Segmenting Highly Articulated Video Objects with Weak-Prior Random Forests

Hwann-Tzong Chen[1], Tyng-Luh Liu[1], and Chiou-Shann Fuh[2]

[1] Institute of Information Science, Academia Sinica, Taipei 115, Taiwan
{pras, liutyng}@iis.sinica.edu.tw
[2] Department of CSIE, National Taiwan University, Taipei 106, Taiwan
fuh@csie.ntu.edu.tw

Abstract. We address the problem of segmenting highly articulated video objects in a wide variety of poses. The main idea of our approach is to model the prior information of object appearance via *random forests*. To automatically extract an object from a video sequence, we first build a random forest based on image patches sampled from the initial template. Owing to the nature of using a randomized technique and simple features, the modeled prior information is considered *weak*, but on the other hand appropriate for our application. Furthermore, the random forest can be dynamically updated to generate prior probabilities about the configurations of the object in subsequent image frames. The algorithm then combines the prior probabilities with low-level region information to produce a sequence of figure-ground segmentations. Overall, the proposed segmentation technique is useful and flexible in that one can easily integrate different cues and efficiently select discriminating features to model object appearance and handle various articulations.

1 Introduction

Object segmentation has been one of the fundamental and important problems in computer vision. A lot of efforts have been made to resolve the problem, but, partly due to the lack of a precise and objective definition itself, fully-automatic unconstrained segmentation is still an "unsolved" vision task. The predicament is further manifested by the success of those characterized with clear aims, e.g., edge or interest-point detection. Nevertheless, the bottom-up segmentation approaches based on analyzing low-level image properties have been shown to achieve stable and satisfactory performances [13], [20], [28], [36], even though the segmentation outcomes (e.g., see [17]) often contextually differ from those produced by humans [16], [21]. Humans have abundant experience on the contexts of images; with our prior knowledge we can infer an object's shape and depth, and thus produce *meaningful* segmentations that are unlikely to be derived by a general-purpose segmentation algorithm.

While fully automatic image segmentation seems to be an ill-posed problem, *figure-ground segmentation*, on the other hand, has more specific goals and is easier to evaluate the quality of segmenting results. Since the emphasis is on

Fig. 1. Examples of unusual and highly articulated poses.

separating the target object(s) (of which some properties are known *a priori*) from the background, it opens up many possibilities regarding how to impose prior knowledge and constraints on the segmentation algorithms. For instance, an algorithm may choose from some predefined object models, such as deformable templates [35] or pictorial structures [11], or learn from the training data [4] to construct the object representation. Once the representation is decided, the algorithm can yield segmentation hypotheses in a top-down fashion [4], and then examines the feasibility of hypothesized segmentations. Indeed top-down and bottom-up segmentation approaches are not mutually exclusive and, when properly integrated, they could result in a more efficient framework [30], [34].

In this paper we address figure-ground segmentation for objects in video. Particularly, we are interested in establishing a framework for extracting non-rigid, highly articulated objects, e.g., athletes doing gymnastics as shown in Fig. 1. The video sequences are assumed to be captured by moving cameras, and therefore background subtraction techniques are not suitable. Furthermore, since we are mostly dealing with unusual poses and large deformations, top-down approaches that incorporate class-specific object models such as active contours [2], exemplars [14], [29], pictorial structures [11], constellations of parts [12], deformable models [22], and deformable templates [35] are less useful here—the degree of freedom is simply too high, and it would require either a huge number of examples or many parameters to appropriately model all possible configurations of the specified object category. Hence, instead of considering *class-specific* segmentation [4], we characterize our approach as *object-specific* [34]: Given the segmented object and background in the initial image frame, the algorithm has to segment the same object in each frame of the whole image sequence, and the high-level prior knowledge about the object to be applied in the top-down segmentation process should be learned from the sole example.

