Shape Constrained Figure-Ground Segmentation and Tracking

Zhaozheng Yin Robert T. Collins
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA16802
{zyin,rcollins}@cse.psu.edu

Abstract

Global shape information is an effective top-down complement to bottom-up figure-ground segmentation as well as a useful constraint to avoid drift during adaptive tracking. We propose a novel method to embed global shape information into local graph links in a Conditional Random Field (CRF) framework. Given object shapes from several key frames, we automatically collect a shape dataset on-the-fly and perform statistical analysis to build a collection of deformable shape templates representing global object shape. In new frames, simulated annealing and local voting align the deformable template with the image to yield a global shape probability map. The global shape probability is combined with a region-based probability of object boundary map and the pixel-level intensity gradient to determine each link cost in the graph. The CRF energy is minimized by min-cut, followed by Random Walk on the uncertain boundary region to get a soft segmentation result. Experiments on both medical and natural images with deformable object shapes are demonstrated.

1. Introduction

Long-term object tracking must be resilient to large changes in appearance of both the object and its surrounding environment. For example, online discriminative analysis between foreground and background is effective to track objects under varying lighting and scene conditions [1, 6]. However, due to a simple object shape representation (*e.g.* a rectangular box) or background distractors, some misclassified pixels are mistakenly included and used to select new tracking features for the next frame or update the object appearance model. If there is no object-level constraint used during the adaptation, the accumulated pixel classification error will lead the tracker to drift off the object. A solution to this problem requires raising the level of abstraction at which the tracker represents its target. The goal must be tracking "objects", not a box of pixels or a color distribution. If we can explicitly segment the foreground from background, it is possible to keep the adaptive model anchored on just the foreground pixels. In other words, explicit figure-ground segmentation allows us to use object shape constraints to avoid drift during adaptive tracking (Figure 1). Meanwhile, a precise object shape representation is also helpful to search for and recognize the same object again after occlusion or tracking failure.

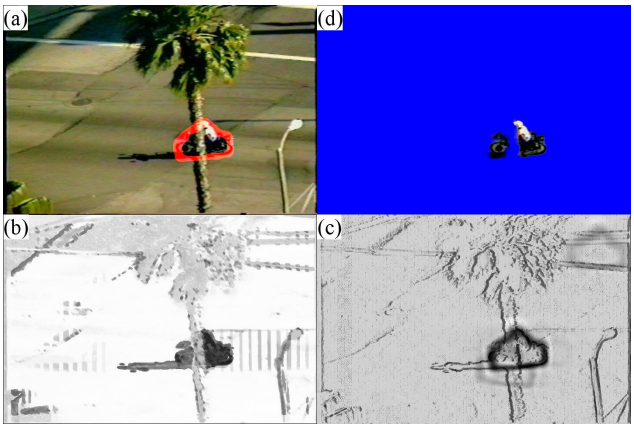

Figure 1. Shape constrained figure-ground segmentation can help reduce pixel classification error when adapting an object model. (a) Probability map of a learned global shape (red) is overlaid on the input image; (b) and (c) represent data and link energy terms in the CRF graph; (d) Resulting soft figure-ground segmentation by graph-cut and Random Walk is shown on a blue screen.

Object segmentation from images/videos has long been an active research field. For example, image segmentation has been treated as a graph partition problem that simultaneously minimizes the cross segment connectivity and maximizes the within segment similarity of pixels [15]. For video sequences, layered motion models have been one of the key paradigms for segmenting objects, assuming restricted parametric motion models or an elliptical shape [17, 19]. In [7, 14], probabilistic fusion of multiple features is used to segment objects in a Conditional Random Field (CRF) framework [11].

Recently, global shape information has been introduced into segmentation systems. For example, rectilinear shape

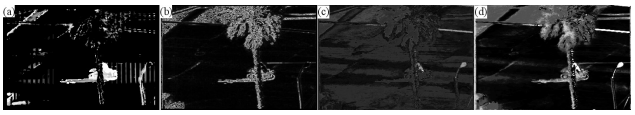

Figure 2. A suite of likelihood features for figure-ground segmentation: (a) motion; (b) local color contrast; (c) spatial color variance; (d) figure-ground color likelihood ratio.

is embedded into the Laplacian matrix by using a "rectilinearity measure"[16] and the diagonal of the Laplacian matrix is modified to provide a shape constrained normalized cut [18]. Using a known shape prior, Freeman and Zhang [8] manually select the object location (in the form of seeds) and align a distance-transformed template there to provide link terms in the graph-cut segmentation system [3]. Given a training set of ground truth segmentations, Levin and Weiss [12] trained a fragment-based segmentation algorithm. An object detector scans all possible subimages until it find a subimage that contains the object. Within the subimage, object parts are searched for by normalized correlation. The fragment locations are used to bias the energy minimizing problem towards finding the right solution.

