Perceptual Organization
(IV)

Introduction to Computational and Biological Vision

CS 202-1-5261

Computer Science Department, BGU

Ohad Ben-Shahar
**Segmentation**

Segmentation as partitioning

Given: \( I \) - a set of image pixels

\( H \) – a region homogeneity predicate (based on visual properties of interest)

A segmentation of the image is a (meaningful) partition of \( I \) into \( m \) regions \( R_i \) such that

\[
R_i \cap R_j = \emptyset \quad \forall i \neq j \\
\bigcup_{i=1}^{m} R_i = I \\
H(R_i) = true \quad \forall i \\
H(R_i \cup R_j) = false \quad \forall i \neq j \quad \text{adjacent}
\]
Segmentation

Is segmentation easy?
Segmentation

Segmentation is conceptually ill defined
Segmentation

Segmentation is conceptually ill defined

25 segments
Segmentation

Segmentation is conceptually ill defined
Segmentation

Segmentation is difficult!!
Segmentation

Segmentation is difficult!!
Segmentation

Segmentation is difficult!!
Segmentation

Segmentation is dual to boundary/edge detection

Segmentation (Regions)
Make explicit intra-region coherence

Edges (Boundaries)
Make explicit inter-region differences
Segmentation

Is segmentation dual to boundary/edge detection???
Segmentation Approaches

Segmentation via thresholding

\[ R(x, y) = \begin{cases} 
0 & I(x, y) < t \\
1 & t \leq I(x, y) 
\end{cases} \]
Segmentation Approaches

Segmentation via thresholding

\[ R(x, y) = \begin{cases} 
1 & I_{\text{min}} \leq I(x, y) < t_1 \\
2 & t_1 \leq I(x, y) < t_2 \\
3 & t_2 \leq I(x, y) < t_3 \\
\vdots & \vdots \\
m & t_{m-1} \leq I(x, y) < I_{\text{max}} 
\end{cases} \]
Segmentation Approaches

Global thresholding

$I(x, y)$

$\tau_i$
Segmentation Approaches

Automatic global threshold selection

$h(I)$

$t$

$t_1, t_{m-1}$
Segmentation Approaches

Global thresholding? Watch out

$I(x, y)$


**Segmentation Approaches**

In real images

- Noise
- gradual changes in illumination

Global thresholding is likely to fail

Possible improvement:

Local adaptive thresholding
Segmentation Approaches

Local adaptive thresholding

\[ W(x,y) = \text{window of pixel } (x,y) \]

\[ R(x, y) = \begin{cases} 0 & I(x, y) < t_{W(x,y)} \\ 1 & t_{W(x,y)} \leq I(x, y) \end{cases} \]
Segmentation Approaches

Adaptive thresholding
Segmentation Approaches

Adaptive thresholding
Segmentation Approaches

Representing segmentations

Region Adjacency Graphs (RAGs)  Region (Picture) Trees
Segmentation Approaches

Split and Merge

Split

Merge
Segmentation Approaches

Region merging

1. Form initial segmentation
2. Compute RAG
3. Repeat
   • Pick an edge $e$ that connects two regions $R_i$ and $R_j$ in the RAG
   • If $H(R_i \cap R_j) = true$ then merge the two regions and update the RAG
Until no more regions can be merged
Segmentation Approaches

Merging statistically similar regions

**Assumption**: region have constant feature value corrupted by statistically independent, additive, normally distributed noise.

**Hypothesis H₀**: Regions should be merged. Their feature values are all drawn from the same single normal distribution with parameters \((\mu_0, \sigma_0)\)

**Hypothesis H₁**: Regions should not be merged. Their feature values are drawn from two different normal distributions with parameters \((\mu_1, \sigma_1)\) and \((\mu_2, \sigma_2)\)

Which hypothesis should be selected?
Segmentation Approaches

Merging statistically similar regions

Probability of any given value

\[ P(v_i) = \frac{1}{\sqrt{2\pi \sigma}} e^{-\frac{(v_i - \mu)^2}{2\sigma^2}} \]

Distribution mean

\[ \hat{\mu} = \frac{1}{n} \sum_{i} v_i \]

