Object-Specific Figure-Ground Segregation

Stella X. Yu
Robotics Institute
Carnegie Mellon University
Center for the Neural Basis of Cognition
Pittsburgh, PA 15213-3890
stella.yu@cs.cmu.edu

Jianbo Shi
Dept. of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389
jshi@cis.upenn.edu

Abstract

We consider the problem of segmenting an image into foreground and background, with foreground containing solely objects of interest known a priori. We propose an integration model that incorporates both edge detection and object part detection results. It consists of two parallel processes: low-level pixel grouping and high-level patch grouping. We seek a solution that optimizes a joint grouping criterion in a reduced space enforced by grouping correspondence between pixels and patches. Using spectral graph partitioning, we show that a near global optimum can be found by solving a constrained eigenvalue problem. We report promising experimental results on a dataset of 15 *objects under clutter and occlusion.*

1. Introduction

The problem we are to solve is illustrated in Fig. 1: partition an image into foreground and background, with objects of interest in the foreground and unknown clutters in the background.

Figure 1. Object-specific figure-ground segregation.

In most object segmentation formulations, only one type of objects is considered and object knowledge is mainly employed to overcome data noise. It is often used when the object of interest is known to be present and some initial estimation of the size and location can be obtained. For example, in the deformable template approach [9], a deformable prototype is used with a deformation space modeled from training data. Some well-known applications are: detecting the eye and mouth [11], tracking shapes in motion [1], and segmenting anatomical parts in medical images [5].

An alternative to deformable templates for object segmentation is proposed in [2]. Instead of a globally constrained template, object knowledge is represented using pairs of image fragments and their figure-ground labeling from a training set. An energy function is formulated for segmenting a test image so that it can be covered by a set of fragments whose appearances match with the data and whose labeling are locally compatible.

This exemplar-based approach is appealing for its flexible representation. However, the authors only showed results on low-resolution (40×30) images, each of which has an object occupying the center, with little background. There are a few problems not easily addressed in their framework. 1) Hallucination: if falsely detected fragments happen to align well locally, then there is no way to prevent a wrong segmentation. This occurs very often when the background has significant clutter. 2) Imprecision: since the local segmentation of training examples is used for any test image, details of region boundaries are inevitably lost. 3) Single object: the energy function defined might be suitable for one object present in the image, not for multiple objects from the same or different classes. It is not trivial to relate their cover scores from different objects so that partial covers from multiple objects are always inferior to a whole cover for one true object.

All these top-down object segmentation methods require image data to conform to object models, whether encoded in templates or fragments. Here, adopting image patches as a representation, we propose a parallel segmentation and recognition system that also addresses the above mentioned shortcomings of [2].

Our basic idea is that image segmentation should take both low-level feature saliency and high-level object famil-

iarity [6] into account. With the guidance of object knowledge, segmentation would not get lost in imaging noise and background clutter. With the verification of low-level feature saliency, we prevent the hallucination of falsely detected object parts standing out from their surroundings.

