Weakly Supervised Learning in Semantic Segmentation

Sheng Zeng

September 15, 2014

Contents

Contents

Image Understanding

Image Understanding

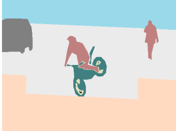
- Motorbike
- Car
- Person
- Street

Image Classification

Image Understanding

- Motorbike
- Car
- Person
- Street

Image Classification



Segmentation

Image Understanding

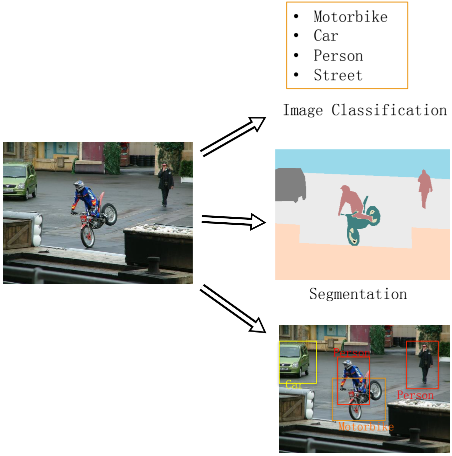
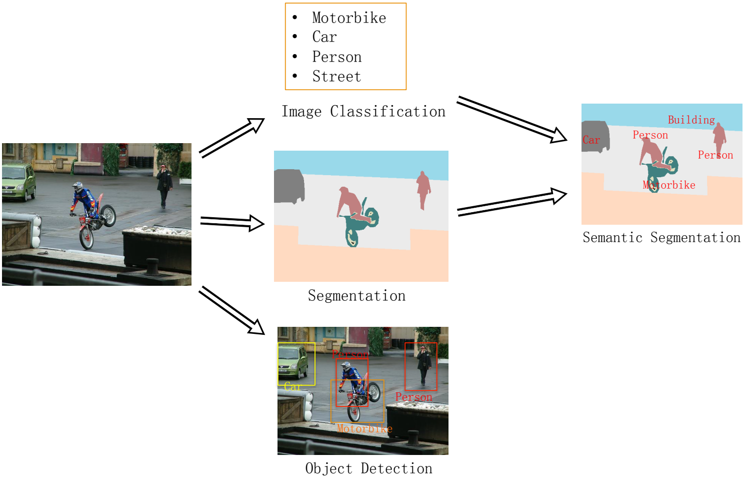


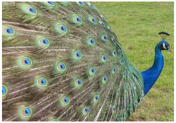
Image Understanding



Common Challenges in Image Understanding

intra-class variability

bird



Common Challenges in Image Understanding

intra-class variability

bird

deformation

cat

Common Challenges in Image Understanding

intra-class variability

deformation

viewpoint changes

bird

cat

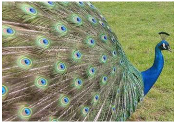
car

Common Challenges in Image Understanding

intra-class variability

bird





deformation

cat



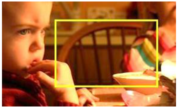
viewpoint changes

car



occlusion

chair


Contents

Motivation

- The labeled data is **limited**!

Motivation

- The labeled data is **limited**!
- Large scale image annotating is **time-consuming**.

Motivation

- The labeled data is **limited**!
- Large scale image annotating is **time-consuming**.
- Weakly labeled data can be **easily obtained** from the internet.

Two Settings of Weakly Supervised Learning

1. Only weakly labeled data. e.g. [Verbeek and Triggs, 2007]

Two Settings of Weakly Supervised Learning

1. Only weakly labeled data. e.g. [Verbeek and Triggs, 2007]
2. A few precisely annotated data + a large mount of weakly labeled data. e.g. [Hoffman et al.,]
   - Domain Adaptation (DA)

Contents

Definition

- Image segmentation is the process of partitioning a digital image into **multiple segments** (sets of pixels, also known as superpixels).

Definition

- Image segmentation is the process of partitioning a digital image into **multiple segments** (sets of pixels, also known as superpixels).
- Image segmentation is typically used to locate **objects** and **boundaries** (lines, curves, etc.) in images.