1.1 Previous Work

In class-specific figure-ground segmentation, the top-down mechanism is usually realized by constructing an object representation for a specified object category and then running the segmentation algorithm under the guidance of the representation. Borenstein and Ullman [4] introduce the fragment-based representation that covers an object with class-related fragments to model the shape of the

object. Their algorithm evaluates the quality of "covering with candidate fragments" to find an optimal cover as the segmenting result. Three criteria are used to determine the goodness of a cover: similarity between a fragment and an image region, consistency between overlapping fragments, and reliability (saliency) of fragments. Tu et al. [30] propose the image parsing framework that combine the bottom-up cues with the top-down generative models to simultaneously deal with segmentation, detection, and recognition. The experimental results in [30] illustrate the impressive performances of parsing images into background regions and rigid objects such as faces and text.

Despite the computational issues, Markov random fields (MRFs) have been widely used in image analysis through these years [15]. Recently several very efficient approximation schemes for solving MRFs [5], [33] and their successful use in producing excellent results for interactive figure-ground segmentation, e.g., [3], [6], [27], have made this group of approaches even more popular. Aiming for the class-specific segmentation, Kumar et al. [18] formulate the object category specific MRFs by incorporating top-down pictorial structures as the prior over the shape of the segmentation, and present the OBJ CUT algorithm to obtain segmentations under the proposed MRF model. Typically, graph-cut-based segmentation approaches use predefined parameters and image features in the energy functions [6]. Although the GMMRF model [3] allows adjustments to the color and contrast features through learning the corresponding parameters from image data, only low-level cues are considered. Ren et al. [26] propose to learn the integration of low-level cues (brightness and texture), middle-level cues (junctions and edge continuity), and high-level cues (shape and texture prior) in a probabilistic framework.

The segmentation task of our interest is related more closely to that of Yu and Shi [34], where they address the object-specific figure-ground segmentation. Given a sample of an object, their method can locate and segregate the same object under some view change in a test image. The algorithm takes account of both pixel-based and patch-based groupings through solving a constrained optimization regarding pixel-patch interactions. Still in Yu and Shi [34] the goal of segmentation is to identify rigid objects in images. We instead consider a figure-ground technique for video, and more importantly, for segmenting articulated objects with large deformations. We are also motivated by the work of Mori et al. [23] that considers detecting a human figure using segmentation. They use Normalized Cuts [28] to decompose an image into candidate segments. To generate the body configuration and the associated segmentation, their algorithm locates and then links those segments representing the limbs and torso of the target human. The experimental results reported in [23] show that the proposed technique can extract from images the baseball players in a wide variety of poses. Concerning the implementation details, their approach requires *logistic regression* to learn the weights of different cues from a set of hand-segmented image templates. In addition, several global constraints are enforced to reduce the complexity of searching a large number of candidate configurations. These constraints are indeed very strong prior knowledge defining what physically pos-

sible configurations of a human body can be, and consequently are not easy to be generalized to other object categories.

A different philosophy from those of the aforementioned approaches is using variational models [24] or level-set PDE-based methods [25] for image segmentation. Approaches of this kind are more flexible to handle deformations. However, integrating different cues or imposing top-down prior models in such frameworks is much more sophisticated, which involves adding intriguing terms accounting for the desired properties into the PDEs, and thus further complicates the numerical formulations, e.g., [8], [9]. It is also hard to include learning-based mechanisms such as parameter estimation and feature selection, to give suitable weights among low-level information and different aspects of prior knowledge.

1.2 Our Approach

Analogous to regularization for optimization problems, there is a trade-off between imposing strong prior knowledge and allowing flexibility of object configurations when we incorporate a top-down scheme into figure-ground segmentation. For the images shown in Fig. 1, class-specific shape (or structure) models, in general, are more restrictive for covering such a wide range of pose variations, otherwise the search space of possible configurations might be too large to be tractable.

We propose a new framework for object-specific segmentation that models the prior information by *random forests* [7], constructed from randomly sampled image patches. Owing to the nature of random forests, the modeled prior knowledge is *weak* but still sufficient for providing top-down probabilistic guidance on the bottom-up grouping. And this aspect of characteristic is crucial for our task. Moreover, the randomized technique also enables cue integration and feature selection to be easily achieved. We shall show that the proposed algorithm is useful and rather simple for video-based figure-ground segmentation, especially when the objects are non-rigid and highly articulated.