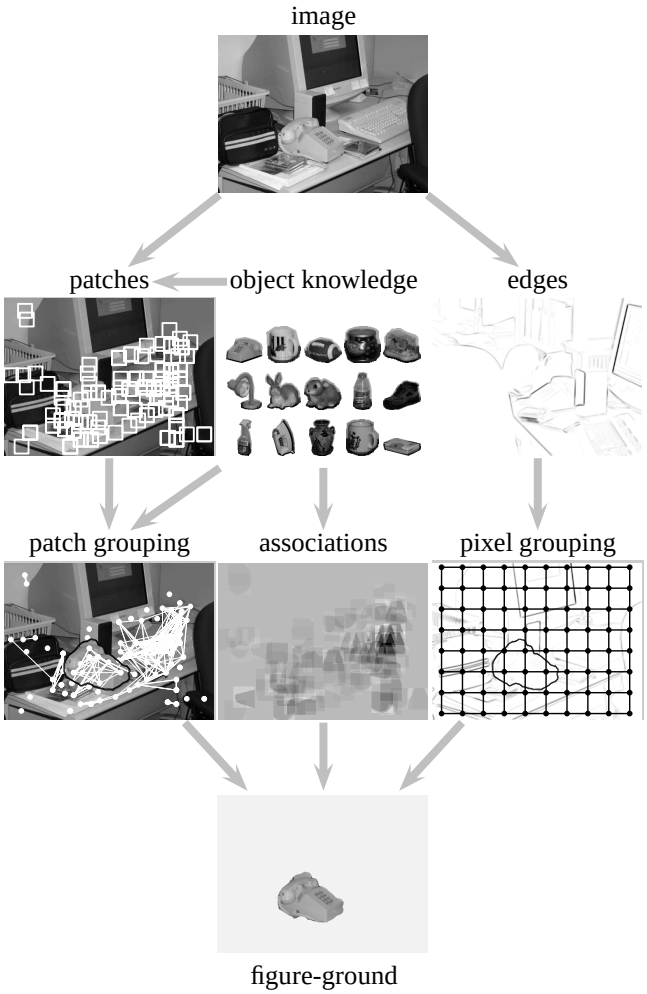


Figure 2. Our approach. A set of patches are detected and identified with object parts in a training set. Patch grouping is to find patches of consistent spatial configurations. Every object part hypothesis is also associated with a local segmentation of pixels. Here shows an overlay of such local segmentations. Dark for figure, white for ground, gray for the noncommittal. At the low-level, edges are first detected. Pixel grouping is to find pixels of similar intensities. Object specific figure-ground segregation is obtained by coupling the patch and pixel grouping in their solution space, enforcing the consistency endowed by the patch-pixel association.

We formulate our method in a graph theoretic framework. Illustrated in Fig. 2, we first detect patches and edges, and then we build two relational graphs for patch and pixel grouping. They share the same representation and grouping criterion, except that the former has patches as nodes and hypothesis compatibility as affinity between two nodes, while the latter has pixels as nodes and feature similarity as affinity between two nodes. We aim at optimizing a combined grouping criterion, in a reduced solution space where patch-pixel correspondence is encoded. These constraints facilitate figure-ground segregation to produce object patches and their pixels in the foreground group, and the rest in the backgroud group. Built upon our earlier work on constrained cuts [10], we can solve near-global optimal solutions efficiently.

2. Integration Model

In this section, we focus on the integration problem, i.e., given pixel grouping cues, patch grouping cues, pixel and patch correspondence cues, how do we integrate them to make a global segregation? We will illustrate what these cues represent here, and defer the discussion on how to get these cues till the next section.

2.1. Representation: Affinity and Indicators

We take a graph-theoretic approach. A relational graph is specified by the following: nodes, edges and their associated weights. Nodes represent the elements to be grouped. Every pair of nodes are connected by an edge, with a weight describing the likelihood that the two elements belong in one group. We assume this weight is nonnegative and symmetric. If we have N nodes and M other nodes, then the pairwise relationships between them can be captured in an $N \times M$ matrix, called the *affinity matrix*. When they are the same set of nodes, the affinity matrix becomes symmetric.

We formulate our grouping problem by graph $\mathbb{G} = \{\mathbb{V}, \mathbb{U}; A, B, C\}$, where node set $\mathbb{V} = \{1, \cdots, N\}$ denotes a total of N pixels, node set $\mathbb{U} = \{N+1, \cdots, N+M\}$ denotes a total of M patches, affinity matrix $A_{N\times N}$ denotes pixel similarity, affinity matrix $B_{M\times M}$ denotes patch compatibility, association matrix $C_{N\times M}$ denotes pixel-patch mutual ownerships. See Fig. 3.

Object-specific figure-ground segregation now becomes a node partitioning problem. Given node set $\mathbb{V}$, let $\Gamma^2_{\mathbb{V}} = \{\mathbb{V}_1, \mathbb{V}_2\}$ denote a division of $\mathbb{V}$ into two disjoint sets: $\mathbb{V} = \mathbb{V}_1 \cup \mathbb{V}_2$, $\mathbb{V}_1 \cap \mathbb{V}_2 = \varnothing$. We want to find $\Gamma^2_{\mathbb{V}}$ and $\Gamma^2_{\mathbb{U}}$ so that $\mathbb{V}_1$ and $\mathbb{U}_1$ contain object pixels and patches, $\mathbb{V}_2$ and $\mathbb{U}_2$ have the rest of pixels and patches as background.

We introduce *probabilistic group indicators* to represent a partition. Let $X = [X_1, X_2]$, where $X_k(i) = \Pr(i \in \mathbb{V}_k)$. For bipartitioning, $X_1(i) + X_2(i) = 1, \forall i$. This renders X_1 a sufficient descriptor for $\Gamma^2_{\mathbb{V}}$. Similarly, we define $Y = [Y_1, Y_2]$ for patch grouping $\Gamma^2_{\mathbb{U}}$.