Traditionally, link terms in the energy function are only determined by pixel-level intensity gradients. Thus, a node in a graph only communicates with its neighboring nodes while being blind to the global object shape. In this paper, we propose a novel method to embed global shape information into local graph links in a CRF framework without human interaction or exhaustive fragment detection. To be able to exploit global shape information, we propose a reliable way to detect figure-ground boundaries by combining different appearance and motion features to yield a region-based probability of object boundary map. We also use simulated annealing and local voting to align deformable shape templates with an image to yield a global shape probability map that indicates each link's shape boundary likelihood. Finally, the global shape probability and region-based probability of object boundary are fused with the pixel-level intensity gradient to determine each link cost in the graph. The total CRF energy is minimized by min-cut. Based on the binary graph-cut segmentation results, we generate a trimap and run Random Walk [9] on the uncertain boundary region to get a soft segmentation result where each pixel is assigned a foreground probability (foreground matte).

The process of shape-constrained segmentation raises the question of how to get object shapes and deform them. Given object shapes from several key frames, we automatically collect a shape dataset on-the-fly. All the shapes are aligned to a reference shape, and the occluded ones are automatically detected and discarded. Statistical analysis is then applied to the refined shape dataset to build a collection of deformable shape templates representing global object shape.

The contributions of this paper include two aspects: integrating global shape information and region-based probability of object boundary into CRF graph link terms (Section 2), and automatic shape model learning (Section 3). Experiments on both medical and natural images with deformable object shapes are demonstrated in Section 4.

2. Figure-Ground Segmentation in a CRF

Conditional Random Fields (CRF) have been widely used to solve segmentation and discriminative labeling problems [11]. Let $\{I_i\}$ and $\{s_i\}$ denote the sets of image pixels and corresponding labels respectively. Label $s_i = 1$ if I_i belongs to the foreground, and $s_i = -1$ otherwise. Let $\{A, B\}$ be a graph where set A denotes pixel node labels s and B contains all the pairwise links between two neighboring nodes. Globally conditioned on the observation I, the joint distribution over labels s is

$$P(s|I) = \frac{1}{Z}\exp\left\{\sum_{i\in A}\sum_{k}\lambda_k\Phi_k(s_i, I) + \sum_{(i,j)\in B}\Psi(s_i, s_j, I)\right\} \propto \frac{1}{Z}\exp\{-E\} \quad (1)$$

where Z is a normalizing factor and E is the energy function to be minimized. The Φ_k's are data association potentials (data terms) generated by different segmentation cues, linearly combined with weights λ_k. Ψ represents pairwise interaction potentials (link terms) between neighboring nodes.

2.1. Segmentation Cue Integration

Segmenting figure from ground using a single feature alone can be expected to be error-prone, especially when nearby confusers are present. However, the CRF model is suitable for probabilistically fusing arbitrary, overlapping and agglomerative observations from both the past and future. We use a suite of useful features for figure-ground segmentation in image sequences including motion, local color contrast, spatial color variance and figure-ground color likelihood ratio (Figure 2, [21]).

Motion is a powerful feature for moving object segmentation. To detect object motion from a moving camera, we adopt the forward/backward motion history image (MHI) method, which combines motion information over a sliding temporal window [20]. Figure 2(a) shows a sample motion detection result denoted as $f_m(\cdot)$. The detection result is noisy due to errors in camera motion compensation, parallax, and fluctuations of background appearance. If the camera is static and background is known, background subtraction can be used to compute the motion feature.

Local color contrast at pixel u is computed as

$$f_c(I, u) = \sum_{v\in N(u)} \frac{\| I_u - I_v \|^2}{\| I_u + I_v \|^2 + \epsilon} \quad (2)$$

where $N(u)$ is a 5×5 local window around pixel u. The small ϵ is added to avoid dividing by zero.

It is observed that colors distributed over a larger spatial area of the image are less likely to belong to a salient foreground object. Instead of modeling all colors in an image by Gaussian mixture models as done in [13], we directly compute a color's x-coordinate variance as

$$\text{var}_x(c) = \frac{\sum_v \delta(I_v = c)(v_x - m_x(c))^2)}{\sum_v \delta(I_v = c) + \epsilon} \quad (3)$$

where $\delta(\cdot)$ is an indicator function, v_x is the x-coordinate of pixel v, and $m_x(c)$ is the x-coordinate mean of those pixels with color c. Similarly, we compute a color's y-coordinate variance for $\text{var}_y(c)$. Although the traditional spatial variance of a color component is a 2-by-2 covariance matrix, here we approximate it by a scalar $\text{var}(c) = \text{var}_x(c) + \text{var}_y(c)$. After normalizing $\text{var}(\cdot)$ to the range $[0, 1]$, the color spatial-distribution at pixel u is defined as

$$f_v(I, u) = 1 - var(I_u) \quad (4)$$

Given the previous frame's foreground probability (soft segmentation), P_F, we can extract an accurate foreground color appearance model, *e.g.* color histogram H_F. Similarly, the background appearance model, H_B, is computed from background probability P_B. The figure-ground color likelihood ratio at pixel u is then computed as

$$f_r(I, u) = \frac{H_F(I_u)}{H_B(I_u) + \epsilon} \quad (5)$$

To maintain temporal coherence, the segmentation result from the previous frame is used as a temporal-prior data term for the current frame.

$$f_t(I, u) = e^{\frac{-\| I_u - I_{u'} \|^2}{2\beta}} \cdot P_F(u') \quad (6)$$

where $\beta = \overline{\| I_u - I_{u'} \|^2}$ is the average of all $\| I_u - I_{u'} \|^2$, and u' is the pixel in the previous frame corresponding to pixel u in the current frame. Since the object and background can be moving between the two frames, we compute dense optical flow [2] to determine the pixel alignment between frames. $P_F(u')$ represents the previous frame's soft segmentation result for pixel u'.

