Object Segmentation by Graph Partitioning*

Stella X. Yu†‡, Ralph Gross† and Jianbo Shi†
Robotics Institute†
Carnegie Mellon University
Center for the Neural Basis of Cognition†
5000 Forbes Ave, Pittsburgh, PA 15213-3890
{stella.yu, rgross, jshi}@cs.cmu.edu

November 5, 2002

Abstract

Segmentation and recognition have long been treated as two separated processes. We propose a mechanism based on spectral graph partitioning that readily couple the two processes into one. A part-based object recognition system detects parts, supplies their partial segmentations and an evaluation of how well possible labellings on these parts go together based on the statistics of object spatial configurations. We integrate the top-level part grouping into the low-level pixel grouping based on feature similarity to segregate objects of interest from the background. We demonstrate that object part consistency and image pixel coherence can work together to eliminate local false positives and labelling ambiguities at high level, while at the same time overcome occlusion and weak contours at low level.

1 Introduction

A good image segmentation must single out meaningful structures such as objects that we recognize from a cluttered scene. Most current segmentation techniques take a bottom-up approach, where image properties such as feature (brightness, texture, motion etc) similarity, boundary smoothness and continuity are adopted to detect perceptually coherent units. Segmentation can also be approached top-down from object models, where object templates are projected onto an image and matching errors are used to determine the existence of the object [2]. Unfortunately, neither approach alone produces satisfactory results.

Without utilizing any object knowledge we have about the scene, the segmentation process can get lost in poor data conditions: weak edges, shadows, occlusions and noise. Object boundaries missed in segmentation could hardly be recovered in subsequent object recognition. Gestaltlists have long recognized this, but circumventing the issue by adding in a grouping factor called familiarity [9].

* to appear to NIPS 02
Without subject to perceptual constraints imposed by low level grouping, an object detection process can produce many false positives in a cluttered scene [5]. A typical ROC curve [5] indicates that the number of false positives increases rapidly in order to ensure a reasonable detection rate. One might argue that this problem can be avoided with more negative training examples. However, negative examples are hard to come by, and even if they were available, the complexity of classifiers would increase tremendously. Most falsely detected structures, on the other hand, are not perceptually salient, which suggest that they can be effectively pruned away by perceptual organization.

Segmentation and object recognition, when performed sequentially [7], often face such a chicken-and-egg dilemma: they rely on each other’s output, thus neither can be perfect without knowing the other. Here, we propose a segmentation mechanism which is intertwined with the object recognition process, as illustrated in Fig. 1.

Our object segmentation has three tightly coupled processes. 1) Top-level: part-based object recognition process. It learns classifiers from training images to detect parts along with the segmentation patterns and their relative spatial configurations. For example, recent work on object segmentation [2] uses the association patterns between image patches and segmentation patterns directly. A few approaches based on pattern classification have been developed for part detection [14, 5]. This is not the focus of our paper. 2) Bottom-level: pixel-based segmentation process. It tries to find perceptually coherent groups using pairwise local feature similarity subject to top-down model feedback. 3) Interactions: linking object recognition with segmentation through the instantiation of part ownerships in pixels. When a part is detected, it back projects foreground-background constraints onto the image data. When a pixel group is formed, it validates parts of the object model. With such representation, we create a parallel object recognition and image segmentation process, the goal of which is to yield self-coherent and mutually consistent high-level part configuration and low-level segmentation results.

We formulate our object segmentation task in a graph partitioning framework. We represent low-level grouping cues using a relational graph with each pixel as a node, high-level grouping cues using a graph with each part labelling as a node. Edge weights in the bottom graph encode the affinity of two pixels based on intervening contours [6], and those in the top graph encode the labelling consistency based on prior knowledge of the co-occurrence of object parts. The two graphs have edges connecting part nodes with their supporting pixel nodes. We seek the optimal graph cut in this joint graph, which separates out the desired part- and pixel- nodes from the rest. We build upon the computational mechanism of spectral graph partitioning methods [12], and achieve object part selection using the subspace constraint method proposed in [15]. We show that our formulation leads to a constrained eigenvalue problem, whose global-optimal solutions can be obtained efficiently.