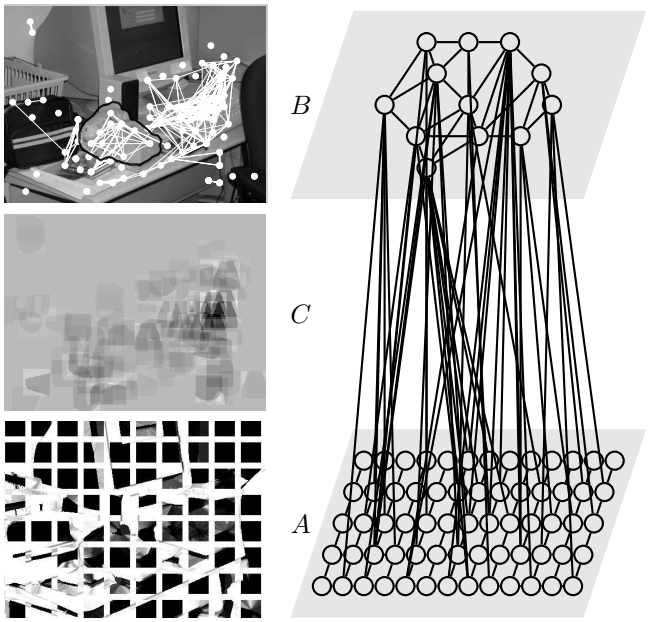


Figure 3. Graph representation. We have N pixels in an image, and M patches given by an object part detection system. They each becomes a node in the graph on the right. A: connections between pixels. They are derived from an edge map, with low value for pixels on the two sides of an edge. The affinities of a set of regularly spaced pixel nodes to their neighbours are superimposed on the edge map. Darker for larger values. B: connections between patches. They are derived from object models, with low value for patches misaligned spatially for the same object hypothesis. Thicker lines for larger affinity. C: connections between pixels and patches. Each pixel has different associations to different patches. Here is a summary of associations to all patches. Darker means larger association to an object part.

2.2. Criterion: Goodness of Grouping

Partitioning within $\mathbb{V}$ or $\mathbb{U}$ itself is a basic grouping problem, for which we adopt the normalized cuts criterion in [7]. Take pixel grouping as an example. Let's define the *connections* $\mathcal{C}$ between node sets $\mathbb{O}, \mathbb{O}' \subset \mathbb{V}$ to be the total weights on edges linking them:

$$\mathcal{C}(\mathbb{O}, \mathbb{O}'; A) = \sum_{i \in \mathbb{O}, j \in \mathbb{O}'} A(i, j). \tag{1}$$

The *degree* $\mathbb{D}$ of a set is its connections to all the nodes:

$$\mathcal{D}(\mathbb{O}; A) = \mathcal{C}(\mathbb{O}, \mathbb{V}; A). \tag{2}$$

A good partition has most of the weights contained within the set itself. This can be measured in *connection ratio* $\mathcal{R}$:

$$\mathcal{R}(\mathbb{O}; A) = \frac{\mathcal{C}(\mathbb{O}, \mathbb{O}; A)}{\mathcal{D}(\mathbb{O}; A)}.$$

Since we want to find a bipartition, the normalized cuts criterion ε aims to maximize the sum of the connection ratios for the two sets:

$$\varepsilon(\Gamma^2_{\mathbb{V}}; A) = \sum_{k=1}^{2} \mathcal{R}(\mathbb{V}_k; A). \tag{3}$$

Likewise, for the case of patch grouping, we desire a partition that maximize $\varepsilon(\Gamma^2_{\mathbb{U}}; B)$.

Given $\Gamma^2_{\mathbb{V}}$ and $\Gamma^2_{\mathbb{U}}$, our joint criterion $\bar{\varepsilon}$ takes both individual goodness and relative importance into account:

$$\bar{\varepsilon}(\Gamma^2_{\mathbb{V}}, \Gamma^2_{\mathbb{U}}; A, B) = \sum_{k=1}^{2} \mathcal{R}(\mathbb{V}_k; A) \cdot \frac{\mathcal{D}(\mathbb{V}_k; A)}{\mathcal{D}(\mathbb{V}_k; A) + \mathcal{D}(\mathbb{U}_k; B)} + \sum_{k=1}^{2} \mathcal{R}(\mathbb{U}_k; B) \cdot \frac{\mathcal{D}(\mathbb{U}_k; B)}{\mathcal{D}(\mathbb{V}_k; A) + \mathcal{D}(\mathbb{U}_k; B)}. \tag{4}$$

