and a set of undirected edges, $\mathcal{E}$. Each edge $e \in \mathcal{E}$ is assigned a cost or weight w_e. There are two special nodes called the sink and source terminals. A cut is a subset of edges C $\subset \mathcal{E}$ such that the terminals become separated by $\mathcal{G}$(C) = $\{\mathcal{V}, \mathcal{E} \backslash \text{C}\}$. The cost of a cut is the sum of costs of the edges

$$\mid C \mid = \sum_{e \epsilon C} w_e. \tag{6}$$

A cut partitions the nodes in the graph corresponding to a segmentation of the underlying image. A minimum weight cut generates a node partitioning that is optimal in terms of properties that represent the edge weights. Powell's minimization method is used to find the parameters of the shape prior (center co-ordinates and radius) that minimize the cost, thus aligning the shape prior with the object to be segmented.

B. Image segmentation with shape priors

The user-marked pixels are used as cues to the desired segmentation. GMMs are used to estimate the probability of each pixel belonging to either of the two classes. RGB and Luv colour spaces are used as features in the GMMs. Boundary properties are defined using standard edge detection methods like Canny edge detector or gradient based methods. The shape prior is imposed on the image and is used to assign weights to the pixels. The distance transform from the shape prior is used to increase the probability of the pixels close to the shape being included in the segmentation. Powell's method of minimization [7] is used to align the shape prior to the image to minimize the cost of the cut.

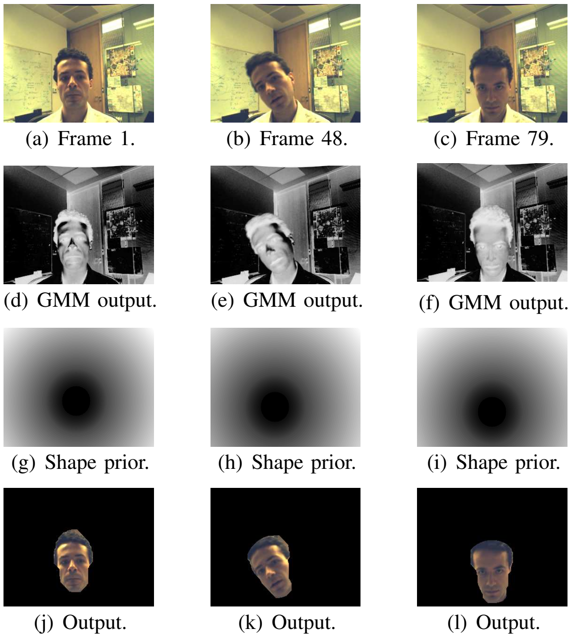
C. Video segmentation with shape priors

Video is a collection of frames and is viewed as a 3D object. A 3D graph is set up using each pixel in each frame as a node. Inter- and intra-frame connectivity between the nodes is established. The first frame is used to train the GMMs based on RGB and Luv color spaces. The shape prior is aligned to the each image using Powell's method to give the minimum cost. An addition proximity term is added to the cost function to penalize discontinuity in the segmentation. The proximity term is calculated using the distance between two shape priors in consecutive frames. The graph cut is perform on the spatio-temporal 3D object and each pixel is assigned as 'foreground' or 'background'.

Figure 1 shows the process of video segmentation using shape priors. The first row contains three frames from the video sequence. The second row shows the logarithmic likelihood ratios of the images in the top row based on a GMM trained on the face. The aligned shape priors are shown in the third row of images. The segmentation of the three frames is displayed in the last row. The frames are chosen in such a way that they contain different orientations of the face. It can be seen that the face is accurately segmented using the circular shape prior even if the face is rotated and translated. The alignment of the shape prior also changes according to the position of the face in the different frames.

IV. Results

This section compares segmentation using shape priors to segmentation using just GMMs and edge detection methods [2]. It shows the advantage of using a shape prior in segmentation. Segmentations using GMMs only, GMMs and edge detection and GMMs and edge detection with shape priors are compared for using video sequences from the Microsoft i2i dataset [9].


(a) Frame 1. (b) Frame 48. (c) Frame 79.

(d) GMM output. (e) GMM output. (f) GMM output.

(g) Shape prior. (h) Shape prior. (i) Shape prior.

(j) Output. (k) Output. (l) Output.