2.2. Region-based Probability of Object Boundary

Since figure-ground segmentation aims to find an accurate boundary between foreground and background, edge pixels deserve more attention. Before exploring global object shape information, we need reliable ways to detect fragments of object boundaries.

The spatial gradient at pixel u is computed using Gaussian derivatives and denoted as $\vec{g}(u) = [g_x(u), g_y(u)]$ with direction $\vec{n}(u) = [\cos(\theta_u), \sin(\theta_u)]$ and magnitude $\| g(u) \|$ where $\theta_u = \tan^{-1}(g_y(u)/g_x(u))$. After normalizing all the pixel gradient magnitudes to the range $[0, 1]$, we consider pixel u as an edge pixel if $\| g(u) \| > T$ (*e.g.* $T = 0.1$), and extract feature vector $V_1(u)$ as

$$V_1(u) = \{\lambda_m f_m(I, p), \lambda_c f_c(I, p), \lambda_v f_v(I, p), \lambda_r f_r(I, p), \forall p,\ s.t.\ p = u + i\vec{n}(u), 0 < i \leq L\}$$

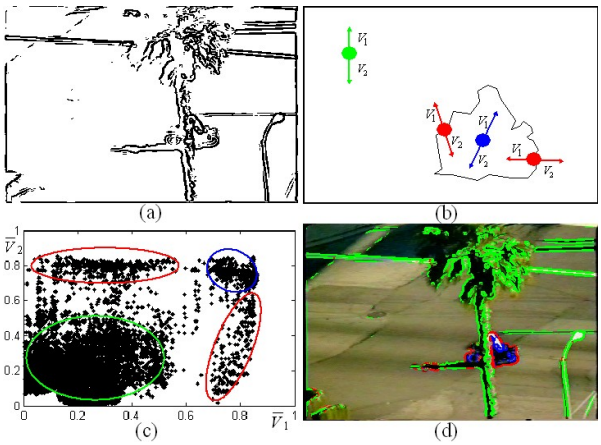

Figure 3. Edge pixel classification. (a) Edge pixels (black); (b) Feature extraction along the normal direction of edge pixels; (c) Cluster analysis on the feature vector's mean; (d) Edge pixel classification: figure-ground boundary(red), foreground edges (blue), background edges (green).

This essentially samples L (*e.g.* $L = 10$) pixels along $\vec{n}(u)$ and generates a feature vector $V_1(u)$ with length $4L$. Similarly, we get $V_2(u)$ along $-\vec{n}(u)$. The motivation for extracting these two vectors is illustrated in Figure 3(c). The feature vector means of each pixel, $(\bar{V}_1(u), \bar{V}_2(u))$, form four clusters, estimated here by the EM algorithm. When both $\bar{V}_1(u)$ and $\bar{V}_2(u)$ are small, pixel u is a background edge. When both of them are large, pixel u is a foreground edge. Otherwise, pixel u is classified as an object boundary pixel. In addition to classifying edge pixels into these three classes, we compute each pixel's boundary probability as

$$P_b(u) = \| V_1(u) - V_2(u) \| \quad (7)$$

2.3. Global Shape Detection

Based on the probability of object boundary map (Figure 4(a)), we align the shape template (Figure 4(b)) to the image by simulated annealing [10]. The minimization function used in the annealing is defined as

$$J(\theta) = \sum_{u \in T(W;\theta), v \in W} P_b(u) \cdot Q(v) \quad (8)$$

where P_b is the probability of object boundary map and $Q(\cdot)$ represents the shape template learned by a method that will be described in Section 4. T is a similarity transformation $\theta = (x, y, \alpha, \sigma)$ corresponding to translation (x, y), rotation α and scaling σ that aligns the template window W with the input image. Originally, annealing tries to find the global minimum in a parameter space while avoiding being trapped at local minima. This requires a high initial temperature and slow cooling process. Here, we allow the annealing temperature to decrease fast but consider all samples during the annealing process, not just the final minimum to which it converges. Figure 4(c) shows a sampled 3D parameter space (x, y, α) with plotted points showing

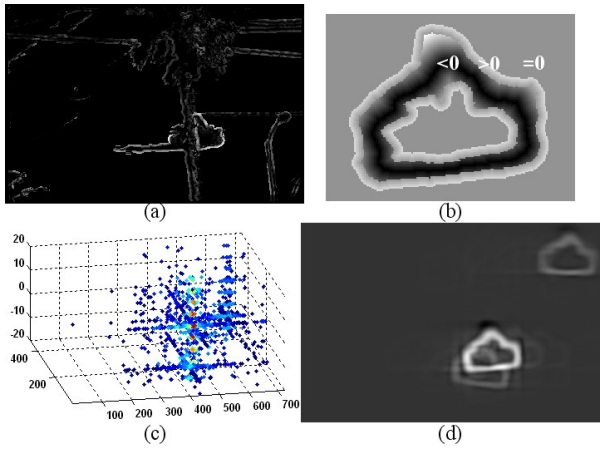

Figure 4. Global shape probability by annealing and voting. (a) Region-based probability of object boundary, P_b; (b) Deformable object shape template; (c) Sampled parameter space by annealing; (d) Global shape probability, P_s, formed by local voting.

the states that simulated annealing visited. Since we anneal fast, we sacrifice the global convergence of annealing, and false local minima may be detected. However, most intermediate sampled points in the parameter space are around the global minimum, and the entire parameter space is "ergodically" sampled because of the Monte Carlo importance sampling property inherited by annealing techniques.