2 Learning Prior Models with Random Forests

In this section we first describe the image cues for constructing the prior models, and then explain the technique of embedding a prior model into a random forest. Given the template and the mask of an object, using Figs. 2a and 2b as an example, we seek to build a useful prior model with a random forest for subsequent video object segmentations. For the experiments presented in this paper, we use color, brightness, and gradient cues, though other cues such as texture and optical flow can be easily included in the same manner. Specifically, we first apply Gaussian blur to each color channel of the template as well as the gray-level intensity, and thus get the smoothed cues as illustrated in Figs. 2c and 2d. From the smoothed intensity we compute the gradient, and then further blur it to get the x and y derivatives as shown in Figs. 2e and 2f. Note that the values of all cues are normalized between 0 and 1. For convenience, these cues

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

Fig. 2. Information used for constructing the prior models. (a) Template. (b) Mask. (c) RGB color cues. (d) Brightness cue. (e) x derivatives. (f) y derivatives.

are combined to get a pseudo-image of six channels. We will treat the pseudo-image template as a pool of patches that constitute our prior knowledge about the object.

2.1 Random Forests

Random forests by Breiman [7] are proposed for classification and regression, and are shown to be comparable with boosting and support vector machines (SVMs) through empirical studies and theoretical analysis. Despite their simplicity and the effectiveness in selecting features, random forest classifiers are far less popular than AdaBoost and SVMs in computer vision, though random forests are indeed closely related to and partly motivated by the shape-recognition approach of randomized trees [1]. Random forests have been used for multimedia retrieval by Wu and Zhang [32]. More recently, Lepetit et al. [19] consider randomized trees for keypoint recognition, and obtain very promising results.

To model the prior information of an object's appearance, we generate a forest of T random binary trees. Each tree is grown by randomly sampling N patches from the pseudo-image template; we run this process T times to obtain T trees. The typical window size of a patch we use in the experiments is 5×5. Since a pseudo-image contains six channels, we actually store each patch as a vector of $5 \times 5 \times 6$ elements (see the illustration at the right hand side of Fig. 3). Let $\mathbf{x}^k$ denote a patch, and $\{\mathbf{x}^k | \mathbf{x}^k \in \mathbb{R}^{\mathsf{d}}\}_{k=1}^N$ be the sample set (hence $\mathsf{d} = 150$). For each patch $\mathbf{x}^k$ we then obtain the label information y^k from the corresponding position (patch center) in the mask. Therefore, we have the labels $\{y^k | y^k \in \{\mathsf{F}, \mathsf{G}\}\}_{k=1}^N$ that record a patch belonging to figure (F) or ground (G). To grow a tree with $\{(\mathbf{x}^k, y^k)\}_{k=1}^N$ involves random feature selection and node impurity evaluation. The tree-growing procedure is described as follows.

1. At each tree-node, we randomly select M features. In this work we consider a feature as the difference between some ith and jth elements of a patch. That is, we repeat M times choosing at random a pair of element indices i and j. With a random pair of indices i and j defining a feature, each patch $\mathbf{x}^k$ would give a feature value $f = \mathbf{x}_i^k - \mathbf{x}_j^k$.
2. For each feature, we need to determine a threshold that best splits the patches reaching the current node by their feature values. The threshold of