Fig. 1. Video segmentation using shape priors. The first row contains the original frames (a-c). The probabilities using GMMs (d-f) are shown in the second row. The distance transform from the aligned shape priors (g-i) is shown in the third row. The segmentations using shape priors (j-l) are shown in the final row.

A. Image segmentation

Figure 2 shows the different steps in segmenting images using shape priors. The two original images are shown in Figures 2(a) and 2(b). The probability of each pixel using the logarithmic likelihood ratio [2] are shown in Figures 2(c) and 2(d). The shape prior is aligned by optimizing its parameters using Powell's method. Figures 2(e) and 2(f) show the distance transform from the aligned shape prior. The outputs of the segmentation are displayed in Figures 2(g) and 2(h). The shape prior is correctly aligned in all images. The face is correctly segmented despite colour and intensity differences.

B. Video segmentation

Figures 3, 4 and 5 are organized in the same way by displaying different methods in different rows. The first row shows the original frames in the sequence. The segmentations of those frames using only colour based GMMs are shown in the second row. The third row displays segmentations using GMMs and edge detection methods. The segmentations from the shape prior, with GMMs and edge detection, are shown in the final row.

Figures 3(k) and 3(l) show that shape priors provide accurate segmentations even if the orientation of the object changes. The face has been tilted to the side, but is accurately segmented using shape priors while other methods fail. Figures 4(j), 4(k) and 4(l) show the effect of changes in the position of the object and background motion on the segmentation. This shows that the shape prior is being correctly aligned to the object using Powell's method. It is observed that using only GMMs as

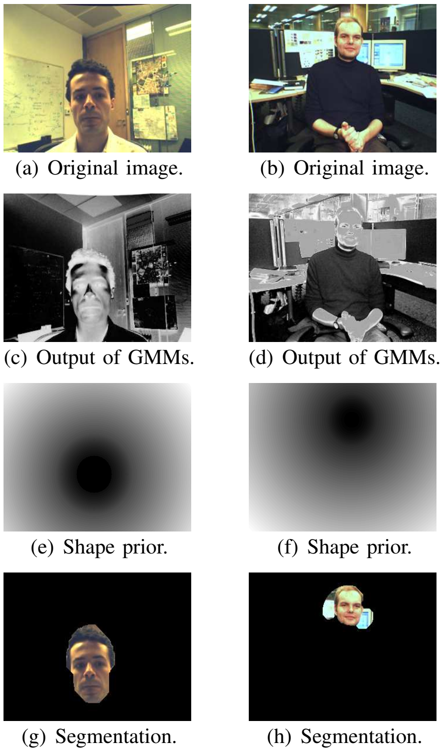

(a) Original image.
(b) Original image.
(c) Output of GMMs.
(d) Output of GMMs.
(e) Shape prior.
(f) Shape prior.
(g) Segmentation.
(h) Segmentation.

Fig. 2. Image segmentation using shape priors and graph cuts. The figure shows (a-b) the original images, (c-d) probability estimation using GMMs, (e-f) distance transform from the shape prior aligned using Powell's method and (g-h) the outputs of the segmentation respectively.

in Figures 3(d) , 3(e) and 3(f) results in many pixels being wrongly classified, because the background and foreground have similar colours. GMMs and edge detection methods are not accurate because of the numerous boundaries in the image and the similarity between foreground and background.

Graph cuts and shape priors provide more accurate segmentations than other methods, even though the background is similar to the object in colour. The segmentation in Figures 5(c) and 5(d) classifies the hands of the person as foreground because they are the same colour as the face. Many pixels from the background are also wrongly classified as foreground. The segmentation using shape priors in Figures 5(g) and 5(h) provide accurate segmentations in these cases.

In general, it can be seen that shape priors result in more accurate segmentations compared to other methods. They overcome certain drawbacks of other methods like background motion, changes in the position and orientation of the object, and the object and background being similar in terms of colour. The motion information from videos is used for accurate segmentation and the preprocessing is reduced.