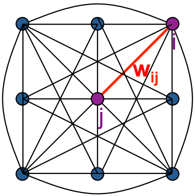
Definition

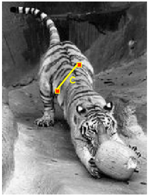
- Image segmentation is the process of partitioning a digital image into **multiple segments** (sets of pixels, also known as superpixels).
- Image segmentation is typically used to locate **objects** and **boundaries** (lines, curves, etc.) in images.
- Image segmentation is the process of **assigning a label** to every pixel in an image such that pixels with the same label share certain characteristics.

Contents

Graph-based Segmentation

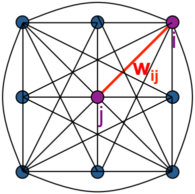
- Treating the images as graphs
  - node for evey pixel
  - link between every pair of pixels
  - similarity W_{ij} for each link

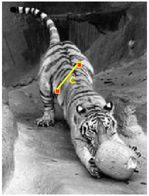




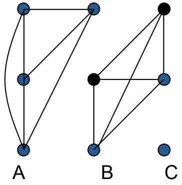
Graph-based Segmentation

- Treating the images as graphs
  - node for evey pixel
  - link between every pair of pixels
  - similarity W_{ij} for each link
- Method
  - minimun cut
  - Normalized cut
  - MRFs Graph cuts



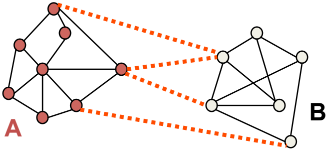


Segmentation by Graph cuts



- Break Graph into Segments
  - Delete links that cross between segments
  - Easiest to break links that have low cost (low similarity)
    - similar pixels should be in the same segments
    - dissimilar pixels should be in the different segments

Cut in Graphs



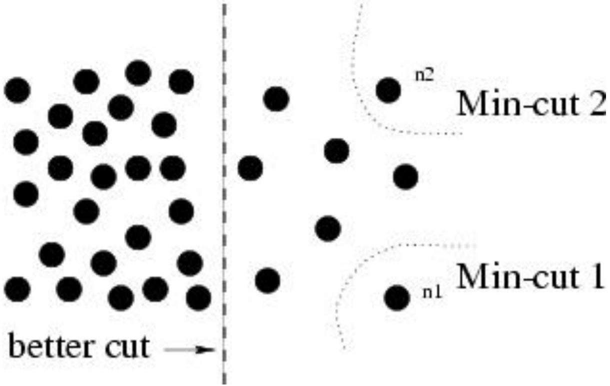
- Link Cut
  - set of links whose removal makes a graph disconnected
  - cost of a cut:

$$cut(A, B) = \sum_{p \in A, q \in B} c_{p,q} \tag{1}$$

- One idea: Find the minimum cut.
  - fast algorithms exist for doing this

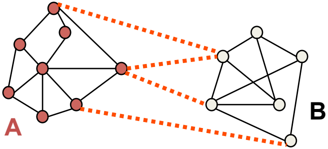
Cut in Graphs

But min cut is not always the best cut...



[Shi and Malik, 2000]

Cut in Graphs



Normalized Cut [Shi and Malik, 2000]

- a cut penalizes large segments
- fix by normalizing for size of segments

$$Ncut(A, B) = \frac{cut(A, B)}{volumn(A)} + \frac{cut(A, B)}{volumn(B)} \qquad (2)$$

- $volume(A)$ = sum of costs of all edges that touch A

Recursive Normalized Cut

1. Given an image or image sequence, set up a weighted graph: $G = (V, E)$
   - Vertex for each pixel
   - Edge weight for nearby pairs of pixels

$$\min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = \min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\mathbf{D} - \mathbf{W})\boldsymbol{y}}{\boldsymbol{y}^T\mathbf{D}\boldsymbol{y}} \tag{3}$$