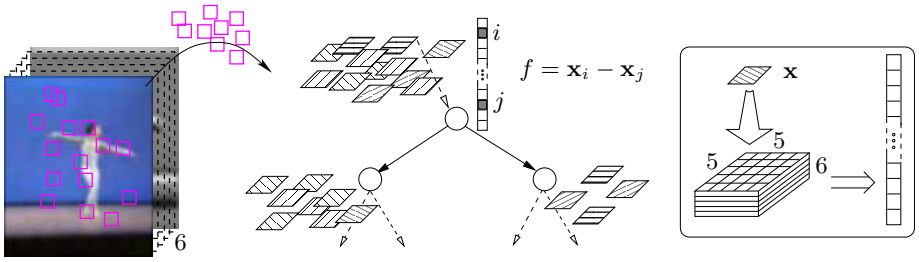


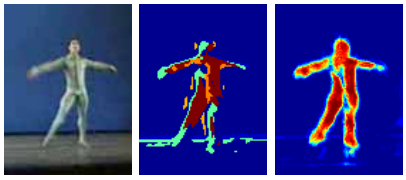
Fig. 3. Growing a random tree. The typical window size of a patch is 5×5. Since a pseudo-image contains six channels, we actually store a patch as a vector of $5 \times 5 \times 6$ elements. A feature is defined as the difference between two randomly selected elements of a patch. We use the random feature (with an optimized threshold) to split the set of patches reaching the current node into two parts forming two child nodes, according to the feature values f of patches being greater or less than the threshold.

feature values for optimal splitting is obtained by maximizing the decrease in *variance impurity* of the distribution of the patches. The idea is to *purify* each child node such that the patches in a child node would almost carry the same label. (For brevity's sake, we skip the definition of impurity and the description of maximizing the impurity drop. The details can be found in [10], p. 400.)

3. Among the M randomly selected features we pick the one that produces the maximum drop in impurity, and use this feature and its threshold to split the current patches into two child nodes according to the feature values of patches being greater or less than the threshold, as illustrated in Fig. 3.
4. We stop splitting a node (thus a leaf node) if the number of arriving patches is less than a chosen constant, or if the predefined limit of tree-height is met.
5. If there is no more node to be split, the growing process is done. Each internal node stores a feature and a splitting threshold, and each leaf node yields a probability $P(y = \mathsf{F}|\mathbf{x}$ reaches this leaf node), which is computed as the proportion of "the patches in this leaf node belonging to the figure F" to "all the patches that reach this leaf node."

The features used in the random trees cover a large number of combinations of differences between two channels with some spatial perturbations. For instance, they may represent the difference between the R and B channels of a patch, or represent two pixels being residing on or separated by an edge. Because the cues have been smoothed, the features are less sensitive to the exact positions they are selected. Moreover, these types of features can be efficiently computed; each computation costs only two memory accesses and one subtraction. Our experimental results show that they are also quite discriminating in most cases.

2.2 Prior Probabilities about the Object's Configuration



new input one tree 25 trees

Fig. 4: Prior probabilities.

After constructing a forest of T trees by repeating the preceding procedure, we have a random forest that models the appearance prior of a target object. Then for each new image frame, we scan the whole image pixel by pixel to run every corresponding patch $\mathbf{x}$ through the random forest, and average the probabilities $\{P_t(y = \mathsf{F}|\mathbf{x})\}_{t=1}^{T}$ advised by the trees. The scheme thus provides top-down probabilistic guidance on the object's configuration. Fig. 4 illustrates the prior probabilities estimated by a single tree and by 25 random trees for a new input image. Suffice it to say, with a random forest of 25 trees, the object's appearance prior can be suitably modeled using the template and mask in Fig. 2. In the next section, we will show how to combine the top-down hints of prior probabilities with the bottom-up segmentations.

3 Applying Prior Models to Segmentation

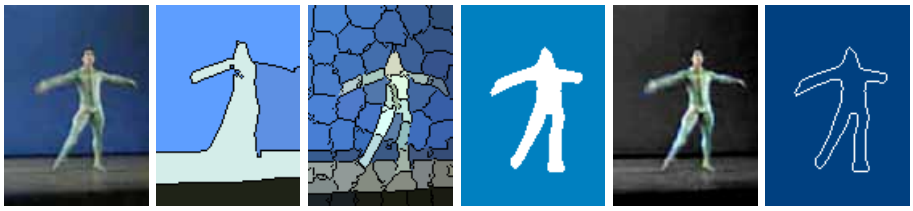
In Figs. 5b and 5c we depict two segmenting results produced by Normalized Cuts [28]: one contains 4 segments and the other 70 segments. The results show that, even though directly using Normalized Cuts with a "few-segment" setting to obtain figure-ground segmentation might not be appropriate, the Normalized Cuts segmentation that produces many regions (over-segmentation) does provide useful low-level information about the segments of the object.

