Toward a Unified Probabilistic Framework for Object Recognition and Segmentation

Huei-Ju Chen[1], Kuang-Chih Lee[2], Erik Murphy-Chutorian[3], and Jochen Triesch[1,4]

[1] Dept. of Cognitive Science, UC San Diego
9500 Gilman Drive, MC 0515, CA 92093-0515, USA
{hjchen,triesch}@cogsci.ucsd.edu
[2] OJOS Inc., http://www.ojos-inc.com/
kclee@ojos-inc.com
[3] Dept. of Electrical and Computer Engineering, UC San Diego
9500 Gilman Drive, MC 0407, CA 92093-0407, USA
{erikmc}@ucsd.edu
[4] Frankfurt Institute for Advanced Studies, J. W. Goethe University
Max-von-Laue-Str. 1, D-60438 Frankfurt/Main, Germany

Abstract. This paper presents a novel and effective Bayesian belief network that integrates object segmentation and recognition. The network consists of three latent variables that represent the local features, the recognition hypothesis, and the segmentation hypothesis. The probabilities are the result of approximate inference based on stochastic simulations with Gibbs sampling, and can be calculated for large databases of objects. Experimental results demonstrate that this framework outperforms a system in which object segmentation and recognition are treated as two independent processes.

1 Introduction

The recognition of objects in cluttered real-world scenes is a complicated task for computer vision systems. Traditional approaches that first try to segment a scene into its constituent objects and then recognize these objects have had little success, since accurate segmentation is often a subjective measure derived from a priori knowledge of an object. There are two potential approaches to overcoming this problem. The first approach completely ignores the segmentation problem and tries to directly detect or recognize objects in cluttered, unsegmented images. Under this approach, specific views of objects are frequently modeled as constellations of localized features [1]. Various formulations of such techniques have achieved excellent performance, and some can also work efficiently with large object databases by sharing features within and between different object models [2, 3]. The second approach tries to simultaneously segment and recognize objects, an idea which seems consistent with our current understanding of visual processing in primate brains [4] but has only recently been considered as a single inference problem [5–9]. Yu *et al.* used a graph cut framework that combined object patches with spatial configurations and low-level edge groups, to detect and segment people [9]. Leibe *et al.* integrated local cues (image patches and implicit shape models [6]) and global cues (silhouettes) to detect and segment

Fig. 1. Left:A simple scene with three objects (*Nuts Can*, *Suisse Mocha Box*, *and Becks Beer*) on a table.**Middle**: The segmentation and recognition results from our proposed algorithm **Right**: Graphical representation of the proposed Bayesian belief network.The shaded box nodes denote the evidences. The circles denote the hidden variables. The ellipses denote the hypernodes composed of hidden variables. The big plate around F_i and G_i comprises N^I number of i.i.d. F_i and i.i.d. G_i.

multiple pedestrians in crowded scenes [7]. Their system run through a series of interactive evidence aggregation steps, using implicit shape models to initialize segmentation of articulated objects, Chamfer matching to enforce global constrains, and the MDL framework to solve ambiguities between overlapping hypotheses. Tu *et al.* proposed a Bayesian framework that unites segmentation and recognition based on a Data Driven Markov Chain Monte Carlo method [8]. Using two specific detection engines, they successfully segmented and classified faces and text.

This paper proposes a novel probabilistic framework that merges object segmentation and recognition in a Bayesian belief network. Instead of looking for a joint interpretation of the whole image, we first obtain a set of promising candidates using a one-pass model [2, 10] and then evaluate each candidate sequentially using a generative approach. Our model can simultaneously process many objects using the same set of features to represent every object. We test our system on a database of cluttered scenes, and demonstrate robust object recognition and segmentation amid significant occlusions.

2 Problem Formulation

Given an observed image I, our goal is to detect the objects in the scene and segment them from the background. Our system solves this problem by the following steps. First, a set of promising object candidates are selected using a discriminative method proposed by Murphy-Chutorian and Triesch [2], and the identities of these candidates are denoted as $\{V_k, k = 1 \cdots K\}$. K is the number of object candidates. For each candidate, we construct a generative model with the same structure and use it to further evaluate whether this candidate is really present. Denote the constructed set of graphical models as $\{\Omega_k, k = 1 \cdots K; \beta\}$, Ω_k associated with the object of the identity V_k. β is the graphical structure shared by every element of $\{\Omega_k\}$ and will be described in section 3.