A is weighed less if its values are in general less than B. Note that this is different from a convex combination of $\varepsilon(\Gamma^2_{\mathbb{V}}; A)$ and $\varepsilon(\Gamma^2_{\mathbb{U}}; B)$. Here we link $\mathbb{V}_1$ to $\mathbb{U}_1$, $\mathbb{V}_2$ to $\mathbb{U}_2$, and weight them separately. In fact, we have:

$$\bar{\varepsilon}(\Gamma^2_{\mathbb{V}}, \Gamma^2_{\mathbb{U}}; A, B) = \varepsilon\left(\Gamma^2_{\mathbb{V} \cup \mathbb{U}}; \begin{bmatrix} A & \\ & B \end{bmatrix}\right). \tag{5}$$

We introduce some further notation. For any nonnegative matrix A, let D_A denote its degree matrix. It is a diagonal matrix with $D_A(i, i) = \sum_j A(i, j), \forall i$. When group indicators are binary, we have:

$$\bar{\varepsilon}(X, Y; A, B) = \sum_{k=1}^{2} \frac{Z_k^T W Z_k}{Z_k^T D_W Z_k}, \tag{6}$$

$$Z = \begin{bmatrix} X \\ Y \end{bmatrix}, W = \begin{bmatrix} A & \\ & B \end{bmatrix}, D_W = \begin{bmatrix} D_A & \\ & D_B \end{bmatrix}. \tag{7}$$

We use this formula to extend the definition of $\bar{\varepsilon}$ to the real domain so that it gives a meaningful measure when X and Y are probabilistic.

2.3. Criterion: Feasibility of Grouping

Measurement $\bar{\varepsilon}$ still leaves pixel grouping and patch grouping under-constrained, which can lead to uninterpretable solutions in our object-specific figure-ground segregation. Ideally, patches in $\mathbb{U}_1$ bring their pixels into $\mathbb{V}_1$, and vice versa. When such grouping correspondence is enforced, we have a smaller but meaningful set of segmentations to look at. Among these feasible solutions, the one yielding the best $\bar{\varepsilon}$ corresponds to the desired grouping.

If spatial configuration of some patches are consistent with their part labels, then they are in $\mathbb{U}_1$. This decision so far has nothing to do with the low-level pixel grouping; it

is entirely based on high-level object models. However, the implication of this patch grouping on pixels is clear. Pixels claimed by the patches are more likely to be in $\mathbb{V}_1$, regardless of their dissimilarity in low level features. Conversely, pixels of similar features are in $\mathbb{V}_1$ in pixel grouping, the patches claiming these pixels are more likely to be in $\mathbb{U}_1$, regardless of their incompatible hypotheses. See Fig. 4. This happens naturally for falsely detected parts which rarely have boundary support in low-level features, therefore we can pull them into background easily.

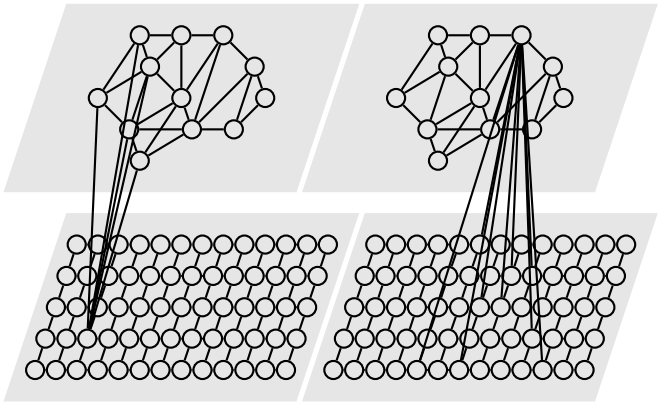


Figure 4. Correspondence constraints between high-level and low-level grouping. The foreground probability of nodes at one level are influenced by their affinitive neighbours at the other. Left: when some patches consistently indicate an object, their common pixels are likely to be foreground; when in conflict, they compete to claim their common pixels with their association strengths. Right: for pixels within a coherent region, they unanimously bring their common patch into the corresponding group. Otherwise, the patch is drawn to the identity of its dominant pixels.

Such double competition between high-level grouping and low-level grouping can be captured in constraints on the two group indicators X and Y through the association affinity C. We first encode the between-patch competition by re-weighting among patches:

$$\bar{C}(i,p) = C(i,p) \cdot \frac{C(i,p)}{\max_q C(i,q)}. \tag{8}$$

For pixel i, its association with patch p does not change if it is the strongest among all the patches; otherwise, $C(i,p)$ gets damped by its proportion to the maximum weight, weak connections becoming even weaker.