The rest of the paper is organized as follows. Section 2 develops our formulation of object segmentation. Section 3 shows our experimental results. Section 4 concludes the paper with a discussion.
Figure 1: Model of object segmentation. Given an image, we detect edges using a set of filter banks as well as body parts based on a set of learned classifiers. The edge responses provide low-level grouping cues, and a relational graph can be constructed with one node for each pixel. The part responses provide high-level grouping cues, and a consistency graph can be constructed with one node for each part hypothesis. Both edges and parts contain ambiguity in local segmentation and labelling. Parts and pixels interact by mutual ownerships based on object knowledge and data coherence. A global partitioning on the coupled graphs fulfills model selection at high level and outputs object segmentation at the low level.
2 Segmentation model

Consider an image shown in Fig. 2a. We are interested in detecting a human-like configuration in the center (Fig. 2b). Our object recognition system detects a set of parts by classifying image patches in a local window (Fig. 2c). For each detected part, there is also a segmentation pattern associated with each hypothesis (head, arm, leg etc) that this part can assume (Fig. 2d). The object recognition system has also learned and supplied the evaluation on the spatial configuration of these parts forming the object of interest. The remaining problem is how to integrate such partial and imprecise object knowledge into a grouping engine.

![Figure 2: Task of segmentation with part detection. Given 120 × 120 image in a), we want to detect the human-like configuration in b). Assume that some object recognition system has detected 11 part candidates in c), each window showing the local data support in obtaining the classification. These parts are labelled for head(1), left-upper-arm(2, 9), left-lower-arm(3, 10), left-leg (11), left-upper-leg(3, 4), left-lower-leg(5), right-arm(6), right-leg(7, 8). Each labelling provides a partial local segmentation pattern in d), with light pixels for figure, dark for ground, and gray for no commitment. The goal of segmentation is to find the best part-pixel combinations that conform to the object knowledge and data coherence.](image)

2.1 Model selection and top-down feedback

**Model selection.** False positives and labelling ambiguities are inevitable in part detection. Grouping at the object recognition level is a model selection problem. We need to encode exclusion constraints: parts at different locations compete for the same object label, different labels at the same location compete for one part realization.

**Top-down feedback.** Object labelling of a part induces a hypothetic local segmentation on the image. Such top-down feedback can correct data by either breaking local coherence, e.g. illusory contours (Fig. 2c Part 1 and 6), or promoting local coherence, serving to ignore occluding edges (Fig. 2c Part 1). Parts of good configurations also selectively bridge their associated pixels into one group. Such long-range binding is essential for popping out the object of interest among many possible equally good low-level segmentations.

A non-essential but very helpful piece of information from object part detection is focus of attention (FOA)[9], where figure-ground segregation can be restricted to a region enclosing all the detected parts (Fig. 2e). This helps binding irrelevant background elements into one group, which in turn facilitates figural popout [15].
2.2 Representations in relational graphs

We build coupled relational graphs for the object segmentation model (Fig.1). Formally, we denote a relational graph by \( G = (V, E, W) \). Let \( N \) be the number of pixels in the image, \( M \) the number of part nodes in the model. The complete node set \( V = \{1, \cdots, N, N+1, \cdots, N+M\} \). The weight matrix for pairwise edge set \( E \) is:

\[
W(A, B, C) = \begin{bmatrix}
A_{N \times N} & C_{N \times M}^T \\
C_{M \times N} & B_{M \times M}
\end{bmatrix},
\]  

(1)

where \( A \) is the pixel-pixel affinity matrix, \( B \) is the part-part affinity matrix, and \( C \) is the part-pixel affinity matrix. All these weights are nonnegative.

Whereas pixel nodes represent pixels, part nodes denote the instantiation of object labelling on a set of pixels. Since we only have labelling on detected parts, and each labelling defines a set of object and non-object pixels, we create a pair of part nodes, called figure and ground nodes, for each part labelling. If \( P \) denotes the total number of part labelling, then \( M = 2P \). We assume that the figure nodes are numbered as \( N+1 : N+P \), corresponding ground nodes as \( N+P+1 : N+M \).

Object segmentation corresponds to a vertex bipartitioning problem, where \( V = V_1 \cup V_2 \) and \( V_1 \cap V_2 = \emptyset \). We assume \( V_1 \) contains a set of pixel and part nodes that correspond to the object, and \( V_2 \) is the rest background pixels and part labellings that correspond to false positives and alternative labellings. Let \( X_1 \) be an \((N+M) \times 1\) vector, with \( X_1(k) = 1 \) if vertex \( k \in V_1 \) and 0 otherwise. It is convenient to introduce the indicator for \( V_2 \), where \( X_2 = 1 - X_1 \) and 1 is the vector of ones.