Distribution variance

\[ \hat{\sigma}^2 = \frac{1}{n} \sum_{i} (v_i - \hat{\mu})^2 \]

\[ P(v_1, v_2, \ldots, v_{m_1+m_2} \mid H_0) = \prod_{i=1}^{m_1+m_2} P(v_i \mid H_0) = \prod_{i=1}^{m_1+m_2} \frac{1}{\sqrt{2\pi \sigma_0}} e^{-\frac{(v_i - \mu_0)^2}{2\sigma_0^2}} \]

\[ = \frac{1}{\left(\sqrt{2\pi \sigma_0}\right)^{m_1+m_2}} e^{-\frac{\sum_{i=1}^{m_1+m_2} (v_i - \mu_0)^2}{2\sigma_0^2}} = \frac{1}{\left(\sqrt{2\pi \sigma_0}\right)^{m_1+m_2}} e^{-\frac{m_1+m_2}{2}} \]
Segmentation Approaches

Merging statistically similar regions

Probability of any given value

\[
P(v_i) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(v_i-\mu)^2}{2\sigma^2}}
\]

Distribution mean

\[
\hat{\mu} = \frac{1}{n} \sum_{i} v_i
\]

Distribution variance

\[
\hat{\sigma}^2 = \frac{1}{n} \sum_{i} (v_i - \hat{\mu})^2
\]

\[
P(v_1, v_2, \ldots, v_{m_1+m_2} \mid H_1) = \prod_{i=1}^{m_1+m_2} P(v_i \mid H_1) = \prod_{i=1}^{m_1} \frac{1}{\sqrt{2\pi\sigma_1}} e^{-\frac{(v_i-\mu_1)^2}{2\sigma_1^2}} \prod_{i=m_1+1}^{m_2} \frac{1}{\sqrt{2\pi\sigma_2}} e^{-\frac{(v_i-\mu_2)^2}{2\sigma_2^2}}
\]

\[
= \ldots = \left(\frac{1}{\sqrt{2\pi\sigma_1}}\right)^{m_1} e^{-\frac{m_1}{2}} \left(\frac{1}{\sqrt{2\pi\sigma_2}}\right)^{m_2} e^{-\frac{m_2}{2}}
\]
Segmentation Approaches

Merging statistically similar regions

Probability of any given value

\[ P(v_i) = \frac{1}{\sqrt{2\pi \sigma}} e^{-\frac{(v_i - \mu)^2}{2\sigma^2}} \]

Distribution mean

\[ \hat{\mu} = \frac{1}{n} \sum_i v_i \]

Distribution variance

\[ \hat{\sigma}^2 = \frac{1}{n} \sum_i (v_i - \hat{\mu})^2 \]

Likelihood ratio

\[ L = \frac{P(v_1, v_2, \ldots, v_{m_1+m_2} \mid H_1)}{P(v_1, v_2, \ldots, v_{m_1+m_2} \mid H_0)} = \frac{\sigma_0^{m_1+m_2}}{\sigma_1^{m_1} \sigma_2^{m_2}} \]
Segmentation Approaches

Region splitting

1. Form initial segmentation
2. Compute RAG
3. Repeat
   - Pick a node $R_i$ in the RAG
   - If $H(R_i) = false$ then split the region and update the RAG

Until no more splits can be done
Segmentation Approaches

Split and Merge

1. Form initial segmentation
2. Compute RAG
3. Repeat
   • Pick a node $R_i$ in the RAG and examine it for splitting. Update RAG if split is exercised.
   • Pick an edge $e$ that connects two regions $R_i$ and $R_j$ in the RAG and examine it for merging. Update RAG if merge is exercised.

Until no more splits can be done
Segmentation Approaches

Segmentation via relaxation

Contextual constraints

\[ X = Y \]
\[ X \neq Z \]
\[ Y \neq Z \]

Measurements

\[ P_X(\text{Apple}) = 100\% \quad P_Y(\text{Apple}) = 50\% \quad P_Z(\text{Apple}) = 50\% \]
\[ P_X(\text{Orange}) = 0\% \quad P_Y(\text{Orange}) = 50\% \quad P_Z(\text{Orange}) = 50\% \]
Segmentation Approaches