3.1 Solving Figure-Ground Segmentation

The key idea is to combine the low-level information of over-segmentation with the prior probabilities derived from the random-forest prior model to complete the figure-ground segmentation. Our algorithm can produce segmenting results like the one shown in Fig. 5d. In passing, based on the segmenting results, it is straightforward to highlight the object in the video or extract its contour, e.g., see Figs. 5e and 5f. We summarize our algorithm as follows.

Voting by prior probabilities: Inside each region of the over-segmentation derived from Normalized Cuts, compute the number of pixels whose prior probabilities are above, say, one standard deviation of the mean prior probabilities. That is, a pixel having a high enough prior probability casts one vote for the region to support it as a part of the figure.

Choosing candidates: A region will be considered as a candidate if it gets more than half of the total votes by its enclosing pixels.



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

Fig. 5. (a) New input. (b) & (c) Two Normalized Cuts segmentations with 4 and 70 segments. (d) Figure-ground segmentation produced by our algorithm. Based on the segmentation result, it is straightforward to (e) highlight the object in the video, or (f) extract the contour for other uses such as action analysis.

Filling gaps: Apply, consecutively, morphological dilation and erosion (with a small radius) to all candidate regions. This will close some gaps caused by artifacts in low-level segmentation.

Merging the candidates Compute the connected components of the outcome in the previous step to combine neighboring candidate regions. Set the largest component as the figure and the rest of the image as the background.

3.2 Locating the Video Object

So far we discuss merely the single-frame case, and assume that a loose bounding-box surrounding the object is given to point out the whereabouts of the object in each frame. Thus we take only the cropped image as input for the aforementioned algorithm. For video, we bring in a simple tracker to find the loose bounding-box of the object. This is also achieved by working on the prior probabilities. We just need to search nearby area of the object's previous location in the previous image frame to find a tight bounding-box that encloses mostly high prior probabilities. We then enlarge the tight bounding-box to get a loose one to include more backgrounds. The optimal tight bounding-box can be very efficiently located by applying the technique of *integral images*, as is used in [31]. Our need is to calculate the sum of prior probabilities inside each bounding-box and find the one yielding the largest sum. For that, we compute the integral image of the prior probabilities. Then calculating each sum would require only four accesses to the values at the bounding-box's corners in the integral image.

3.3 Updating the Random Forest

Since we are dealing with video, it is natural and convenient to update the random forest based on previous observations and segmenting results. The updated random forest would be consolidated with new discriminating features to distinguish the object from the changing backgrounds. We propose to update the random forest by cutting and growing trees. The following two issues are of concern to the updating: 1) which trees should be cut, and 2) which patches could be used to grow new trees.

Cutting old trees. Because the foreground mask and the object appearance given at the beginning are assumed to be accurate and representative, we use them to assess the goodness of the trees in the current random forest. For an assessment, we sample K patches from the original template to construct $\{\tilde{\mathbf{x}}^k\}_{k=1}^K$ that all correspond to the object (with labels $\tilde{y}^k = \mathsf{F}$). We run them through the random forest of T trees and get the probabilities $P_t(\tilde{y}^k = \mathsf{F}|\tilde{\mathbf{x}}^k)$, where $t = 1, \ldots, T$ and $k = 1, \ldots, K$. In addition, we compute the average probability $\bar{P}(\cdot) = \sum_t P_t(\cdot)$ of each patch over T trees. The *infirmity* of a tree is evaluated by

$$L_t = -\sum_k \bar{P}(\tilde{y}^k = \mathsf{F}|\tilde{\mathbf{x}}^k)\ \log\ P_t(\tilde{y}^k = \mathsf{F}|\tilde{\mathbf{x}}^k)\,. \tag{1}$$

The negative logarithm of a probability measures the error made by a tree for a given patch. And the average probability $\bar{P}(\tilde{y}^k = \mathsf{F}|\tilde{\mathbf{x}}^k)$ gives a larger weight to the patch that is well predicted by most of the trees. Hence a tree will be penalized more by $\bar{P}(\tilde{y}^k = \mathsf{F}|\tilde{\mathbf{x}}^k)$ if it performs relatively poor than others on $\tilde{\mathbf{x}}^k$. We cut the top T' trees that give the largest values on L_t.