2.3 Part-pixel incidence matrix \( H \)

If two parts, e.g. labelled as head and left arm, are located in a statistically feasible relative positions, then the affinity between their object pixels shall be enhanced, affinity between their non-object pixels shall be enhanced, while the separation between these object pixels and non-object pixels shall be increased.

We can show that this intuition of turning high-level part affinity into low-level pixel affinity is characterized by \( H^T E H \), where \( H_{P \times N} \) is the part-pixel incidence matrix, and \( E_{P \times P} \) is part affinity matrix. \( H_{pi} = 1 \) if pixel \( i \) lies on the object side, and \( H_{pi} = -1 \) if \( i \) on the background side, as shown in Fig. 2d. Weights in \( E \) measure pairwise labelling compatibility based on learned statistical dependencies between object parts. Note that figure nodes and ground nodes of the same part labelling take the same weights with respect to other parts, while there are no connections between them, i.e. \( B = \begin{bmatrix} E & 0 \\ 0 & E \end{bmatrix} \). Given \( H \) and \( E \) by an object detection system, we can encode top-down bias into \( A \) and \( C \).

2.4 Encoding top-down bias

Let \( \bar{A} \) be the affinity matrix computed from the image alone. \( H^T E H \) represents what the pixel-to-pixel interaction should be based on our high-level models, while \( \bar{A} \) represents what it is. To consolidate these two views, we simply take the weighted average.
Let $\alpha$ be a weighting factor, $f$ be a rectifying function: $f(x) = x$ for $x \geq 0$ and 0 otherwise. Then
\[
A_{ij} = f(\tilde{A}_{ij} + \alpha(H^T EH)_{ij}), \ i, j = 1 : N.
\]
(2)

Matrix $C$ describes the coupling between higher-level object parts and lower-level pixel segmentation. We define it by propagating the incidence relationships to neighbouring pixels using data affinity $A$. For part node $k$ and pixel node $i$, let $\gamma(k)$ denote the part labelling number corresponding to node $k$. We have:
\[
C_{ki} = \max_{j \in \{j : H\gamma(k)_j \cdot H\gamma(k)_i > 0\}} H\gamma(k)_j A_{ij}, \ k = N + 1 : N + M, i = 1 : N.
\]
(3)

2.5 Encoding exclusion for model selection

In the formulation presented thus far, nothing prevents two parts from claiming the same pixels, or pixels supporting several copies of the same part. To encode mutual exclusion on ambiguous object labels and false labelling, we need to understand how the higher-level object recognition process and lower-level segmentation is coupled.

The disambiguation occurs in two ways. First, we enforce one winner among part nodes in competition, i.e., only one of the part labelling can be validated to the object group and the rest go to the background group. This is done by constraining the indicating variables. Formally, let $U$ be a superset of nodes to be grouped and $S$ be that of nodes to be separated. We have $X_1(i) = X_1(j), \forall (i,j) \in U_m, \ m = 1 : |U|$ and $\sum_{k \in S_m} X_1(k) = 1, \ m = 1 : |S|$, where $| \cdot |$ is the cardinality of a set. Second, the coupling matrix $C$ propagates lower-level image segmentation onto the object part partitioning. Only parts whose corresponding pixels form a salient group can be grouped together in the graph partition. Together, these two processes tightly couple the top-down and bottom-up information, in sorting through the ambiguities in local pixel-pixel, pixel-part, part-part associations.

2.6 Segmentation as an optimization problem

To maintain the relative independence between the bottom- and top-level grouping, we decompose $W$ into two terms, with factor $\beta$ weighting the importance of top-down bias,
\[
W = (1 - \beta)W_1 + \beta W_2, \ W_1 = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, \ W_2 = \begin{bmatrix} 0 & C^T \\ C & B \end{bmatrix}.
\]
(4)

We extend the normalized cuts criteria to multiple weight matrices by optimizing a linearly weighted sum of normalized cuts on each graph [16]. Put together, we formulate a constrained optimization problem:
\[
\epsilon(X_1) = \sum_{t=1}^2 \sum_{i=1}^2 \frac{X_t^T (D_t - W_t) X_t}{X_t^T D_t X_t} \frac{X_t^T D_t X_t}{X_t^T D X_t} ,
\]
\[
\text{s.t.} \quad X_1(i) = X_1(j), \forall (i,j) \in U_m, \ m = 1 : |U|,
\]
\[
\sum_{k \in S_m} X_1(k) = 1, \ m = 1 : |S|.
\]
where $D_l$ is the degree matrix of $W_l$ and $D = D_1 + D_2$. Let $x = X_1 - \frac{X_1^T D X_1}{X_1^T D X_1}$. By relaxing the constraints into the form of $V^T x = 0$ [15], we can show [16, 15] that the minimizer of Eqn. (5) is the nontrivial leading eigenvector $x^*$ in the continuous domain:

$$QD^{-1}Wx^* = \lambda x^*, \quad Q = I - D^{-1}V(V^T D^{-1}V)^{-1}V^T.$$  

To summarize, given $\bar{A}, E$ and $H$, weight factors $\alpha$ and $\beta$, partial grouping constraints $V$ from the supersets $U$ and $S$, we do the following to find the optimal segmentation:

Step 1: Compute $A$: $A_{ij} = f(\bar{A}_{ij} + \alpha(H^T E H_{ij})$.  
Step 2: Compute $C$: $C_{ki} = \max_{j \in \{j: H_{\gamma(k)}H_{\gamma(k)} > 0\}} H_{\gamma(k)}A_{ij}$.  
Step 3: Form $B = \begin{bmatrix} E & 0 \\ 0 & E \end{bmatrix}$ and $W = \begin{bmatrix} (1 - \beta)A & \beta C^T \\ \beta C & \beta B \end{bmatrix}$.  
Step 4: Compute degree matrix $D$, $D_{ii} = \sum_j W_{ij}$.  
Step 5: Solve the nontrivial leading eigenvector $x^*$ of $Q^T D^{-1}Wx^* = \lambda x^*$ using $D$, $W$, $V$.  
Step 6: Threshold $x^*$ to get a discrete segmentation.

We avoid computing $H^T E H$ and $Q$ directly by taking advantage of their rank deficiencies. It can be shown that the increase in computational complexity is negligible given $M \ll N$.

3 Results

We convolve an image with oriented filters to extract edge magnitudes and phases. Pixel affinity $\bar{A}$ is computed using intervening contours [6], which is simply a Gaussian function of the maximum magnitude of edges crossing the line connecting two pixels. We obtain the segmentation by thresholding the nontrivial leading eigenvector with its mean value.
In Fig. 3, we run comparisons on the synthetic image used in Fig. 2. Image segmentation alone gets lost in a cluttered scene. With focus of attention, only large uniform groups can pop out, which can be unrelated to the object of interest or lose other parts. Without feedback correction from object models, unwanted edges (caused by occlusion) and weak edges (illusory contours) can completely change the grouping by local intrusion or leakage into the background. Only with the top-down bias of part grouping as well as its built-in model selection can the regions in an image best conforming to the object configuration be segmented out from the many possibilities of part combinations and the rest background.

We apply our method to human body detection in a single image. We manually labelled five body parts (both arms, both legs and the head) of a person walking on a treadmill in all 32 images of a complete gait cycle. Using the magnitude thresholded edge orientations in the handlabeled boxes as features, we train Fisher linear classifiers.
for each body part. In order to account for the appearance changes of the limbs through the gait cycle we use two separate models for each arm and each leg, bringing the total number of models to 9. Each individual classifier is trained to discriminate between the body part and a random image patch. We iteratively retrain the classifiers using false positives until optimal performance is reached over the training set. In addition we train linear color-based classifiers for each body part to perform figure-ground discrimination on the pixel level. Alternatively a general model of human appearance using e.g. filter responses as in [13] could be used.

In Fig. 4, we show the results on a test image. Body part detectors return many false positives, which can be reduced when the statistics of their relative spatial configurations are used. Though the part-pixel incidence matrix $H$, derived from the color classifier, is neither precise nor complete, and the edges are weak at many object boundaries, the two subprocesses complement each other in our part-pixel grouping system and output a reasonably good object segmentation.

![Image of human body segmentation](image)

**Figure 4:** Human body segmentation. Image size: 261 × 183. $\bar{A}$ is computed for pixel pairs within a radius of 8. First row: part detection using classifiers. Object labels and the number of detections are given. A total of 27 parts are detected. Last row: magnitudes of edge responses, Ncut and segmentation results by old low-level grouping and the new part-pixel grouping method.

Finally, we see from both examples that it is in fact faster to compute the solutions using part-pixel grouping. The main reason is that the solution space is greatly reduced and the optimal solution dominates alternatives. This provides an explanation for the speed of biological vision systems given their slow neurochemical substrates.
4 Acknowledgment

This work is supported by DARPA HumanID: ONR N00014-00-1-0915, NSF CareMedia: IIS-0105219, ARDA V ACE: MDA908-00-C-0037.

References