We consider the sampled parameter space as an aggregated voting space, and project these votes back to the image space to get each pixel's global shape probability (Figure 4(d))

$$P_s(u) = \sum_{\theta} J(\theta) \cdot Q(T^{-1}(u;\theta)) \tag{9}$$

2.4. Graph-cut for Random-Walk

In Eq.1, data terms, $\Phi_k(s_i, I)$, measure how likely it is that node i has label s_i given image I without considering other nodes in the graph. The kth data term at node i with label s_i is defined as

$$E_k(s_i, I) = -\Phi_k(s_i, I) = -\log p(s_i | f_k(I)) \tag{10}$$

where the f_k's are related to different segmentation cues. The link term between nodes s_i and s_j given image I is defined as

$$\begin{aligned} E(s_i, s_j, I) \quad = \quad & \delta(s_i \neq s_j) \frac{\gamma + e^{\frac{-\|I_i - I_j\|^2}{2\beta}}}{1+\gamma} \frac{1}{d(s_i, s_j)} \\ & \cdot \left(1 - \frac{P_b(s_i) + P_b(s_j)}{2}\right) \\ & \cdot \left(1 - \frac{P_s(s_i) + P_s(s_j)}{2}\right) \end{aligned} \tag{11}$$

where $d(\cdot)$ is an Euclidean distance measure, γ is a dilution constant (*e.g.* $\gamma = 1$), $f_k(\cdot)$, $P_b(\cdot)$ and $P_s(\cdot)$ are normalized

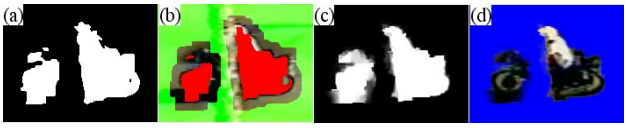

Figure 5. Soft segmentation by graph-cut and Random-Walk. (a) Binary mask from graph-cut; (b) Trimap: known foreground (red), known background (green), unmarked region is unknown; (c) Foreground matte, P_F; (d) Foreground matting on a blue screen.

to the range $[0, 1]$. The first term in Eq.11 is widely used in graph-cut segmentation systems [3, 7] to describe how local neighboring nodes interact. For example, if the pixel color difference is small, this term encourages nodes s_i and s_j to have the same label. In addition to the pixel-level interaction, we enhance the graph link term by the region-based probability of object boundary (second term) and global shape probability (third term), thus the total energy function attracts the min-cut to occur around the figure-ground boundary.

Based on the above data and link terms, the min-cut algorithm generates a binary segmentation mask (Figure 5(a)). In natural images, the transition between foreground and background usually happens gradually, thus the binary hard decision is usually not an accurate representation of the boundary. We propose instead to use Random Walk matting [9] to assign foreground opacity to those uncertain pixels. Based on a trimap of marked pixels (the eroded binary mask represents known foreground pixels and the complement of the dilated binary mask represents background seeds), the Random Walk algorithm generates the soft segmentation foreground probability, P_F, as shown in Figure 5(c).

2.5. Online Parameter Tuning

The weighting factors, λ_k's, for different potential functions are key parameters to be tuned in the CRF. For long video sequences, it is tedious to get a complete ground truth data set for training. Furthermore, it is desirable to dynamically update the parameters over time to adapt to scene and lighting changes. We apply discriminant analysis on each segmentation cue to measure its ability to discriminate between figure and ground. This is motivated by the fact that the segmentation features with high foreground/background discrimination deserve high weight so they can make a significant contribution to the segmentation in the next frame. Inspired by Fisher linear discriminant analysis, we define weight λ_k based on the following measure of figure-ground separability

$$\lambda_k = \max\left(0, \frac{m(f_k, P_F) - m(f_k, P_B)}{\text{var}(f_k, P_F) + \text{var}(f_k, P_B)}\right) \tag{12}$$

where $m(f, p)$ and $\text{var}(f, p)$ denote the mean and variance of function f with respect to distribution p respectively, $\lambda_k > 0$ and we normalize such that $\sum_k \lambda_k = 1$. The weights computed from the current frame are then applied to the next frame.

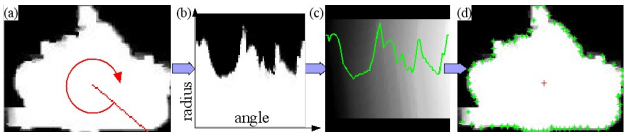

Figure 6. Finding the boundary contour of a soft-segmentation shape. (a) 2D shape template, P_F; (b) Transformed shape template in a polar coordinate system; (c) Accumulated score table generated by Dynamic Programming with the best path overlaid; (d) When converted back to image coordinates, the 1D radial curve defines a boundary contour on the shape template image.