Segmentation via relaxation

\[ B = \{b_1, \ldots, b_n\} \quad \text{set of objects to be labeled} \]

\[ \Lambda = \{1, 2, \ldots, m\} \quad \text{set of possible labels} \]

\[ p^0_i(\lambda) \quad \text{the measured confidence that } b_i \text{ should be labeled } \lambda. \]

\[ p^0_i(\lambda) \geq 0 \]

\[ \sum_{\lambda=1}^{m} p^0_i(\lambda) = 1 \quad \forall i \]

\[ r_{ij}(\lambda, \mu) \quad \leftrightarrow \quad \text{the strength of compatibility between the hypotheses “} b_i \text{ has label } \lambda \text{” and “} b_j \text{ has label } \mu \text{”} \]
Segmentation Approaches

Segmentation via relaxation
Segmentation Approaches

Segmentation via clustering
Segmentation Approaches

Segmentation via clustering

Problem formulation:

Given a set of data points $x_i$, find $K$ clusters $C_j$, with representatives $m_j$, such that the total fit measure of data points to clusters is minimized

$$D = \sum_{k=1}^{K} \sum_{x_i \in C_k} d(x_i, m_i) \rightarrow \min$$

Least square error measure:  

$$d(x_i, m_i) = \|x_i - m_i\|^2$$

Image segmentation as clustering:

- $x_i$: feature vectors associated with pixels
- $C_j$: segments
- $m_j$: representative (mean) feature vector for segment $j$
Segmentation Approaches

Iterative K-means clustering

1. Chose randomly the set of K cluster centers $m_1, m_2, \ldots, m_K$

2. Repeat
   - Allocate each data point to the cluster whose center is nearest
   - Update all $m_i$ to the center point of their cluster

Until cluster centers are unchanged
Segmentation Approaches

Iterative K-means clustering

6-clusters
Segmentation Approaches

Graph theoretic approach to segmentation
Segmentation Approaches

Graph theoretic approach to segmentation
Segmentation Approaches

Graph theoretic approach to segmentation

\[ S_{ij} = \text{similarity (weight/affinity) based on intensity, color, texture, etc…} \]
Segmentation Approaches

Graph theoretic approach to segmentation

Segmentation = Graph cuts
Segmentation Approaches

Graph theoretic approach to segmentation

Given:

- a graph representation \((V,E)\) of the image and a pairwise similarity measure

Compute:

- A partition of the graph into disjoint sets \(V_1, V_2, \ldots, V_m\) such that the total similarity is maximized within each \(V_i\) and is minimized between any two \(V_i\) and \(V_j\)

Recursive formulation:

- A partition of the graph into disjoint sets \(A\) and \(B\) and apply recursively to \(A\) and \(B\).
Segmentation Approaches

Graph theoretic approach to segmentation

\[ A \cup B = V \]

\[ A \cap B = \emptyset \]

\[ cut(A, B) = \sum_{e_1 \in A; e_2 \in B} S_{e_1, e_2} \]

Segmentation \neq \text{Minimum cuts}
Segmentation Approaches

Graph theoretic approach to segmentation

\[ \text{cut}(A, B) = \sum_{e_1 \in A; e_2 \in B} S_{e_1,e_2} \]

\[ W = [S_{ij}] \]
Segmentation Approaches

Graph theoretic approach to segmentation

\[ \text{cut}(A, B) = \sum_{e_1 \in A; e_2 \in B} S_{e_1,e_2} \]

\[ W = [S_{ij}] \]

\[ d_i = \sum_j S_{ij} \]

Degree of a node

Similarity Matrix
Segmentation Approaches

Graph theoretic approach to segmentation

\[
cut(A, B) = \sum_{e_1 \in A; e_2 \in B} S_{e_1, e_2}
\]

\[
W = [S_{ij}]
\]

\[
d_i = \sum_j S_{ij}
\]

Degree of a node

\[
vol(A) = \sum_{i \in A} d_i
\]

Volume of a set

Similarity Matrix
Segmentation Approaches

Graph theoretic approach to segmentation

Normalized cuts

\[ Ncut(A, B) = \frac{cut(A, B)}{vol(A)} + \frac{cut(B, A)}{vol(B)} \]