V. Conclusions and Future Work

It can be concluded that using shape priors with graph cuts can result in very accurate segmentations. The comparison of segmentations using shape priors to those without shape priors clearly shows the usefulness of the shape prior. The segmentations are more accurate than other methods even with the object to be segmented is similar to the background. The motion of the object or the background in a video does not adversely affect the performance of the segmentation. The average time taken for a segmentation is 0.2 seconds for images and 2 seconds per frame for videos. Thus it can be

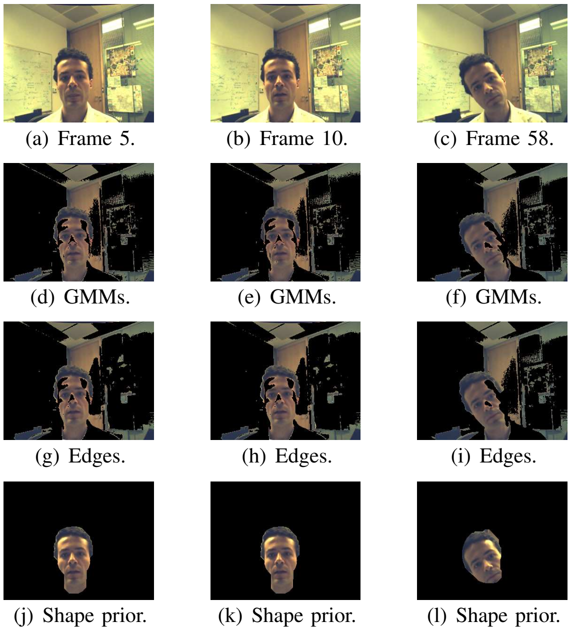

(a) Frame 5.
(b) Frame 10.
(c) Frame 58.
(d) GMMs.
(e) GMMs.
(f) GMMs.
(g) Edges.
(h) Edges.
(i) Edges.
(j) Shape prior.
(k) Shape prior.
(l) Shape prior.

Fig. 3. Comparison of segmentation methods. Some frames (a-c) from the original sequence are shown in the first row. Segmentations using graph cuts and colour GMMs (d-f), GMMs with edge detection methods (g-i) and GMMs with shape priors (j-l) are shown.

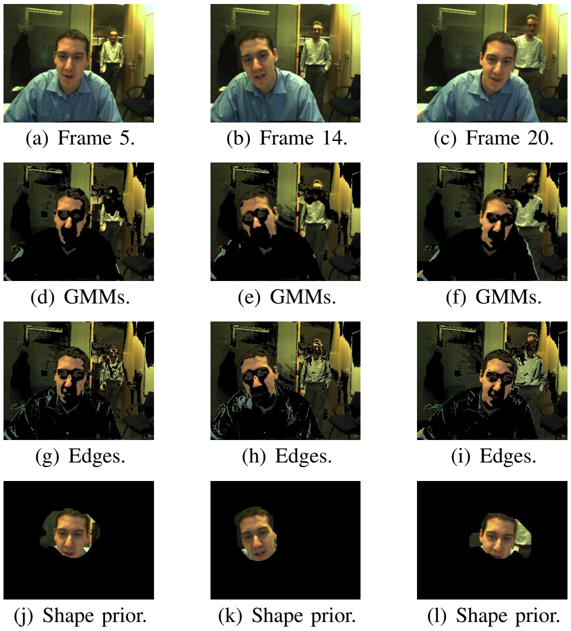
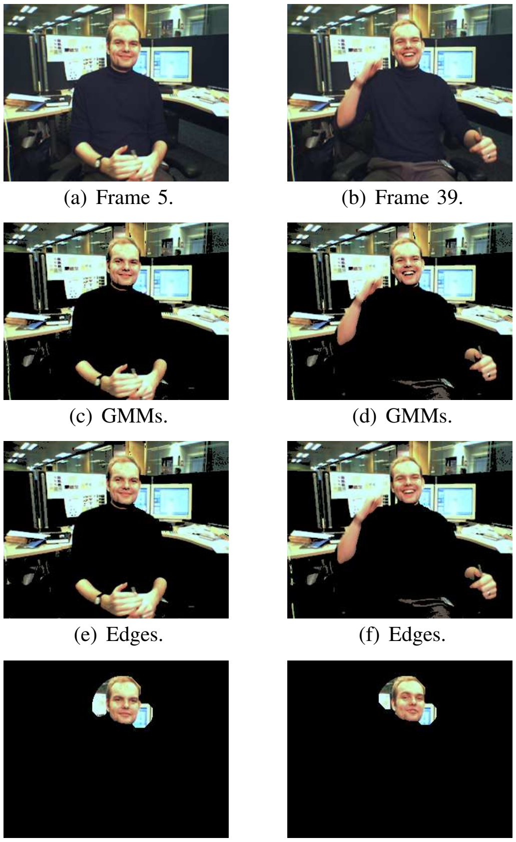

(a) Frame 5.
(b) Frame 14.
(c) Frame 20.
(d) GMMs.
(e) GMMs.
(f) GMMs.
(g) Edges.
(h) Edges.
(i) Edges.
(j) Shape prior.
(k) Shape prior.
(l) Shape prior.