Given Ω_k, we want to decide whether the object of the identity V_k is present as well as to compute its segmentation. We formulate this problem in the context of Bayesian inference. The results are denoted as the object hypothesis H

and segmentation S (for simplicity, the subscript k is dropped.). This section introduces all the observed and latent variables in a model Ω_k.

2.1 Observations: I, G_i, N^I, and E

Let I be an image with $c \times r$ pixels (we use $c = 640$ and $r = 480$). Let E be a corresponding edge map obtained with the boundary detection algorithm developed by Fowlkes et al. [11].

Let N^I denote the number of detected interest points[5] in I. The properties of these interest points are represented by $\{G_i | i = 1, \cdots, N^I\}$, in which each element, G_i, is a two-tuple vector, $\{\mathbf{g}_i^g, \mathbf{l}_i^g\}$. $\mathbf{l}_i^g$ is the pixel location of the ith interest point, and $\mathbf{g}_i^g$ is a local feature vector at $\mathbf{l}_i^g$, consisting of a 40-dimensional *Gabor-jet*[6] [2]. Two Gabor-jets J_1 and J_2 can be compared by calculating the cosine of the angle between them:

$$\texttt{Sim}(J_1, J_2) = \frac{J_1^T J_2}{||J_1||||J_2||}, \tag{1}$$

where J_1^T denotes the transpose of J_1 and $||\cdot||$ is the Euclidean norm.

2.2 Object Hypothesis H and Segmentation S

Assuming there are N_o types of objects in the database, X^h, is a random variable indicating if the object V_k is present or not. An object hypothesis H can be specified by

$$H \equiv \{X^h, \mathbf{l}^h\}, \tag{2}$$

where X^h and $\mathbf{l}^h$ denote object presence and its location in the test image, respectively. Priors of $X^h, \mathbf{l}^h$ are described as follows. $P(X^h = 1) = P(X^h = 0) = 0.5$. In our system, $\mathbf{l}^h$ is computed in a 2D-Hough transform space which partitions the image space into a set of 32×32 bins [2] and then converted back in $c \times r$ space. We assume that the prior $P(\mathbf{l}^h)$ is uniformly distributed in this $c/32 \times r/32$ 2D-Hough space.

A segmentation S is represented by

$$S \equiv \{m^s, \mathbf{l}^s, c^s, \phi^s\}, \tag{3}$$

$\mathbf{l}^s$ is the position relative to the location of the object hypothesis in the image. m^s is a discrete random variable indexing which of a number of trained contours of the object V_k is present in the scene. Each index value is associated with a set of contour points that represent the positions of a trained contour relative to the reference position. To generate a contour in the training stage, we first manually choose a small set of the contour points. Then we interpolate the rest of the contour points by fitting a B-spline to these points. We repeat this process for

[5] We use the interest operator proposed in [12] with the minimum distance between interest points set to five pixels and the eigenvalue threshold set to 0.03.

[6] Our Gabor jets contain the absolute responses of complex Gabor wavelets at 8 orientations and 5 spatial scales. For details, see [2].

N^{V_k} training views of each object, constructing the value space of m^s as a set of indexes of these N^{V_k} contours. Note the superscript, V_k, allows for different objects to have a different number of contour models associated with them. c^s is the scale of the contour and we make $P(c^s)$ uniformly distributed between 0.5 and 1.5. ϕ^s, the contour score of segmentation, is a continuous random variable with a value domain between 0 and 1. Priors are all uniform distributions: e.g. $P(m^s) = \frac{1}{N^{V_k}}$, $P(\mathbf{l}^s) = 1/(rc)$.

2.3 Shared Features $\{F_i\}$

In order to expedite the process of object recognition and segmentation, we adopt the feature-sharing method proposed by [2] to cluster a large set of Gabor-jets into a shared feature vocabulary. Each cluster center corresponds to a shared feature and is associated with many different objects. In the training stage, N^f shared features are learned along with their relative displacements from the centers of the different objects. In our system, we use a vocabulary with ($N^f = 4000$) features.