Growing new trees. We need to grow T' new trees to replace those being cut. The patches required for constructing new trees are sampled from three sources: 1) From the inside of the figure segmentation we sample the patches that are of very high prior probabilities and label them as the figure patches; 2) From the area outside the figure segmentation but inside the bounding-box, we sample the patches that are of very low prior probabilities and mark them as background patches; 3) From the area outside the loose bounding-box, we sample the patches that are of high prior probabilities, and also label them as background patches—these patches are prone to cause misclassifications.

After the updating, we have a mixture of old and new trees. The updated random forest still provides an effective prior model of the object, and becomes more robust against the varying background.

Note that our presentation in Section 3 is to first detail what needs to be done for each single frame, and then describe how to handle an image sequence. In practice, the algorithm of applying a prior model to segmenting a video object is carried out in the following order: 1) locating the bounding box, 2) solving figure-ground segmentation, and 3) updating the random forest.

4 Experiments

We test our approach with some dancing and gymnastics video clips downloaded from the Web[3]. Some of the image frames are shown in Fig. 6, as well as the prior probabilities and the figure-ground segmentations produced by our approach. The objects in these video sequences demonstrate a wide variety of poses. Many

[3] http://www.londondance.com http://www.shanfan.com/videos/videos.html http://www.rsgvideos.com

of the poses are unusual, though possible, and therefore provide ideal tests to emphasize the merits of our algorithm. Note that even though we only test on human figures, our approach is not restricted to a specific object category, and hence should be equally useful in segmenting other types of rigid or non-rigid objects.

The following are a summary of the implementation details and the parameters used. In our experiments the size of a random forest is $T = 25$. The height limit of a tree is set to 6. To grow each tree, we sample $N = 1800$ patches, each of size 5×5 as mentioned earlier, from the template image. (Specifically, the template image is enlarged and shrunk by 5% to add some scale variations. Therefore, we have three scales of the template for drawing samples; we sample 600 patches under each scale.) A typical template size is 150×100, which is also the size of the loose bounding box used in the subsequent processes to locate the target object. Recall that, at each tree node, we need to randomly select M features as a trial, we have $M = 20$ for all the experiments. Regarding updating the random forest, for each updating we cut $T' = 10$ trees and grow new ones to keep the size of forest ($T = 25$). Overall, we find the above setting of random forests can model the objects quite well.

Our current implementation of the proposed algorithm is in MATLAB and running on a Pentium 4, 3.4 GHz PC. About the running time, building the initial random forest of 25 trees takes 5 seconds. And it takes 25 seconds to produce the figure-ground segmentation for a 240×160 input image (including 15 seconds for Normalized Cuts, 5 seconds for computing the prior probabilities, and 3 seconds for updating the random forest).

5 Conclusion

We present a new randomized framework to solve figure-ground segmentation for highly articulated objects in video. Although previous works have shown that using top-down class-specific representations can improve figure-ground segmentations, such representations, which are usually built upon strong constraints and specific prior knowledge, might lack flexibility to model a wide variety of configurations of highly articulated objects. Our approach to the problem is based on modeling weak-prior object appearance with a random forest. Instead of constructing a representation for a specific object category, we analyze a video object by randomly drawing image patches from the given template and mask, and use the patches to construct the random forest as the prior model of the object. For an input image frame, we can derive the prior probabilities of the object's configuration from the random forest, and use the prior to guide the bottom-up grouping of over-segmented regions. Our experimental results on segmenting different video objects in various poses demonstrate the advantages of using random forests to model an object's appearance—a learning-based mechanism to select discriminating features and integrate different cues. For future work, we are interested in testing other filter-based cues to make our algorithm more versatile, as well as handling occlusion and multi-object segmentation.