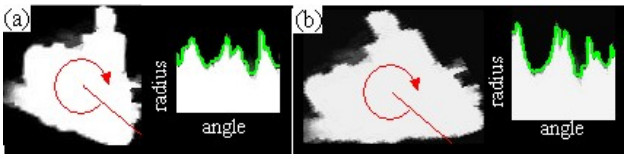

Figure 7. Shape template alignment. (a) A shape template to be aligned and its 1D radial curve; (b) Aligned shape template and its 1D radial curve with regard to the reference in Figure 6.

3. Shape Model Learning

Given initial object shapes from several key frames, we automatically collect a shape dataset on-the-fly using the above segmentation algorithm. From the collected soft segmentation masks, P_F's, we learn a set of deformable general shape templates and apply them to segment similar objects in other videos.

3.1. 1D Radial Contour Shape Representation

To effectively align and analyze shape templates, we transform 2D object shapes into 1D radial curves in a polar coordinate system [4]. As shown in Figure 6, we rotate a radial line around the shape template's centroid from 0 to 360 degrees, at 5 degree intervals, and sample pixels along these radial lines. This transforms 2D shape templates into an (angle, radius) domain. The next step is to search for the best continuous boundary radius along each radial line, i.e. an optimal path from left to right in Figure 6(b) along which the sum of accumulated gradient information is maximized. We use Dynamic Programming to find the path, resulting in a 1D radial contour shape representation, $\rho = \{\rho_\alpha : 0 \leq \alpha \leq 360^\circ\}$ (Figure 6(c)).

3.2. Shape Alignment and Occlusion Detection

Before we perform statistical analysis on the shape dataset, we need to align each shape sample to the same reference. Given a reference shape (*e.g.* Figure 6(a)) and a template to be aligned (Figure 7(a)), we extract their 1D radial curves and treat the alignment as a 1D signal processing problem. Normalized cross-correlation is applied to find the signal's translational angular shift in polar coordinates (rotation in image coordinates). The two signals' DC components are related to the radius change (scaling in image coordinates). The aligned shape template and its radial curve is shown in Figure 7(b).

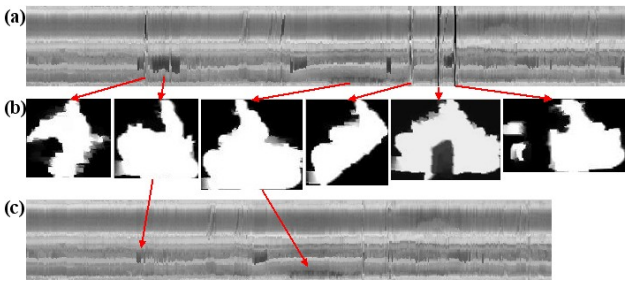

Figure 8. Shape dataset and pruning. (a) The shape dataset after alignment is summarized in a pattern image. Each column represents a template's 1D radial curve. Each row represents the radius at a specific angle from all shape templates; (b) Irregular regions in the pattern image are caused by occluded or deformed shape templates; (c) Occluded shapes are detected and removed while deformed shapes are kept for variation analysis.

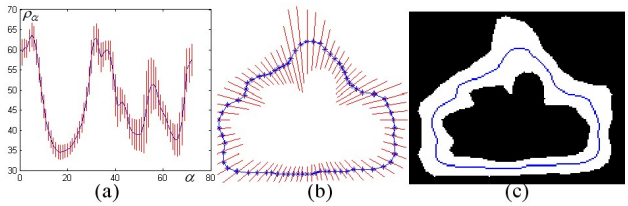

Figure 9. Contour variance. (a) Mean (blue) and variance (red) of the 1D radial contour shape representation; (b) The 1D contour shape is converted back to image coordinates; (c) Mean shape curve (blue) and ROI corresponding to different contour point's variances (white).

Using the 1D radial contour shape representation, we automatically acquire an aligned shape dataset, while avoiding the manual landmark initialization and matching required by common shape alignment procedures. Figure 8(a) shows the aligned shape dataset where each column represents a template's 1D radial curve and each row represents the radius at a specific angle from all shape templates. To detect occluded templates, we compute the mean $\bar{\rho}_\alpha$ and variance $var(\rho_\alpha)$ for each specific angle α over all the aligned shape templates. If $|\rho_\alpha(i) - \bar{\rho}_\alpha| > 3 * var(\rho_\alpha)$, this curve point is considered as a bad one of shape $\rho(i)$. If shape $\rho(i)$ contains k (*e.g.* $k = 5$) bad curve points, we treat this shape as an occluded one and remove it from the shape dataset.

3.3. Deformable Shape Template Generation

Based on the refined shape dataset (Figure 8(c)), we recompute $\bar{\rho}_\alpha$ and $var(\rho_\alpha)$ for every α and transform them back to image coordinates (Figure 9). This provides us a mean shape curve and a region-of-interest (ROI) defined by each boundary contour point's variance. Our shape template for global shape detection is illustrated in Figure 4(b). It looks similar to a contour convolved with a Difference-of-Gaussian filter, but different bandwidths, $var(\rho_\alpha)$, are applied radially to each point around the mean shape curve.