Given an image I, each Gabor-jet extracted at each detected interest point will activate a shared feature. Let F be a collection of these active features and denote them as

$$F \equiv \{F_i\} \equiv \{f_i^{id}, \mathbf{l}_i^f | i = 1, ..., N^I\}. \tag{4}$$

Each individual feature, F_i, contains the following attributes: f_i^{id} denotes the shared feature identity, and $\mathbf{l}_i^f$ denotes its location. As described in the last paragraph, each shared feature has two attributes: the Gabor jet and the relative displacements from the centers of the objects. We denote these two attributes: $\mathbf{g}_i^f$ as the Gabor-jet of the shared feature of the identity f_i^{id}, and $\delta_{\mathbf{i}}^{\mathbf{f}}$ as the positions relative to the centers of all the object hypotheses that share this feature. Both $\mathbf{g}_i^f$ and $\delta_{\mathbf{i}}^{\mathbf{f}}$ are learned in the training stage. The prior distribution, $P(F_i)$, is the product of $P(f_i^{id})$ and $P(\mathbf{l}_i^f)$. We choose uniform distributions for both, i.e. $P(f_i^{id}) = (N^f)^{-1}$ and $P(\mathbf{l}_i^f) = (rc)^{-1}$.

3 Graphical Representation

The right-most diagram in figure 1 illustrates the structure of the Bayesian belief network. Given this model, the following three important posterior probability distributions can be decomposed into the equations,

$$\begin{aligned} P(F_i|\{G_i\}, H, S, E) &\propto P(F_i|H)P(G_i|F_i), \forall i \\ P(H|\{G_i\}, \{F_i\}, S, E) &\propto P(H)P(S|H)\prod_{i=1}^{N^I} P(F_i|H), \\ P(S|\{G_i\}, H, \{F_i\}, E) &\propto P(S|H)P(E|S), \end{aligned} \tag{5}$$

Each posterior probability in Equation 5 captures the problems of feature activation, object recognition, and segmentation, respectively. The probabilities on the right hand side of Equation 5 can be readily evaluated. The formulation of

each likelihood is described in Section 3.1, and the inference process by stochastic simulation is detailed in Section 3.2.

3.1 Likelihood Models

In this section we describe the conditional distributions of the graphical model Ω_k. Let $X_{f_i^{id}}$ be a Bernoulli random variable describing the presence of feature f_i^{id} in I. Let $\mathbf{d}$ be a location offset of object V_k from feature f_i^{id}. $\mathbf{d}$ is a function of object identity and shared feature identity and this function is learned during the training stage. $P(F_i|H)$ is formulated as

$$P(F_i|H) \propto \begin{cases} P(X_{f_i^{id}} = 1|X^h) \exp^{-\alpha\|\mathbf{l}^h - \mathbf{l}_i^f - \mathbf{d}\|^2}, \forall i, \text{ if } \|\mathbf{l}_i^f - \mathbf{d}\|^2 \leq R \\ \frac{1}{rcN^f} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad \text{, otherwise} \end{cases} \tag{6}$$

where $P(X_{f_i^{id}} = 1|X^h)$, the associations between object models and shared features, are learned during training [10].

$P(G_i|F_i)$ represents the likelihood of a shared feature, F_i, activated by the interest point, G_i, and can be denoted as

$$P(G_i|F_i) \propto \exp^{\mathtt{Sim}(\mathbf{g}_i^f, \mathbf{g}_i^g)} \delta(\|\mathbf{l}_i^f - \mathbf{l}_i^g\|), \forall i. \tag{7}$$