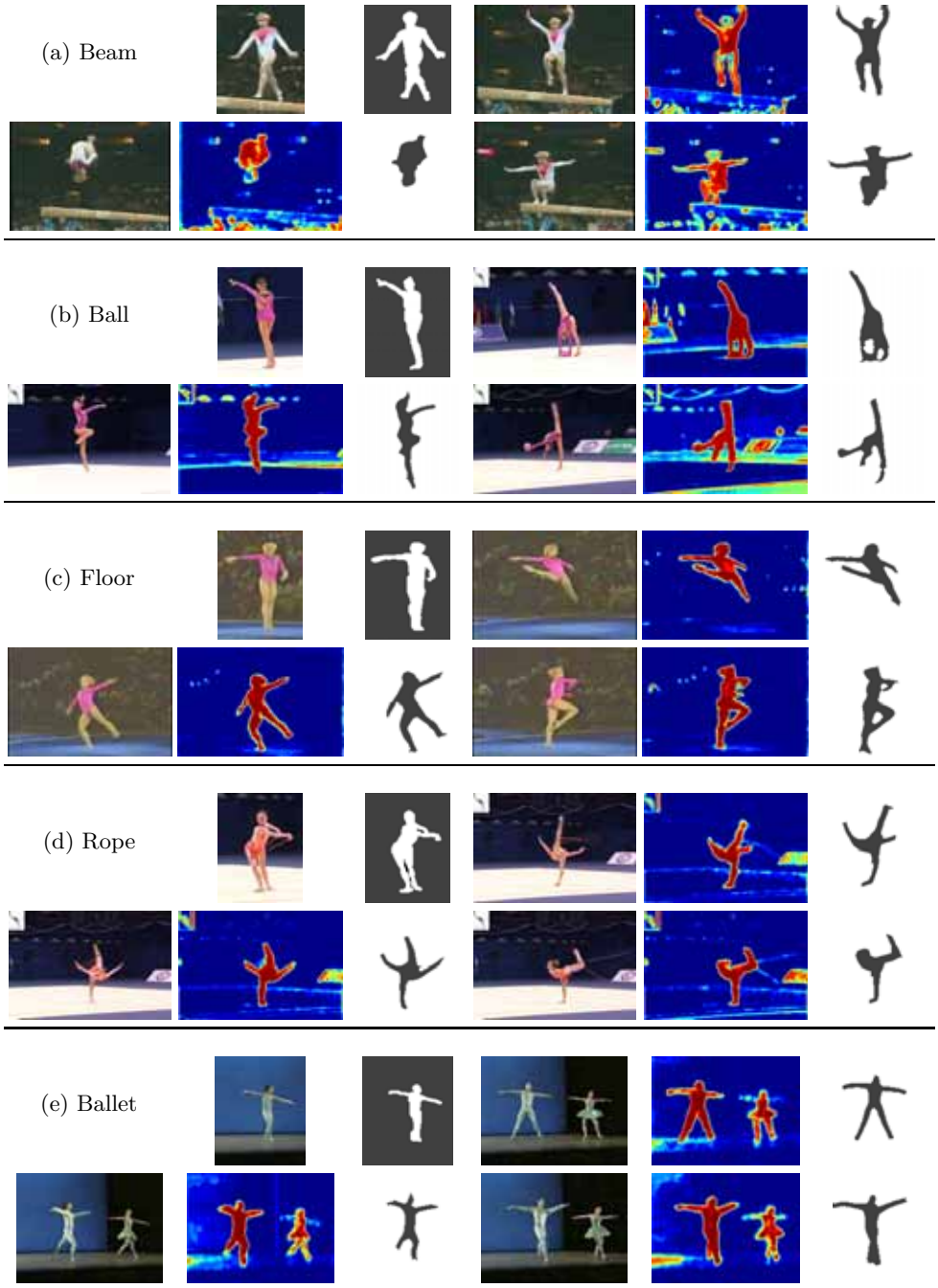


Fig. 6. The first two images shown in each of the five experiments are the template and the mask used for constructing the random forest. For each experiment we show three examples of the input frame, the prior probabilities, and the figure-ground segmentation produced by our approach.