[1] Details: http://www.cs.berkeley.edu/~malik/papers/SM-ncut.pdf

Recursive Normalized Cut

1. Given an image or image sequence, set up a weighted graph: $G = (V, E)$
   - Vertex for each pixel
   - Edge weight for nearby pairs of pixels

$$\min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = \min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\mathbf{D} - \mathbf{W})\boldsymbol{y}}{\boldsymbol{y}^T\mathbf{D}\boldsymbol{y}} \tag{3}$$

2. Solve for eigenvectors with the smallest eigenvalues: $(\mathbf{D} - \mathbf{W})\boldsymbol{y} = \lambda\mathbf{D}\boldsymbol{y}$

[1] Details: http://www.cs.berkeley.edu/~malik/papers/SM-ncut.pdf

Recursive Normalized Cut

1. Given an image or image sequence, set up a weighted graph: $G = (V, E)$
   - Vertex for each pixel
   - Edge weight for nearby pairs of pixels

$$\min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = \min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\mathbf{D}-\mathbf{W})\boldsymbol{y}}{\boldsymbol{y}^T\mathbf{D}\boldsymbol{y}} \quad (3)$$

2. Solve for eigenvectors with the smallest eigenvalues: $(\mathbf{D}-\mathbf{W})\boldsymbol{y} = \lambda\mathbf{D}\boldsymbol{y}$
3. Use the eigenvector with the second smallest eigenvalue to bipartition the graph
   - Note: this is an approximation

[1] Details: http://www.cs.berkeley.edu/~malik/papers/SM-ncut.pdf

Recursive Normalized Cut

1. Given an image or image sequence, set up a weighted graph: $G = (V, E)$
   - Vertex for each pixel
   - Edge weight for nearby pairs of pixels

$$\min_{\boldsymbol{x}} Ncut(\boldsymbol{x}) = \min_{\boldsymbol{y}} \frac{\boldsymbol{y}^T(\mathbf{D} - \mathbf{W})\boldsymbol{y}}{\boldsymbol{y}^T\mathbf{D}\boldsymbol{y}} \tag{3}$$

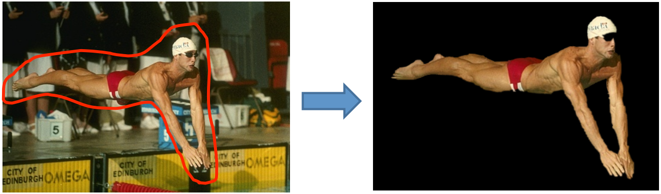
2. Solve for eigenvectors with the smallest eigenvalues: $(\mathbf{D} - \mathbf{W})\boldsymbol{y} = \lambda\mathbf{D}\boldsymbol{y}$
3. Use the eigenvector with the second smallest eigenvalue to bipartition the graph
   - Note: this is an approximation
4. Recursively repartition the segmented parts if necessary [1]

[1] Details: http://www.cs.berkeley.edu/~malik/papers/SM-ncut.pdf

Normalized Cut: Pros and Cons

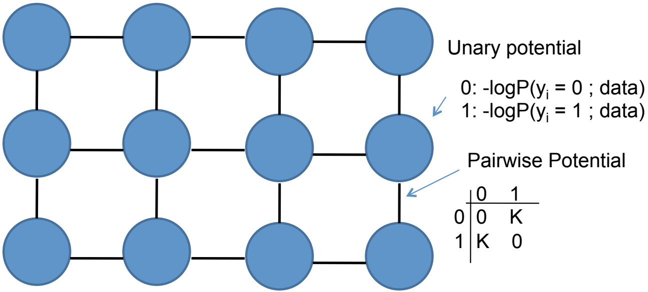
- Pros
  - Generic framework, can be used with many different features and affinity formulations
  - Provides regular segments
- Cons
  - Need to chose number of segments
  - High storage requirement and time complexity
  - Bias towards partitioning into equal segments