$P(S|H)$ represents the probability distribution of the segmentation, S, given an object hypothesis, H. $P(S|H)$ can be denoted as

$$P(S|H) \propto P(\phi^s|X^h) \sum_{(x,y)^T \in A} \delta(\|\mathbf{l}^s - (x,y)^T\|), \tag{8}$$

where A is a 2D discrete space $\{-L_a, \cdots, +L_a\}^2$. $L_a = 10$ in current implementation. We assume both $P(\phi^s|X^h = 0)$ and $P(\phi^s|X^h = 1)$ are Gaussian distributions parametrized by (μ_0, σ_0) and (μ_1, σ_1) respectively, which are learned in the training stage. The learning algorithm of these two distributions is explained as follows. Given a training image, we used the same one-pass system [2] to compute a set of image locations that each object is most likely present, i.e. each object is associated with a image location, no matter whether the object is present or not. Then for each object, we used our trained contours to match edges in the image around the neighborhood of the associated location. We also allow some transitions and scale variations during matching. Then we take the average of the edge values at every point of a contour and call this a contour score. This matching is repeated for all training images and for each object. In the end we obtain two distributions of the contour score for each object, one associated with its presence and the other associated with its absence. In this paper, we will show that the associations between contour scores and object presence help to improve recognition.

$P(E|S)$ represents the likelihood of the segmentation, S, given an edge map, E. Let $\mathtt{Edge}(E, S)$ compute the sum of edge values of E only at every point location of the contour of identity m^s, at a reference point $\mathbf{l}^s$ and scaled by c^s. Let $\mathtt{Length}(m^s)$ be the number of contour points of the contour m^s. $P(E|S)$ can be denoted by

$$P(E|S) \propto \exp^{\frac{1}{\mathtt{Length}(m^s)}\mathtt{Edge}(E,S)}. \tag{9}$$

3.2 Stochastic Simulation

So far we have presented our belief network. We use stochastic simulations for approximate inference and the simulation is based on Gibbs sampling, a scheme of Markov Chain Monte Carlo, which is commonly used to approximate the Bayesian inference of a high-dimension probability distribution. Our algorithm has two steps. First, we obtain a set of promising object candidates based on Murphy-Chutorian and Triesch's system [2]. The number of the candidates, denoted by K, determines the number of Gibbs sampling processes we need to run. We construct K graphical models, $\{\Omega_k\}$, each of which associated with one candidate. Second, we perform Gibbs sampling for each model to decide presence of each candidate and if present, its segmentation. The sampling process is described as follows. When starting a new Gibbs sampling process, we use the location of the candidate to initialize $\mathbf{l^f}$. Feature nodes are initialized in the following way: for all i, the ith feature node is initialized by $\arg\max_{F_i} P(G_i|F_i)$. Then we draw a sample by drawing its components from their full conditional distributions. $T + T_m$ samples are drawn sequentially. The first T samples are discarded and the next T_m samples are used to compute expectation of the state value. Denote this expectation as θ^*. If X^h component of θ^* is larger than 0.5, we recognize this object and segment it simultaneously. The algorithm is summarized in Box 1.

3.3 Resolving Partial Occlusion

After all objects have been detected and segmented in the input image, the partial occlusion can be further resolved by checking the edge consistency at the boundaries of all the overlapped areas between each pair of objects. Ideally the contour of a frontal object has stronger edge responses than that of the occluded object does in the overlapped areas. By checking edge consistency we can remove the overlapped areas from the segment of the occluded object and achieve a better segmentation result. The detailed algorithm is summarized in Figure 1.

4 Experimental Results

We evaluate our probabilistic framework using the CSCLAB [2] database, which consists of 1000 images of ($N_o = 50$) everyday objects with significant occlusions among 10 different cluttered backgrounds (See supplementary materials). The 1000 images are composed of 500 single object scenes and 500 multiple object scenes. Each multiple-object scene has 3 – 7 objects. Each object model is trained on all single-object scenes and the first 200 multiple-object scenes in which it appears. All the objects are presented in different scales and positions and in slightly different viewing directions. Scale varies over more than an octave.

Figure 1 and Figure 2 display several segmentation and recognition results in different cluttered scenes. These results demonstrate the capability of our system to deliver precise recognition and segmentation under difficult conditions that include significant partial occlusions and scale variations. Segmentation results are currently evaluated by human and about 60% of the results are rated as good

Obtain a set of promising object hypotheses only using jet features

We pick a set of K object candidates based on the system proposed by Murphy-Chutorian and Triesch [2]. A candidate is picked if the posterior probability of its presence of is larger than 0.1. For each object candidate, k, we construct a graphical model, Ω_k, to further verify whether this object is present or not in an input image I.

