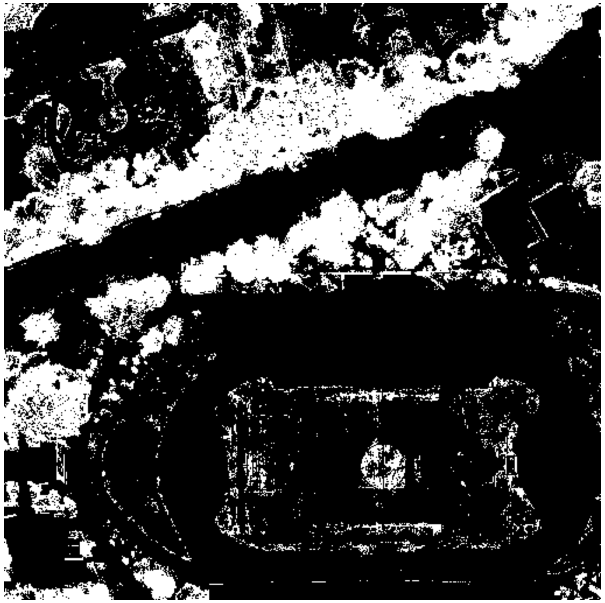
Tree detection from aerial imagery

Lin Yang, Xiaqing Wu, Emil Praun and Xiaoxu Ma

Motivation: 3D city modeling

Overview: pixel labeling

- Assign a {tree, non-tree} label to each pixel.



(a) after classification

(b) after refinement

Overview: tree localization

- Locate individual tree crowns by template matching and greedy selection.