Graph cuts Segmentation



[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

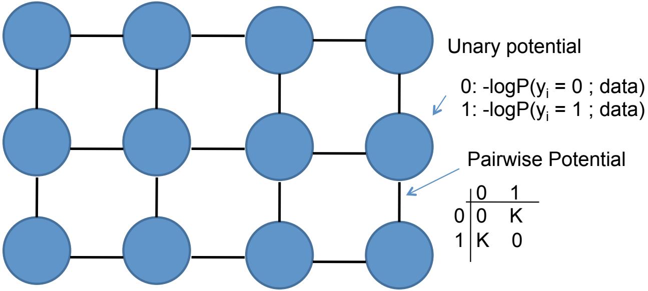
Markov Random Fields



$$Energy(\mathbf{y};\theta,data) = \sum_i \psi_1(y_i,\theta,data) + \sum_{i,j\in edges} \psi_2(y_i,y_j,\theta,data)$$

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Markov Random Fields

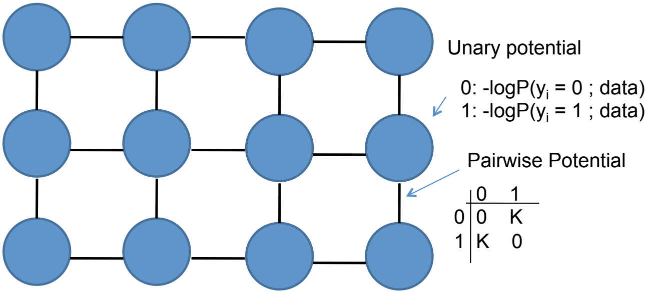


$$Energy(\mathbf{y}; \theta, data) = \sum_i \psi_1(y_i, \theta, data) + \sum_{i,j \in edges} \psi_2(y_i, y_j, \theta, data)$$

- Cost to assign a label to each pixel

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Markov Random Fields

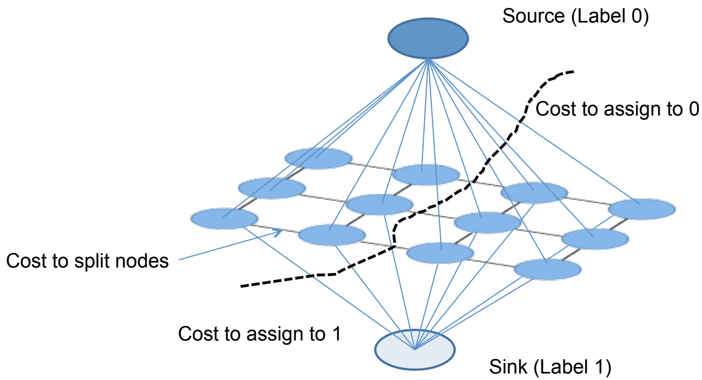


$$Energy(\mathbf{y};\theta,data) = \sum_i \psi_1(y_i,\theta,data) + \sum_{i,j\in edges} \psi_2(y_i,y_j,\theta,data)$$

- Cost to assign a label to each pixel
- Cost to assign a pair of labels to connected pixels

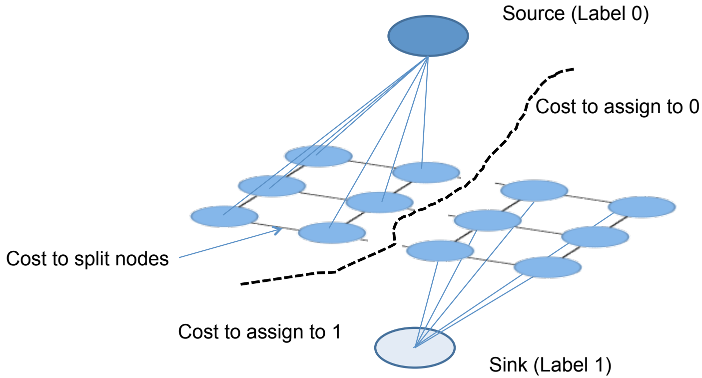
[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Solving MRFs with Graph cuts