Approximate the joint distribution of each model

Begin
Set a counter $k = 1$.
In model Ω_k,

Observations: (I, E, $\{G_i, i = 1 \cdots N^I\}$)
I: input image; E: the edge map computed from I; $\{G_i, i = 1 \cdots N^I\}$: the Gabor-jets at interest points

Latent Variables: (H, S, $\{F_i, i = 1 \cdots N^I\}$)
H: the object hypothesis; S: the segmentation; $\{F_i, i = 1 \cdots N^I\}$: a set of active features

Set L, the number of the objects found in the input image I, to 0.

Gibbs sampling
In model Ω_k,

1. Initialization: Set the iteration counter $t = 1$ and set initial values $\theta^0 = (\theta^0_{F_1}, \cdots, \theta^0_{F_{N^I}})$, where $\theta^0_{F_1}, \cdots, \theta^0_{F_{N^I}}$ are assigned sequentially by $\arg\max_{F_1} P(G_1|F_1), \cdots, \arg\max_{F_{N^I}} P(G_{N^I}|F_{N^I})$. We initialize θ^0_S as follows. m^s is an integer randomly picked between 1 and N^{V_k}. The initial relative position, $\mathbf{l^s}$, is set to the location of the $k-$th object candidate. c^s is set to 1 and phi^s is set to 0.2, which is close to the average contour score for the training images. To initialize θ^0_H, we set the value of X^h to 1 and set the value of $\mathbf{l}^h$ to the location of the $k-$th object candidate.
2. Obtain a new value $\theta^t = (\theta^t_F, \theta^t_H, \theta^t_S)$ from θ^{t-1} through successive generation of values
   $\theta^t_{F_i} \sim P(F_i|\theta^{t-1}_H, \theta^{t-1}_S, G, E), \forall i$
   $\theta^t_H \sim P(H|\theta^{t-1}_S, \{\theta^t_{F_i}\}, G, E)$
   $\theta^t_S \sim P(S|\theta^t_H, \{\theta^t_{F_i}\}, G, E)$
3. Change counter t to $t + 1$ and return to the previous step until $t > T + T_m$. (T=3000 and m=1000).

Performing Recognition and Segmentation: Discard the first T samples and compute the expectation value of the state using the next T_m samples. Denote this expectation as θ^*_k. If the X^h component of θ^*_k is larger than 0.5, increase L by one and θ^*_k is a new detection. Set $k = k + 1$. Loop back to Step 1 to repeat Gibbs sampling for the next model until $k > K$.
End

Resolving Partial Occlusion (Optional): Resolve the partial occlusion by checking the edge consistency in the boundary of the overlapped areas between each pair of detected objects.

Box 1. Summary of the proposed stochastic simulation algorithm

Fig. 2. Recognition and Segmentation results: the red lines depict the contour of each object. The partial occlusion has been resolved correctly for each object

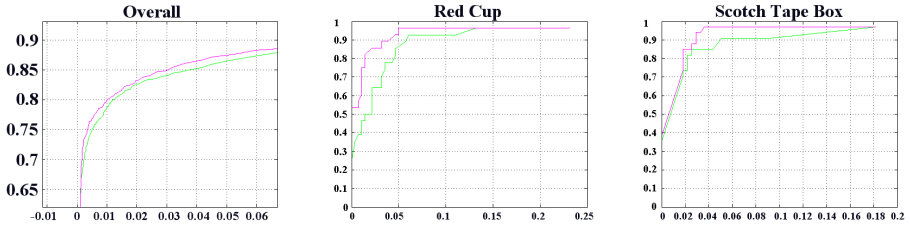

Fig. 3. ROC curves of the integrated system (*pink line*) and the recognition-alone system (*green line*) **Left**: Overall performance **Middle**: Performance comparison with respect to the "Red Cup" object **Right**: Performance comparison with respect to the "Scotch Tape Box" object

results. Also note that some of the transparent objects in our database, such as the clear cup and the water bottle, can still be segmented and recognized well under this framework.