After the non-maximum suppression among patches, we consider between-pixel competition by normalizing weights among pixels:

$$Y = D_{\bar{C}}^{-1}\bar{C}X. \tag{9}$$

This equation links the probabilities for nodes in one set to the other. For example, given the foreground probability of every pixel, the foreground probability of a patch is the expectation of those of its member pixels. If the majority of these pixels are in $\mathbb{V}_1$, then this patch as well as any other patch claiming most of these pixels is probably in $\mathbb{U}_1$. Eqn. (9) can be rewritten as

$$LZ = 0, \quad L = [D_{\bar{C}}^{-1}\bar{C}, -I]. \tag{10}$$

where I is an identity matrix of appropriate dimensions and L is assumed full rank.

2.4. Solution: Constrained Optimization

Putting the goodness and feasibility of grouping together, we have a constrained optimization problem:

$$Z^* = \arg\max \bar{\varepsilon}(Z;W), \quad \text{subject to} \quad LZ = 0. \tag{11}$$

Our low-level pixel grouping and high-level patch grouping are coupled in their solution space through pixel-patch interactions. We have a modular computational framework, yet it is not at all feedforward.

Note that our formulation is not the same as maximizing $\varepsilon(\Gamma_{\mathbb{V}}^2; A + \bar{B})$, which is a simple addition of two grouping processes, with the patch affinity B converted into an equivalent pixel affinity matrix $\bar{B} = (D_{\bar{C}}^{-1}\bar{C})^T B (D_{\bar{C}}^{-1}\bar{C})$ using the constraint in Eqn. (9).

With $z = Z_1 - \alpha$, where $\alpha = \frac{Z_1^T D_W Z_1}{1^T D_W 1}$ is a partition-dependent constant, we turn the above into a constrained eigenvalue problem:

$$z^* = \arg\max \frac{z^T W z}{z^T D_W z}, \quad \text{subject to} \quad Lz = 0. \tag{12}$$

The solution is the eigenvector with the largest nontrivial eigenvalue $\lambda < 1$:

$$QD_W^{-1}Wz^* = \lambda z^*, \tag{13}$$

where Q is a projector onto the feasible solution space:

$$Q = I - D_W^{-1}L^T(LD_W^{-1}L^T)^{-1}L. \tag{14}$$

We avoid explicitly computing Q by calculating $QD_W^{-1}Wz^*$ directly in the iteration of an eigensolver. Usually the rank of L is small, so it can be done with little increase in time and space complexity.

Here is an overview of our algorithm.

1: Detect edges.
2: Evaluate pixel feature similarity A.
3: Detect patches.
4: Evaluate patch consistency B.
5: Evaluate pixel-patch ownership C.
6: Form D_W and L.
7: Solve $QD_W^{-1}Wz^* = \lambda z^*$.
8: Threshold z^* for a discrete segmentation.

3. Implementations

An image is first convolved with quadrature pairs of oriented filters f's to extract orientation energy OE [4]. Let $\underline{i}$ denote the location of pixel or patch i. Pixel affinity A is inversely correlated with the maximum magnitude of edges crossing the line connecting two pixels:

$$A(i,j) = \exp\left(-\frac{1}{2\sigma_e^2}\cdot\left[\frac{\max_{t\in(0,1)} OE(\underline{i}+t\cdot\underline{j})}{\max_k OE(\underline{k})}\right]^2\right),$$
$$OE = (g*f_{odd})^2 + (g*f_{even})^2. \quad (15)$$

$A(i,j)$ is low if i, j are on the two sides of a strong edge.

For patches, we use the results provided in [3]. Shown in Fig. 5, parts are represented by exemplars sampled from a few angles and scales. Local color, intensity, and orientation histograms are computed as features.

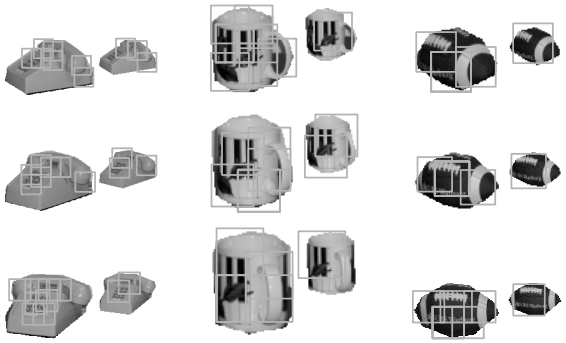


Figure 5. Samples of object parts in a training set.