[1]Derek Hoiem@MRFs and Graph Cuts Segmentation

Solving MRFs with Graph cuts



[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Graph cuts Segmentation

1. Define graph
   - usually 4-connected or 8-connected

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Graph cuts Segmentation

1. Define graph
   - usually 4-connected or 8-connected
2. Define unary potentials
   - Color histogram or mixture of Gaussians for background and foreground

$$unary\_potential(x) = -\log\left(\frac{P(c(x);\theta_{foreground})}{P(c(x);\theta_{background})}\right)$$

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Graph cuts Segmentation

1. Define graph
   - usually 4-connected or 8-connected
2. Define unary potentials
   - Color histogram or mixture of Gaussians for background and foreground

$$unary\_potential(x) = -\log\left(\frac{P(c(x);\theta_{foreground})}{P(c(x);\theta_{background})}\right)$$

3. Define pairwise potentials

$$pairwise\_potential(x,y) = k_1 + k_2 \exp\left\{\frac{-\|c(x)-c(y)\|}{2\sigma^2}\right\}$$

[1]Derek Hoiem@MRFs and Graph Cuts Segmentation

Graph cuts Segmentation

1. Define graph
   - usually 4-connected or 8-connected
2. Define unary potentials
   - Color histogram or mixture of Gaussians for background and foreground

$$unary\_potential(x) = -\log\left(\frac{P(c(x);\theta_{foreground})}{P(c(x);\theta_{background})}\right)$$

3. Define pairwise potentials

$$pairwise\_potential(x,y) = k_1 + k_2 \exp\left\{\frac{-\|c(x)-c(y)\|}{2\sigma^2}\right\}$$

4. Apply graph cuts [Kolmogorov and Zabin, 2004]

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Graph cuts: Pros and Cons

- Pros
  - Very fast inference
  - Can incorporate recognition or high-level priors
  - Applies to a wide range of problems (image labeling, recognition)
- Cons
  - Need unary terms (not used for generic segmentation)

[1] Derek Hoiem@MRFs and Graph Cuts Segmentation

Other Segmentation Algorithms

- Cluster-based Segmentation
  - Mean Shift
  - K-means
  - ...
- Edge-based Segmentation
  - Watershed Segmentation
    - Hierarchical segmentation from soft boundaries
  - ...

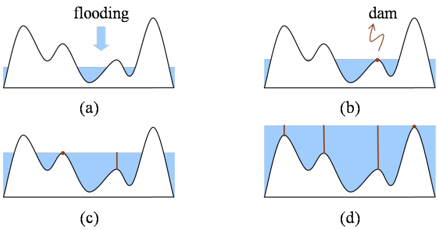
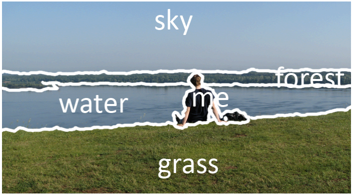


Figure: The concept of watershed

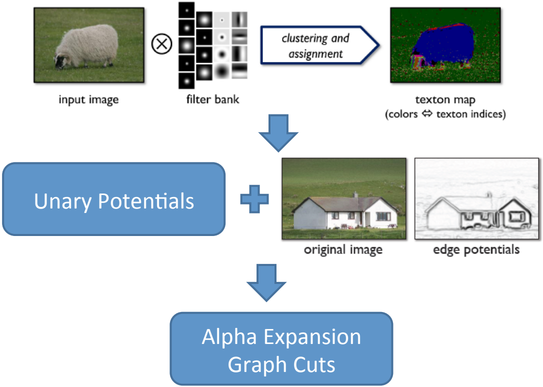
Contents

Definition

- **Semantic segmentation** (or pixel classification) associates one of the **pre-defined** class labels to each pixel
- The input image is divided into the regions, which correspond to the objects of the scene or ‘stuff’



Overview



[Shotton et al., 2006]