NP-Hard!! 😞
Segmentation Approaches

Graph theoretic approach to segmentation

\[
Ncut(A, B) = \frac{\text{cut}(A, B)}{\text{vol}(A)} + \frac{\text{cut}(B, A)}{\text{vol}(B)}
\]

\[
D = \begin{bmatrix}
d_1 & 0 & 0 & 0 \\
0 & d_2 & 0 & 0 \\
0 & 0 & \ddots & 0 \\
0 & 0 & 0 & d_n
\end{bmatrix}
\]

Laplacian Matrix \( L = D - W \)

Segmentation vector \( \vec{x} = (x_i) \) \( x_i = \begin{cases} 
1 & i \in A \\
-1 & i \notin A
\end{cases} \)
Segmentation Approaches

Graph theoretic approach to segmentation

\[ \text{Ncut}(A, B) \rightarrow \min \quad \Leftrightarrow \quad \frac{\text{cut}(A, B)}{\text{vol}(A)} + \frac{\text{cut}(B, A)}{\text{vol}(B)} \] 

\[ y = (1 + x) - b(1 - x) \]

\[ b = \frac{k}{1 - k} \]

\[ k = \frac{\sum_{x_i > 0} d_i}{\sum_i d_i} \]
**Segmentation Approaches**

Graph theoretic approach to segmentation

\[ Ncut(A, B) = \frac{cut(A, B)}{vol(A)} + \frac{cut(B, A)}{vol(B)} \]

\[ \min (D - W)y = \lambda Dy \]

\[ z = D^{\frac{1}{2}}y \quad \Leftrightarrow \quad D^{-\frac{1}{2}}LD^{-\frac{1}{2}}z = \lambda z \quad \Leftrightarrow \quad Mz = \lambda z \]

Relax (allow any y)
Segmentation Approaches

Graph theoretic approach to segmentation

Given an image for segmentation:

1. Set up a graph $G=(V,E)$ with weights measuring similarities between pixels.
2. Compute $W,D,L$
3. Solve for the eigenvectors of $D^{-\frac{1}{2}} (D-W) D^{-\frac{1}{2}}$
4. Use the eigenvector of the second smaller eigenvalue to bipartition the graph.
5. Apply recursively as needed.
Segmentation Approaches

Graph theoretic approach to segmentation
Segmentation Approaches

Graph theoretic approach to segmentation
Segmentation Approaches

Graph theoretic approach to segmentation

(a)  (b)
(c)  (d)  (e)  (f)  (g)
Segmentation Approaches

Leap to the present - Machine learning and Semantic Segmentation

• Artificial neural networks (ANNs) are systems broadly inspired by the biological neural networks that constitute animal brains.

• ANNs consist of “neurons” where each neuron receives input signals, processes them (typically with a non linear activation function) and outputs a signal that drives consecutive neurons.

• Each edge has a weight that increases (“excites”) or decreases (“inhibits”) the unit activation based on the signal in that edge.

• It can be shown that it is possible to model any logical operator (e.g. AND, OR, XOR).

\[
\begin{align*}
    n_i &= w_{i1} + w_{i2} + \ldots + w_{ik} \\
    o_i &= f(n_i)
\end{align*}
\]
Segmentation Approaches

Leap to the present - Machine learning and Semantic Segmentation

• One popular class of ANNs are feedforward consist of an input layer and an output layer. In between, there can be any number of hidden layers.

• The structure of the network changes depending on the desired task.

Working with ANNs is typically “supervised”, i.e., consists of two stages:

• **Training** - labeled data (e.g. pictures segmented by a human) is used together with a loss function to calculate optimal edge weights.

• **Classification** - the neural net is used to classify new input.
Segmentation Approaches

Leap to the present - Machine learning and Semantic Segmentation

- In computer vision, a commonly used network is a convolutional neural network (CNN), i.e., a layered network with weight sharing between different units in the hidden layer.
- The input layer’s neurons can be the brightness or color values of an image’s pixels.
- Each layer is a convolution of the previous layer.
- Kernel values are determined by edges weight and kernel size by network structure.
**Segmentation Approaches**

Leap to the present - Machine learning and Semantic Segmentation

- A diverse dataset is segmented manually (can be millions of images)
- Networks designed by experts are used. They contain a combination of convolution layers and additional “non linear” layers.
- The training process can be very computational extensive. Advancement in GPU technologies made turned this process realistic in time.
- The results are phenomenal:
Segmentation Approaches

Leap to the present - Machine learning and Semantic Segmentation