Acknowledgements

This work was supported in part by grants NSC 94-2213-E-001-005, NSC 94-2213-E-001-020, and 94-EC-17-A-02-S1-032.

References

1. Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9:1545–1588, 1997.
2. A. Blake and M. Isard. *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer-Verlag, 1998.
3. A. Blake, C. Rother, M. Brown, P. Perez, and P.H.S. Torr. Interactive image segmentation using an adaptive GMMRF model. In *ECCV*, volume 1, pages 428–441, 2004.
4. E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV*, volume 2, pages 109–122, 2002.
5. Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *CVPR*, pages 648–655, 1998.
6. Y.Y. Boykov and M.P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *ICCV*, volume 1, pages 105–112, 2001.
7. L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
8. T. Chan and W. Zhu. Level set based shape prior segmentation. In *CVPR*, volume 2, pages 1164–1170, 2005.
9. D. Cremers, F. Tischhauser, J. Weickert, and C. Schnorr. Diffusion snakes: Introducing statistical shape knowledge into the Mumford-Shah functional. *IJCV*, 50(3):295–313, December 2002.
10. R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley, 2001.
11. P.F. Felzenszwalb and D.P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, January 2005.
12. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, volume 2, pages 264–271, 2003.
13. C.C. Fowlkes, D.R. Martin, and J. Malik. Learning affinity functions for image segmentation: Combining patch-based and gradient-based approaches. In *CVPR*, volume 2, pages 54–61, 2003.
14. D. Gavrila. Pedestrian detection from a moving vehicle. In *ECCV*, volume 2, pages 37–49, 2000.
15. S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *PAMI*, 6(6):721–741, November 1984.
16. http://www.cs.berkeley.edu/projects/vision/grouping/segbench/. *The Berkeley Segmentation Dataset and Benchmark*.
17. http://www.cs.berkeley.edu/∼fowlkes/BSE/cvpr-segs/. *The Berkeley Segmentation Engine*.
18. M.P. Kumar, P.H.S. Torr, and A. Zisserman. OBJ CUT. In *CVPR*, volume 1, pages 18–25, 2005.
19. V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *CVPR*, volume 2, pages 775–781, 2005.

20. J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, 43(1):7–27, June 2001.
21. D.R. Martin, C.C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, 2001.
22. T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: A survey. *MIA*, 1(2):91–108, 1996.
23. G. Mori, X. Ren, A.A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *CVPR*, volume 2, pages 326–333, 2004.
24. D. Mumford and J. Shah. Boundary detection by minimizing functionals. In *CVPR*, pages 22–26, 1985.
25. N. Paragios and R. Deriche. Coupled geodesic active regions for image segmentation: A level set approach. In *ECCV*, volume 2, pages 224–240, 2000.
26. X. Ren, C. Fowlkes, and J. Malik. Cue integration for figure/ground labeling. In *NIPS 18*, 2005.
27. C. Rother, V. Kolmogorov, and A. Blake. GrabCut: interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004.
28. J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, August 2000.
29. K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *ICCV*, volume 2, pages 50–57, 2001.
30. Z. Tu, X. Chen, A.L. Yuille, and S.C. Zhu. Image parsing: Unifying segmentation, detection, and recognition. In *ICCV*, pages 18–25, 2003.
31. P. Viola and M.J. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, volume 1, pages 511–518, 2001.
32. Y. Wu and A. Zhang. Adaptive pattern discovery for interactive multimedia retrieval. In *CVPR*, volume 2, pages 649–655, 2003.
33. J.S. Yedidia, W.T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. *Exploring Artificial Intelligence in the New Millennium*, pages 239–269, 2003.
34. S.X. Yu and J. Shi. Object-specific figure-ground segregation. In *CVPR*, volume 2, pages 39–45, 2003.
35. A.L. Yuille, D.S. Cohen, and P.W. Hallinan. Feature extraction from faces using deformable templates. *IJCV*, 8(2):99–111, 1992.
36. S.C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *PAMI*, 18(9):884–900, September 1996.