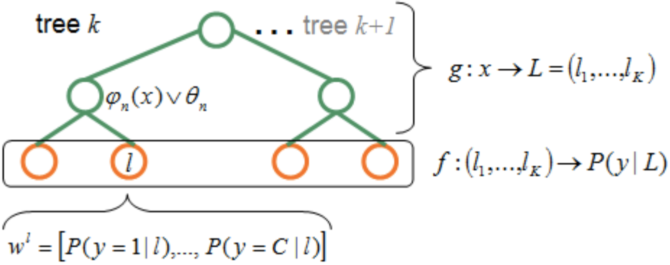
Contents

Difference from Traditional Model

- Only image-level labels for training stage
- How to calculate unary potential from weakly labeled images

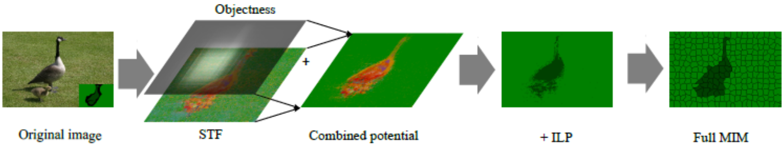
Contents

Figure out Unary Potential from Weakly Labeled Images



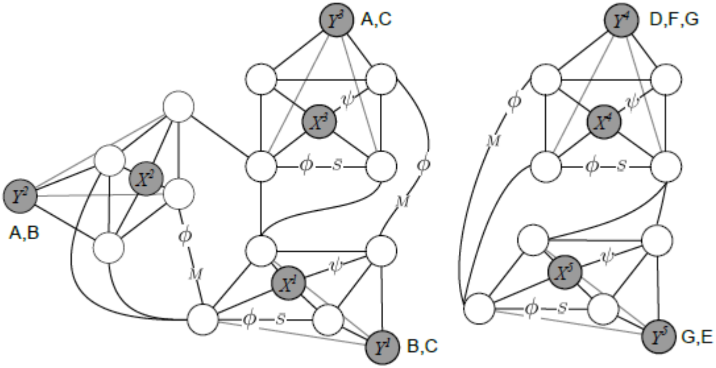
[Vezhnevets and Buhmann, 2010]

Multi-Image Model



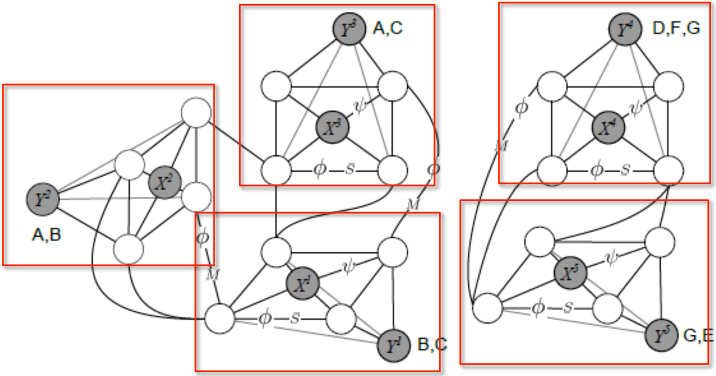
- Unary potential: Naive Bayes appearance model + Objectness prior
- Pairwise potential: Multi-Image Model [Vezhnevets et al., 2011]

Multi-Image Model



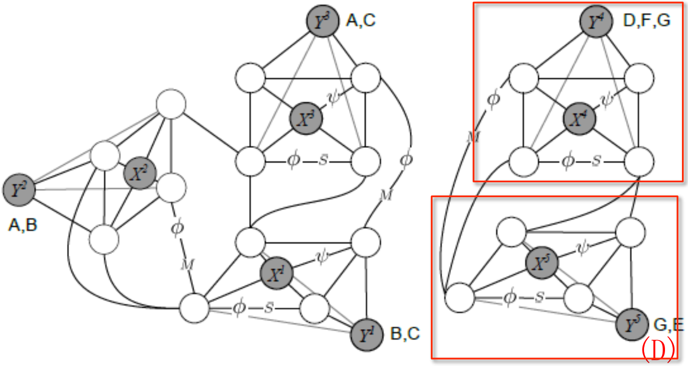
[Vezhnevets et al., 2011]