Based on an optimal distance measure d learned from a training set in order to maximize discrimination among objects, patch p is labeled with the nearest neighbour p' with score $d(p,p')$, see Fig. 6. There could be multiple patches detected at the same location, corresponding to multiple object part hypotheses of the same local area in the image.

We measure the quality of detected patches in two terms. One is its own credibility denoted in a diagonal matrix E_C:

$$E_C(p,p) = \exp\left(-\frac{1}{2\sigma_p^2}\left[\frac{d(p,p')-d_{min}}{d_{max}-d_{min}}\right]^2\right), \quad (16)$$

where d_{max} and d_{min} are the minimum and maximum d values of all patches in image g. The more similar patch p is to p', the better the confidence. The other is its compatibility with nearby patches. Let $S(p',p)$ be the binary object silhouette of the training image to which part p' belong, registered to the location p in image g. Two patches p and q are consistent, if $S(p',p)$ and $S(q',q)$ overlap well. This measure increases with the distance between p and q:

$$E_S(p,q) = \exp\left(-\frac{1}{2\sigma_s^2}\left[1-\frac{||S(p',p)\wedge S(q',q)||_1}{||S(p',p)\vee S(q',q)||_1}\right]^2\right)$$
$$\cdot\left(1-\exp\left(-\frac{1}{2\sigma_d^2}\cdot-\frac{||\underline{p}-\underline{q}||_2^2}{r(p)\cdot r(q)}\right)\right), \quad (17)$$

where $\wedge$ and $\vee$ are logical and/or operations, $||\cdot||_k$ is L_k-norm, $r(p)$ is the radius of patch p. In particular, $E_S(p,q) = 0$, if $\underline{p} = \underline{q}$. These two factors combined, we have patch compatibility measures E for patch grouping:

$$E = E_C^T \cdot E_S \cdot E_C. \quad (18)$$

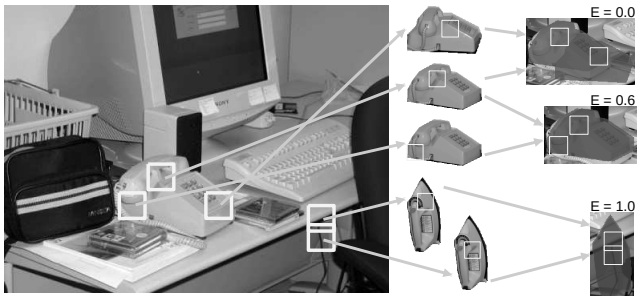


Figure 6. Patches have high affinity if their corresponding object silhouettes overlap well.

$S(p',p)$ projects an expected local segmentation of pixels. We denote it by matrix H_S, each column of which has N pixels, taking the corresponding values of $S(p',p)$ within a window 25% larger than the patch itself: $+1$ for object pixels and -1 for background pixels. Taking an average of H_S based on the affinity with neighbouring pixels and patches, we refine this initial estimation by (Fig. 7):

$$H = A\cdot H_S\cdot E. \quad (19)$$

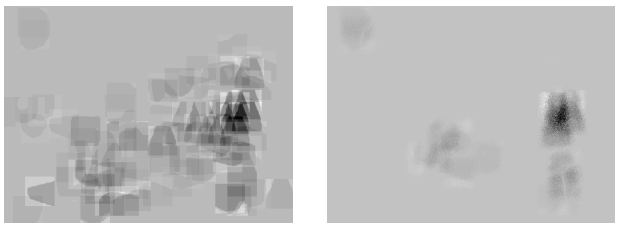
Finally, since each detected patch *could* have both foreground and background pixels, we assign two nodes to it. The resulting foreground and background patch nodes take pixels of positive and negative associations respectively, have no affinity between them and take the same affinity patterns as their parents:

$$B = \gamma\begin{bmatrix}E & \\ & E\end{bmatrix}, \quad \gamma = \frac{1}{2}\cdot\frac{1^TA1}{1^TE1} \quad (20)$$
$$C = \begin{bmatrix}H\odot(H>0), & -H\odot(H<0)\end{bmatrix}, \quad (21)$$

where $\odot$ is element-wise product, γ is to balance the pixel and patch graphs so that their total degrees match.

We have four parameters for the Gaussian functions used to evaluate these affinity measures. They are fixed for all test images: $\sigma_e = 0.02$, $\sigma_p = 0.33$, $\sigma_s = 0.08$, $\sigma_d = 0.17$.

Once we get the optimal eigenvector, we compare 10 thresholds uniformly distributed within its range and choose the discrete segmentation that yields the best criteria $\bar{\varepsilon}$.