Figure 10 shows a toy example to illustrate why our template is suitable for global shape detection. This figure compares different transforms for computing graph link terms. The unsigned distance transform of a shape template is used in [8] for defining a graph link term, but it is not suitable for global shape detection. As shown in Figure 10(b), this template prefers image regions with zero values to minimize $J(\cdot)$ and the global shape probability map shows high confidence on these false regions. The signed distance transform of a shape template is used in [22] for a graph link term, but this template is biased to prefer clutter regions with a lot of edges and the shape boundary is not well detected (Figure 10(c)). The third template is obtained by three steps: computing the unsigned distance transform of a shape template, subtracting the transformed template by a positve constant h, and setting the nonnegative values to zero. The extreme case is $h = 1$ where the global shape detection becomes a template matching problem. As shown in Figure 10(d), this shape template performs better than the signed/unsiged distance transformed template, but clutter regions are still not well detected. Furthermore, this template suffers from the scale drifting problem reported in [5]. The result of our DoG-like shape template is shown in Figure 10(e), where three shape boundaries are well detected.

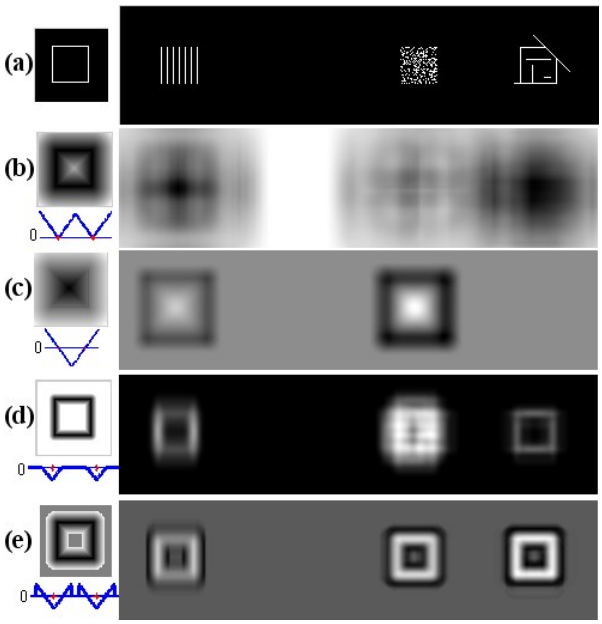

Figure 10. Comparing different shape templates for detection. (a) left: original shape template, right: testing image; (b) left: unsigned distance transform of the shape template, right: corresponding global shape probability P_s; (c) signed distance transform and P_s; (d) band restricted distance transform and P_s; (e) our shape template and P_s.

4. Experimental Results

We tested our approach on both natural color images and medical MRI images. The top half of Figure 11 shows aerial video of a motorcycle that changes its shape and appearance throughout the video and often undergoes partial or full occlusion. The bottom half of Figure 11 demonstrates how the air bladder of a zebrafish deforms its shape across a sequence of MRI slices, and has appearance similar to a nearby confusor. As shown in Figure 11, without human interaction the global object shape is well aligned to new images by annealing and local voting. The global shape probability and region-based probability of object boundary enhance the pixel-level graph link terms and attract the graph-cut to occur around the figure-ground boundary. Thus, by applying a global shape constraint, the object is well segmented from its surrounding background clutter and its appearance model is adapted over time without drift. When the object is fully occluded, the global shape probability is low everywhere in the image and the segmentation sytem will not update the object model. Using a learned deformable shape template our method works well much of the time. However, when there is no shape constraint added to the graph-cut system, the segmentation results are quite noisy, as shown in column 5 of Figure 11.

One challenge for shape-constrained segmentation comes from handling articulated object shapes such as a human walking. As shown in Figure 12, the collected shape dataset does not have a uniform pattern as in Figure 8(c), due to periodicity of the human gait. For each specific posture within a walking cycle, we first cluster and extract all similar shapes from the dataset and run statistical shape analysis on them separately. This generates a collection of deformable shapes and adds the shape configuration index as an extra parameter in the global shape detection step. Figure 13 shows the shape-constrained segmentation results on different human walking videos. One video editing application of this segmentation is matting foreground object appearance onto a new background. In Figure 14, we demonstrate switching the foreground objects between two videos and aligning their walking paces to be the same.

Some hard cases remain to be solved. When the object is heavily occluded (*e.g.* the motorcycle passes behind a moving car in Figure 8), the Random Walk algorithm can not compute the foreground probability correctly from the cluttered uncertain region. We think motion layer extraction from high-resolution videos could help solve this problem.