Multi-Image Model



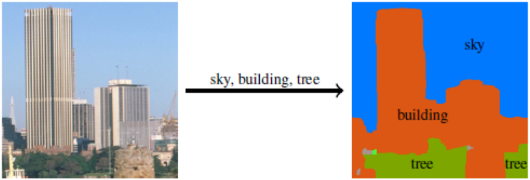
[Vezhnevets et al., 2011]

Multi-Image Model



[Vezhnevets et al., 2011]

Image Level Prior

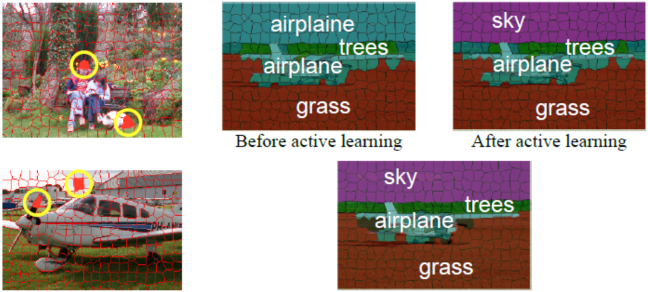


[Xu et al., 2014]

- Significance of Image Level Prior
  - Truth-tag 44% vs. CNN-tag 28%

Active Learning

Active learning — Semantic segmentation on test set

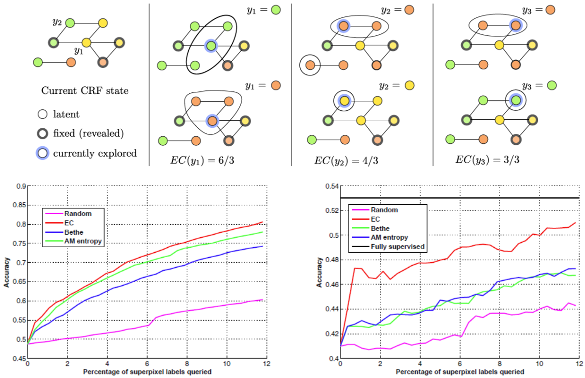


Before active learning — After active learning

Which class are these superpixels? — Test image with overlaid ground truth

[Vezhnevets et al., 2012]

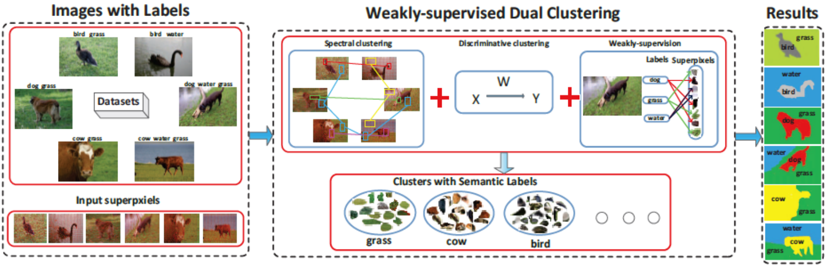
Active Learning



[Vezhnevets et al., 2012]

Contents

Dual Clustering for Semantic Segmentation



[Liu et al., 2013]

Dual Clustering for Semantic Segmentation

- Spectral Clustering

$$\min_{Y,W} Tr[Y^T L Y] + \alpha \left\| X^T W - Y \right\|_F^2 + \beta \left\| W \right\|_{2,1} + \gamma \sum_{i=1}^{I} \sum_{c=1}^{C} \left| \max_{x_{ij} \in X_i} y_{ij}^c - g_{ic} \right|$$

Dual Clustering for Semantic Segmentation

- Spectral Clustering

$$\min_{Y,W} Tr[Y^T LY] + \alpha \left\|X^T W - Y\right\|_F^2 + \beta \left\|W\right\|_{2,1} + \gamma \sum_{i=1}^{I} \sum_{c=1}^{C} \left| \max_{x_{ij} \in X_i} y_{ij}^c - g_{ic} \right|$$