H_S H

Figure 7. Signed pixel-patch associations. H_S is a high-level projection to image g based on object models. After diffusion with both pixel and patch affinity, H is refined and propagated. Due to the signed representation, the final association is canceled out if a boundary is expected at a uniform region. Each pixel has different associations to different patches. Here we show the summation of its weights to all patches. Dark for foreground, white for background.

4. Results

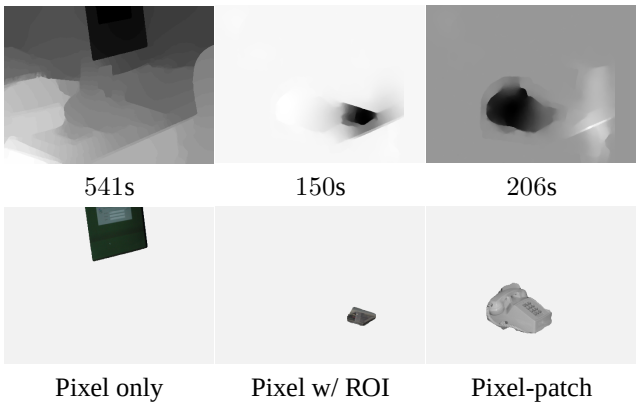


Figure 8. Comparison of pixel grouping, focused pixel grouping and pixel-patch grouping. Row #1: optimal eigenvectors. Row #2: segmentations. The MATLAB running times are given with 1GHz CPU and 1GB memory.

In Fig. 8, we compare grouping in three conditions: $\#1$: low-level pixel grouping only, where $\varepsilon(\Gamma^2_{\mathbb{V}}; A)$ is maximized; $\#2$: we use object detection to narrow down a region of interest (ROI) which contains several candidate objects, and then optimize $\varepsilon(\Gamma^2_{\mathbb{V}}; A)$ for these pixels; $\#3$: our joint pixel-patch grouping. Low-level grouping alone is attracted to a large uniform region that is irrelavent to our objects of interest. With focus of attention, it is still easily distracted by falsely detected object parts, with strong edges separating from the background. Only with the guidance of patch grouping, the object of interest, despite very weak contrast at its boundaries, pops out from the rest of the clutter. In addition, because the solution space is reduced, the computational time is in fact reduced.

In Fig. 9, the first 4 results show that occlusion, imprecision of part locations, orientation and scales are tolerated in the joint segmentation, with boundaries cleanly determined by the data. The last two results show that patches themselves are not sufficient to describe an object. When the detected patches are not correct so that boundaries are obscured, then the object could be merged into background as a whole. When patches do not cover the object, the joint grouping could damage or even miss the object.

5. Conclusions

We have developed a joint optimization model to integrate detected object parts and edges to produce object-specific figure-ground segregation. Our results show that it does not hallucinate object boundaries like most top-down object segmentation, nor does it get lost in regions of uninterested features as most low-level image segmentation. Imprecision of patch detection and poor contrast of edges are tolerated to certain degree.

However, since we lack a good representation of objects for segmentation, especially for the global shape information rather than just the patches provided, our integration model is not fully demonstrated. In particular, we would like to include another process - contour grouping. With the guidance of object models, we might eliminate the major problem of low-level contour grouping: random continuation of edgels [8]. How to get good estimation of grouping correspondence also warrants further research.

On the other hand, our formulations can also be considered an integration framework for node grouping and hyper-edge grouping. Instead of viewing patches as independent nodes, we can regard them as hyper-edges defined on basic elements - pixel nodes. The interaction matrix describes the incidence relationships. This provides a way to include high-order relationships into one grouping framework. It has already been noted that pairwise relationships are not enough in describing grouping constraints, for example we need to describe cues or hypotheses that are only valid depending on other cues. Our work could potentially provide such a representation.

Acknowledgments

We thank Shyjan Mahamud for many exciting discussions and generously providing all part detection results. We also thank CMU Hebert Lab for material support. This research is sponsored by DARPA HumanID ONR N00014-00-1-0915 and NSF IRI-9817496.