5. Conclusion

We formulate shape constrained figure-ground segmentation in a CRF graph model and propose a new method to embed global shape probability and region-based probability of object boundary into graph link terms. After the high-level object information is integrated with different low-level segmentation cues, the total energy function attracts the graph cut to occur around the figure-ground boundary. The binary graph-cut mask is refined by Random-Walk to

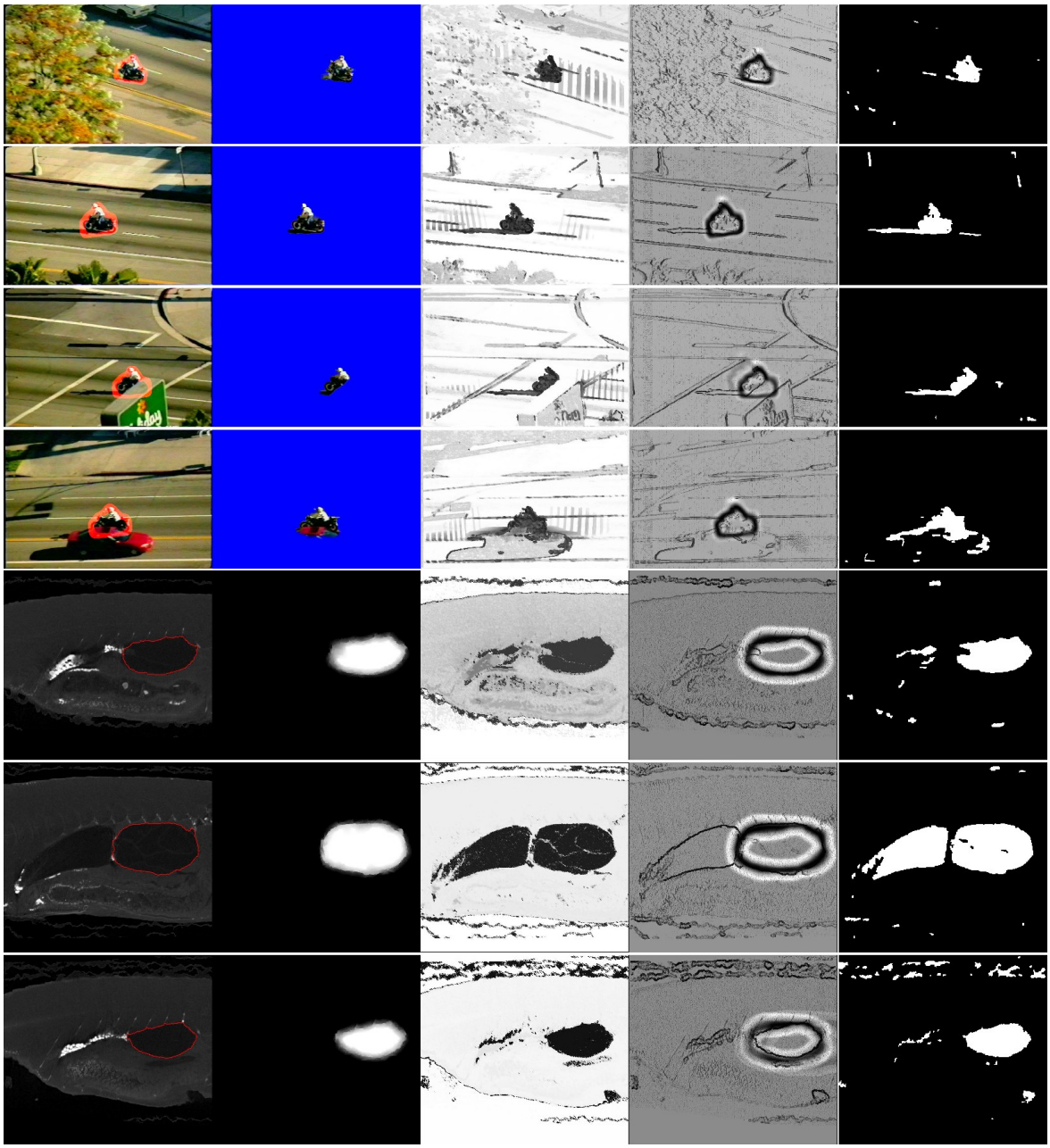

Figure 11. Shape constrained soft segmentation results compared to unconstrained graph-cut binary segmentation. Column 1: input image with overlaid shape; Column 2: soft segmentation mask; Column 3: data term in the graph; Column 4: link term enhanced with region-based probability of object boundary and global shape probability; Column 5: binary segmentation by graph-cut without shape constraint. The upper four rows are from an airborne video and the bottom three rows are from an MRI sequence of zebrafish slices.

get a soft segmentation.

We automatically collect a shape dataset on-the-fly and develop an effective algorithm to transform these 2D shape templates to 1D radial contour shapes for alignment. Statistical analysis on the aligned shape dataset removes occluded shapes and generates a mean radial contour shape with corresponding variances. We convert this 1D shape representation back into image coordinates to obtain a deformable template for global shape detection.

The shape constrained figure-ground segmentation method can be used to avoid drift during adaptive tracking. The collected shape templates could also be helpful to search for and recognize the same object again after occlusion or tracking failure.

Acknowledgement

This work was funded under the NSF Computer Vision program via grant IIS-0535324 on Persistent Tracking.

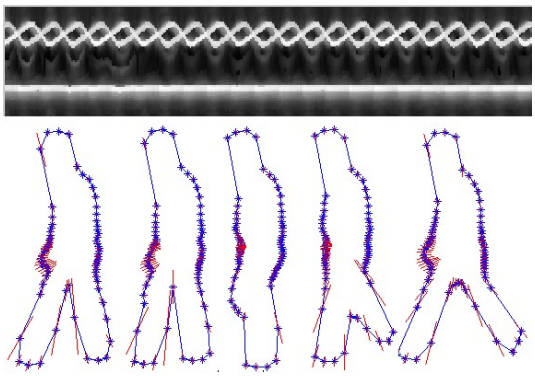

Figure 12. The top row is an aligned radial contour dataset automatically collected from the CMU MoBo dataset. The bottom row shows several samples from a collection of deformable human shapes learned from this dataset.

