Concurrent Object Recognition and Segmentation by Graph Partitioning

Stella X. Yu \textsuperscript{1,2} Ralph Gross \textsuperscript{1} Jianbo Shi \textsuperscript{1}

Robotics Institute \textsuperscript{1}
Carnegie Mellon University
Center for the Neural Basis of Cognition \textsuperscript{2}
Why segmentation needs recognition?

Why recognition needs segmentation?
Image segmentation is often object-blind

1. Do not know which regions make up an object.
2. Easily miss object boundaries due to lighting and clutter.
Object detection is often overwhelmed

1. Tradeoff between false positives and detection rate.
2. Constraints in reducing false detection: increase in classifier complexity and training size.

(Schneiderman, 02): vasc.ri.cmu.edu/demos/faceindex
Characteristics of false positives

1. Lack of high-level part label compatibility.
2. Lack of low-level image feature saliency.
Overview of our object segmentation

Arm-1

Head

Leg-1

Arm-2

Leg-2
Representation

Graph: \( G = (V, E, W) = (\text{nodes}, \text{edges}, \text{weights}) \)

Node set: \( V = V_{\text{pixels}} \cup V_{\text{patches}} \)

Edge set: \( E = E_{\text{pixel-pixel}} \cup E_{\text{patch-patch}} \cup E_{\text{pixel-patch}} \)

Weights: \( W = \begin{bmatrix} A & C^T \\ C & B \end{bmatrix} \)

- \( A \): pixel-pixel similarity matrix
- \( B \): patch-patch compatibility matrix
- \( C \): pixel-patch association matrix
patches

image pixel-patch association edges segmentation

evaluation integration

object

YU, GROSS & SHI: NIPS'02 VS18 – p.8/17
Computing pixel-pixel similarity $A$

\[
A(i, j) = \exp \left( -\frac{1}{2\sigma^2} \cdot \left[ \frac{\max_{t \in (0,1)} OE(i + t \cdot j)}{\max_k OE(k)} \right]^2 \right), \quad k = \text{location of } k.
\]

\[
A(1, 3) \approx 1
\]

\[
A(1, 2) \approx 0
\]
Computing patch compatibility and competition

\[ B(1, 2) \approx 1 \]
\[ B(10, 5) \approx 0 \]

7 and 8 cannot both be parts of the object

\[ B(p, q) \] is small if \( p, q \) form rare configurations for part labels \( \hat{p}, \hat{q} \):

\[ B(p, q) = \exp \left( -\frac{1}{2} (p - q - \mu_{\hat{p} \hat{q}})^T \Sigma_{\hat{p} \hat{q}}^{-1} (p - q - \mu_{\hat{p} \hat{q}}) \right), \quad p = \text{location of } p. \]
Computing pixel-patch association $C$

\[ C(i, p) = \begin{cases} 
1, & \text{if } i \text{ is an object pixel of patch } p \\
0, & \text{otherwise}
\end{cases} \]
Find low-cost cuts subject to patch competition
Encoding graph cuts

Segmentation: \( V = V_1 \cup V_2 = \text{object nodes} \cup \text{the rest}. \)

Indicators: \( X = [X_1, X_2] = [\text{is-object}, \text{is-nonobject}]. \)

Degree: \( D = \text{diag}(W \cdot 1). \)

Cuts criterion: \( \max \text{NCuts}(X) = \frac{X_1^T W X_1}{X_1^T D X_1} + \frac{X_2^T W X_2}{X_2^T D X_2}. \)

(Shi and Malik, 97)
Competing nodes: pairs of patches of the same label.

\[ S = N + \{\{2, 9\}, \{3, 10\}, \{4, 11\}, \{7, 8\}, \{1, 12\}\} \]

e.g. \[ X_1(N + 2) + X_1(N + 9) = 1. \]

\[ N = \text{total number of pixels} \]

Exclusion condition: one winner only

\[ \sum_{k \in S_m} X_1(k) = 1, \ m = 1 : |S|. \]

\[ S_m = \text{a set of nodes in competition}. \]
Computational solution

Change of variable:

\[ x = X_1 - \frac{X_1^T D X_1}{1^T D 1}, \]

we have constrained eigenvalue problem:

\[ x^* = \arg \max x^T W x \frac{x^T D x}{x^T D x}, \quad \text{subject to } L^T x = 0. \]

Eigensolution in the relaxed continuous domain:

\[ Q D^{-1} W x^* = \lambda x^*, \]

\[ Q = I - D^{-1} L (L^T D^{-1} L)^{-1} L^T. \]
Results I

segmentation alone

segmentation-recognition

44 seconds

17 seconds
Results II

segmentation alone: 68s

segmentation-recognition: 58s