- Discriminative Clustering

Dual Clustering for Semantic Segmentation

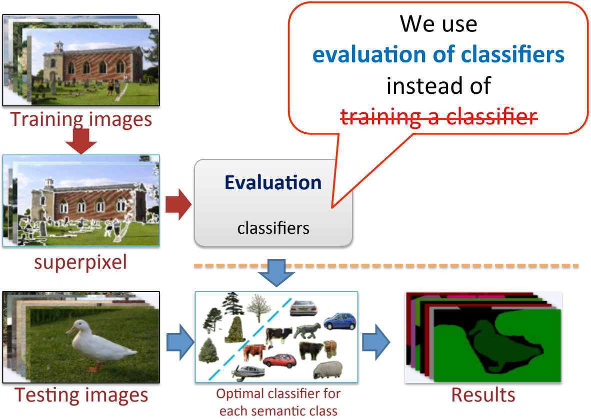
- Spectral Clustering

$$\min_{Y,W} Tr[Y^T L Y] + \alpha \left\| X^T W - Y \right\|_F^2 + \beta \left\| W \right\|_{2,1} + \gamma \sum_{i=1}^{I} \sum_{c=1}^{C} \left| \max_{x_{ij} \in X_i} y_{ij}^c - g_{ic} \right|$$

- Discriminative Clustering
- Weakly-Supervised Constraint

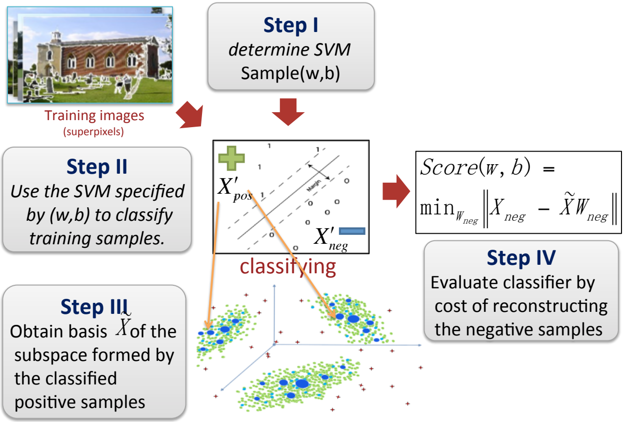
Contents

Classifier Evaluation for Weakly Supervised Learning



[Zhang et al., 2013]

Classifier Evaluation for Weakly Supervised Learning



References I

Hoffman, J., Guadarrama, S., Tzeng, E., EECS, U., Donahue, J., Girshick, R., Darrell, T., and Saenko, K.
From large-scale object classifiers to large-scale object detectors: An adaptation approach.

Kolmogorov, V. and Zabin, R. (2004).
What energy functions can be minimized via graph cuts?
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 26(2):147–159.

Liu, Y., Liu, J., Li, Z., Tang, J., and Lu, H. (2013).
Weakly-supervised dual clustering for image semantic segmentation.
In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2075–2082. IEEE.

Shi, J. and Malik, J. (2000).
Normalized cuts and image segmentation.
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 22(8):888–905.

Shotton, J., Winn, J., Rother, C., and Criminisi, A. (2006).
Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation.
In *Computer Vision–ECCV 2006*, pages 1–15. Springer.

References II

Verbeek, J. and Triggs, B. (2007).
Region classification with markov field aspect models.
In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE.

Vezhnevets, A. and Buhmann, J. M. (2010).
Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning.
In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3249–3256. IEEE.

Vezhnevets, A., Buhmann, J. M., and Ferrari, V. (2012).
Active learning for semantic segmentation with expected change.
In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3162–3169. IEEE.

Vezhnevets, A., Ferrari, V., and Buhmann, J. M. (2011).
Weakly supervised semantic segmentation with a multi-image model.
In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 643–650. IEEE.

Xu, J., Schwing, A. G., and Urtasun, R. (2014).
Tell me what you see and i will show you where it is.
interpretation, 34:12.

References III

Zhang, K., Zhang, W., Zheng, Y., and Xue, X. (2013).
Sparse reconstruction for weakly supervised semantic segmentation.
In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1889–1895. AAAI Press.