The ROC curves in Figure 3 compare the performance of the unified model with the feed-forward recognition system that has no segmentation. Our unified segmentation and recognition system performs well with a 87.5% true positive rate at a 5% false positive rate on the difficult 50 object detection task. As can be seen, by integrating segmentation, we achieve an increase in the detection rate for any fixed false positive rate. Interestingly, the performance increase is quite significant for some objects that were hard to recognize because of little texture, small sizes, or strong occlusions. For example, for the object "Red Cup", the detection rate is improved from 39% to 60% at 1% false positive rate, and from 87% to 96% at 5% false positive rate. For the object "Scotch Tape Box", the detection rate is increased from 74% to 85% at 2% false positive rate, and from 85% to 96% at 4% false positive rate. Most objects show some improvement under our integrated segmentation and recognition framework.

The average computation time is around 9 minutes per image (640x480 resolution) on a standard Intel Pentium-4 CPU (2.40GHz) machine.

5 Conclusion

We proposed a novel probabilistic framework that integrates image segmentation and object recognition based on a Bayesian belief network. Our model consists of three latent nodes: *object hypothesis*, *segmentation*, and *wavelet features*. The joint distribution is approximated by Gibbs sampling. Because the learned object models provide a very good initial belief about object hypotheses in a very fast feed-forward fashion, the simulated distribution more likely converge to the true distribution in reasonable steps. This property would be even more beneficial when in the future we extend the state space of our model, such as allowing rotations in contour matching, or allowing part of the contour points to move in order to achieve better segmentation and recognition. Due to the shared feature vocabulary, our system is scalable to recognizing large numbers of objects. Experimental results demonstrate that our method outperforms a feed-forward version of the system that does not try to segment the objects. Our probabilistic framework can easily incorporate different types of features for both recognition and segmentation, which could further improve performance.

Our current system can be extended to perform full 3-D object recognition and segmentation (in contrast to the single pose version described here) by simply adding more training images of different poses of each object. An important direction for further research is to develop a method for learning the contour models without manual segmentation of training images. The segmentation result can be further improved using active contour algorithms such as Snakes [13] and Deformable Templates [14].

Acknowledgments. Parts of this research were supported by NSF under grant IIS-0208451.

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
2. Murphy-Chutorian, E., Triesch, J.: Shared features for scalable appearance-based object recognition. In: Proc. of IEEE Workshop on Applications of Computer Vision (WACV 2005), Breckenridge, Colorado, USA (2005)
3. Torralba, A.B., Murphy, K.P., Freeman, W.T.: Sharing features: Efficient boosting procedures for multiclass object detection. In: Computer Vision and Pattern Recognition (CVPR). Volume 2. (2004) 762–769
4. Peterson, M.A.: Shape recognition can and does occur before figure-ground organization. Current Directions in Psychological Science **3** (1994) 105–111
5. Leibe, B., Schiele, B.: Interleaved object categorization and segmentation. In: British Machine Vision Conference (BMVC). (2003)
6. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV'04 Workshop on Statistical Learning in Computer Vision, Prague, Czech Republic (2004) 17–32

7. Leibe, B., Seemann, E., Schiele, B.: Pedestrian detection in crowded scenes. In: Computer Vision and Pattern Recognition (CVPR). (2005) 878–885
8. Tu, Z., Chen, X., Yuille, A., Zhu, S.C.: Image parsing: Segmentation, detection, and object recognition. In: International Conference on Computer Vision (ICCV). (2003)
9. Yu, S., Gross, R., J. Shi, a.: Concurrent object segmentation and recognition with graph partitioning. Proceedings of Neural Information Processing Systems (2002)
10. Murphy-Chutorian, E., Aboutalib, S., Triesch, J.: Analysis of a biologically-inspired system for real- time object recognition. Cognitive Science Online (accepted paper) (2005)
11. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color and texture cues. Pattern Recognition and Machine Intelligence **26** (2004) 530–549
12. Shi, J., Tomasi, C.: Good features to track. Proc. of IEEE Conf. on computer Vision and Pattern Recognition (CVPR) (1994)
13. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. In: International Conference on Computer Vision (ICCV). (1987) 259–268
14. Blake, A., Isard, M.: Active Contours. Springer (1998)