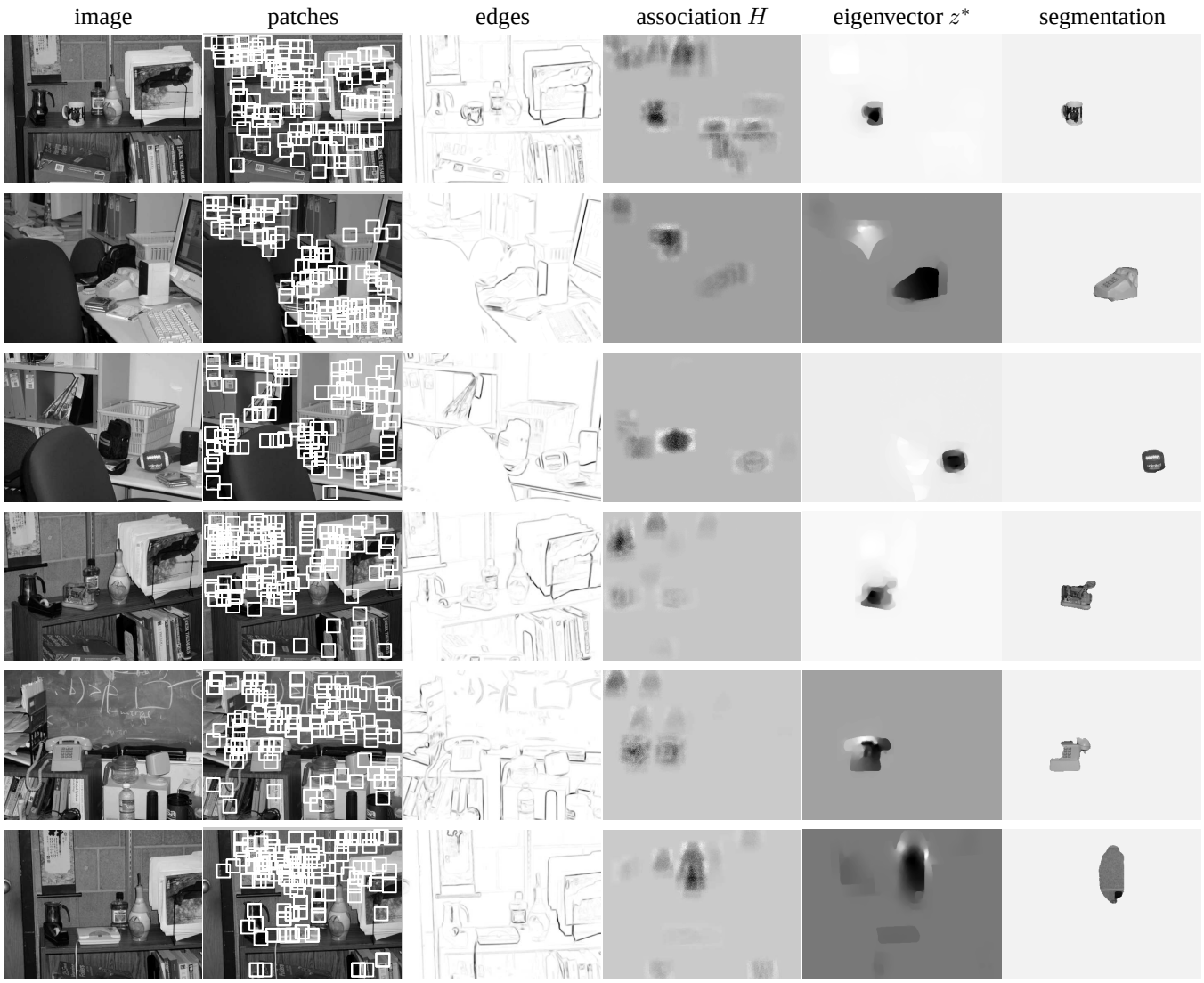


Figure 9. Results for six images.

References

[1] 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.

[2] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *European Conference on Computer Vision*, 2002.

[3] S. Mahamud. *Discriminative distance measures for object detection*. PhD thesis, Carnegie Mellon University, 2002.

[4] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 2001.

[5] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108, 1996.

[6] M. A. Peterson. Object recognition processes can and do operate before figure-ground organization. *Current Directions in Psychological Science*, 3:105–11, 1994.

[7] J. Shi and J. Malik. Normalized cuts and image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 731–7, June 1997.

[8] L. R. Williams and D. W. Jacobs. Stochastic completion fields: A neural model of illusory contour shape and salience. *Neural Computation*, 9(4):837–58, 1997.

[9] C. Xu, D. L. Pham, and J. L. Prince. *Medical image segmentation using deformable models*, pages 129–74. SPIE, 2000.

[10] S. X. Yu and J. Shi. Grouping with bias. 2001.

[11] A. L. Yuille, D. S. Cohen, and P. W. Hallinan. Feature extraction from faces using deformable templates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 104–9, 1989.