Fig. 4. Comparison of segmentation methods. Some frames (a-c) from the original sequence are shown in the first row. Segmentations using graph cuts and colour GMMs (d-f), GMMs with edge detection methods (g-i) and GMMs with shape priors (j-l) are shown.



(a) Frame 5.

(b) Frame 39.

(c) GMMs.

(d) GMMs.

(e) Edges.

(f) Edges.

(g) Shape prior.

(h) Shape prior.

Fig. 5. Comparison of segmentation methods. Some frames (a-b) from the original sequence are shown in the first row. Segmentations using graph cuts and colour GMMs (c-d), GMMs with edge detection methods (e-f) and GMMs with shape priors (g-h) are shown.

concluded that using shape priors with graph cuts can improve segmentation of images and videos.

Aligning the shape prior to the desired object is done using Powell's method. The shape prior tested in this paper is circular. This work can be extended further to include complex shape priors like ellipses or a collection of shapes. Other gradient descent methods of minimization can be used for accurate alignment. A detailed performance evaluation can be conducted by varying the parameters of the segmentation.

REFERENCES

[1] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. *Graph cut based image segmentation with connectivity priors*. Technical report, 2008.

[2] M. Kulkarni and F. Nicolls. *Interactive Image Segmentation using Graph Cuts*. PRASA 2009: Proceedings of the 20th Annual Symposium of the Pattern Recognition Association of South Africa, pages 99-104, 2009.

[3] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp. *Image segmentation with a bounding box prior*. pages 277-284, 2009.

[4] Matthieu Bray, Pushmeet Kohli, and Philip H. S. Torr. *Posecut: Simultaneous segmentation and 3D pose estimation of humans using dynamic graph-cuts*. In ECCV, pages 642-655, 2006.

[5] Pushmeet Kohli, Jonathan Rihan, Matthieu Bray, and Philip H. S. Torr. *Simultaneous segmentation and pose estimation of humans using dynamic graph cuts*. International Journal of Computer Vision, 79(3):285-298, 2008.

[6] Y. Boykov and M. P. Jolly. *Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images*. volume 1, pages 105-112, July 2001.

[7] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. (1988). *Numerical recipes in C*. Cambridge: Cambridge University Press.

[8] Yuri Boykov and Vladimir Kolmogorov. *An experimental comparison of mincut/ max-flow algorithms for energy minimization in vision*. IEEE Trans. Pattern Anal. Mach. Intell., 26(9):1124-1137, 2004.

[9] Microsoft Research. Microsoft i2i dataset, April 2010. URL http://www.research. microsoft.com/vision/cambridge/i2i.

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

[11] M. Pawan Kumar, Philip H. S. Torr, and A. Zisserman. *Obj cut*. In CVPR '05 - Volume 1, pages 18-25, 2005.

[12] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. *Interactive image segmentation using an adaptive GMMRF model*. In ECCV, pages 428-441, 2004.

[13] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. *Efficient matching of pictorial structures*. In CVPR, 2000.

[14] Pedro F. Felzenszwalb, Daniel P. Huttenlocher, and Jon M. Kleinberg. *Fast algorithms for large-state-space HMMs with applications to web usage analysis*. In NIPS, 2003.

[15] Olga Veksler. *Star shape prior for graph-cut image segmentation*. In ECCV (3), pages 454-467, 2008.

[16] Daniel Freedman and Tao Zhang. *Interactive graph cut based segmentation with shape priors*. In CVPR '05 - Volume 1, pages 755-762, 2005.

[17] George Vogiatzis, Philip H. S. Torr, and Roberto Cipolla. *Multi-view stereo via volumetric graph-cuts*. In CVPR (2), pages 391-398, 2005.

[18] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. *Bilayer segmentation of live video*. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 53-60, 2006.

[19] Yin Li, Jian Sun, and Heung yeung Shum. *Video object cut and paste*. ACM Transactions on Graphics, 24:595-600, 2005.

[20] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother. *Bi-layer segmentation of binocular stereo video*. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2, pages 407-414, 2005.