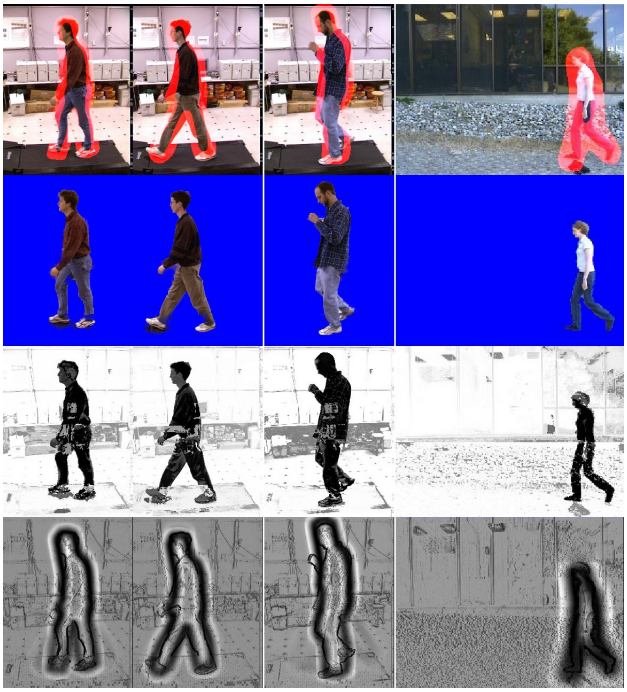

Figure 13. Applying the learned gait shapes to different video sequences. Row 1: input image with overlaid shape; Row 2: soft segmentation matte on a blue screen; Row 3 and 4: data and link terms in the graph. Left three columns are from the CMU MoBo dataset and the fourth column is from http://www.cs.brown.edu/~black/images.html.

References

[1] S. Avidan, "Ensemble Tracking," IEEE Trans. PAMI, 29(2): 261-271, 2007.

[2] M. Black and P. Anandan, "A Framework for the Robust Estimation of Optical Flow," In ICCV, p231-236, 1993.

[3] Y. Boykov and G. Funka-lea, "Graph Cuts and Efficient N-D Image Segmentation," IJCV, 109-131, 2006.

[4] Y. Chen, T. Huang and Y. Rui, "Optimal Radial Contour Tracking by Dynamic Programming," In ICIP, 2001.

Figure 14. Foreground replacement for video editing.

[5] R. Collins, "Mean-shift Blob Tracking through Scale Space," In CVPR, p234-240, 2003.

[6] R. Collins, Y. Liu and M. Leordeanu, "On-line Selection of Discriminative Tracking Features," IEEE Trans. PAMI, 27(10): 1631-1643, 2005.

[7] A. Criminisi, G. Cross, A. Blake and V. Kolmogorov, "Bilayer Segmentation of Live Video," In CVPR, p53-60, 2006.

[8] D. Freedman and T. Zhang, "Interactive Graph Cut Based Segmentation With Shape Priors," In CVPR, 2005.

[9] L. Grady, "Random Walks for Image Segmentation," IEEE Trans. PAMI, 28(11):1768-1783, 2006.

[10] L. Ingber, "Simulated Annealing: Practice Versus Theory," J. of Math. and Comp. Modeling 18(11), 29-57, 1993.

[11] S. Kumar and M. Hebert, "Discriminative Random Fields," IJCV, 68(2):179-201, 2006.

[12] A. Levin and Y. Weiss, "Learning to Combine Bottom-Up and Top-Down Segmentation," In ECCV, 2006.

[13] T. Liu, J. Sun, N. Zheng, X. Tang and H. Shum, "Learning to Detect A Salient Object," In CVPR, 2007.

[14] X. Ren and J. Malik, "Tracking as Repeated Figure/Ground Segmentation," In CVPR, 2007.

[15] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," IEEE Trans. PAMI, 22(8): 888-905, 2000.

[16] A. Sinop and L. Grady, "Uninitialized, Globally Optimal, Graph-Based Rectilinear Shape Segmentation - The Opposing Metrcs Method," In ICCV, 2007.

[17] H. Tao, H. Sawhney and R. Kumar, "Object Tracking with Bayesian Estimation of Dynamic Layer Representations," IEEE Trans. PAMI, 24(1): 75-89, 2002.

[18] D. Tolliver, G. Miller and R. Collins, " Corrected Laplasians: Closer Cuts and Segmentation with Shape Priors," In CVPR, 2005.

[19] J. Wang and E. Adelson, "Layered Representation for Motion Analysis," In CVPR, p361-366, 2003.

[20] Z. Yin and R. Collins, "Moving Object Localization in Thermal Imagery by Forward-backward MHI," In CVPR workshop on OTCVBS, 2006.

[21] Z. Yin and R. Collins, "Online Figure-Ground Segmentation with Edge Pixel Classification," In BMVC, 2008.

[22] T. Zhang and D. Freedman, "Tracking Objects Using Density Matching and Shape Priors," In ICCV, 2003.