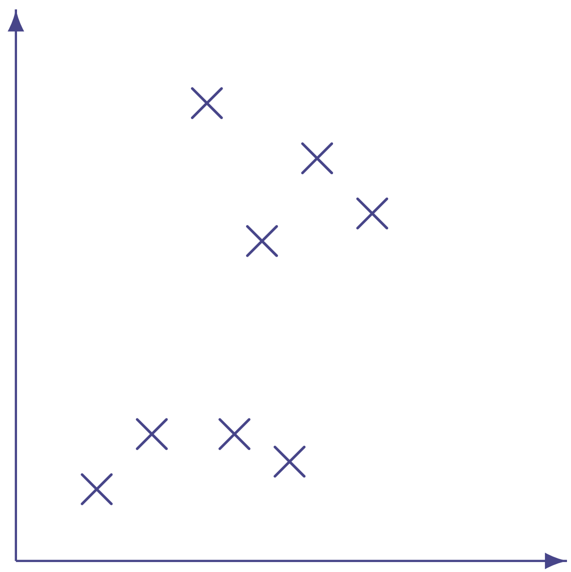
Background: classification

- Classifier: $f(\mathbf{x}) = \{+, -\}$

Background: classification

- Classifier: $f(\mathbf{x}) = \{+, -\}$

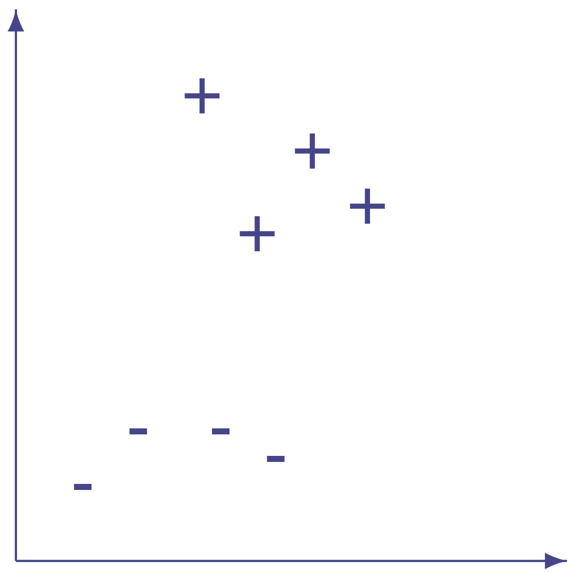
Training: fit $f(\mathbf{x})$



Background: classification

- Classifier: $f(\mathbf{x}) = \{+, -\}$

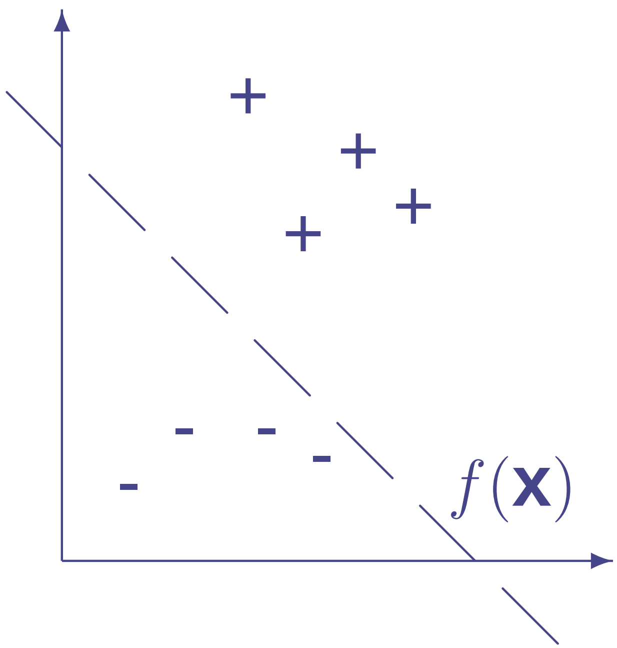
Training: fit $f(\mathbf{x})$



Background: classification

- Classifier: $f(\mathbf{x}) = \{+, -\}$

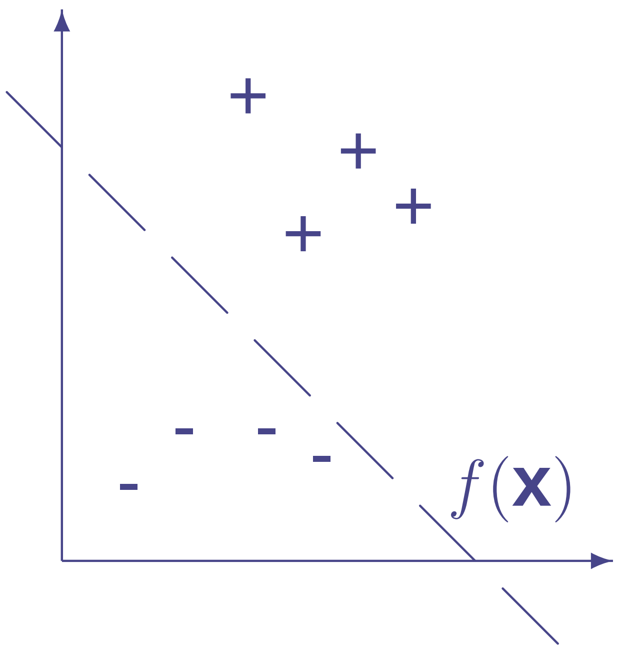
Training: fit $f(\mathbf{x})$



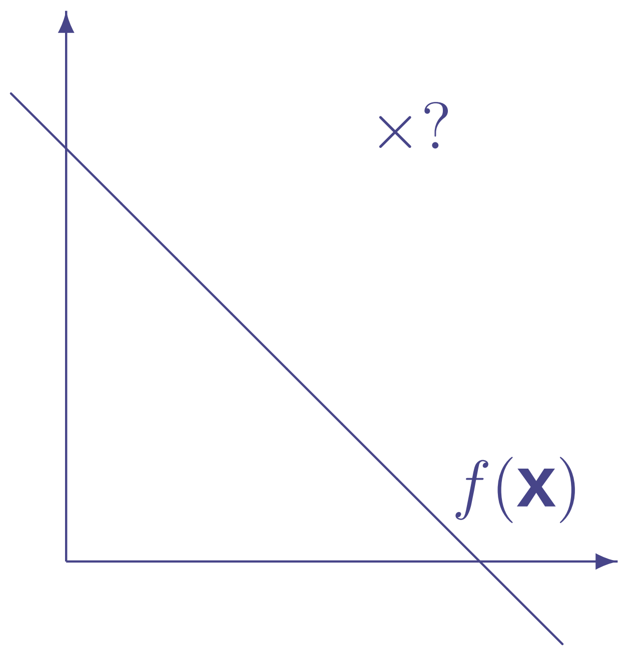
Background: classification

- Classifier: $f(\mathbf{x}) = \{+, -\}$

Training: fit $f(\mathbf{x})$



Classification



Pixel labeling: features

- Color – CIE L*a*b*, Illumination-invariant [Chong *et al*. 2008]
- Texture – edge response detected by Gaussian derivative filter on 3 scales and 6 orientations
- Entropy – entropy on 3 scales
- Other – all informative pixel features such as NIR or LiDAR data can be added if available

Pixel labeling: classification

- Adaboost classification

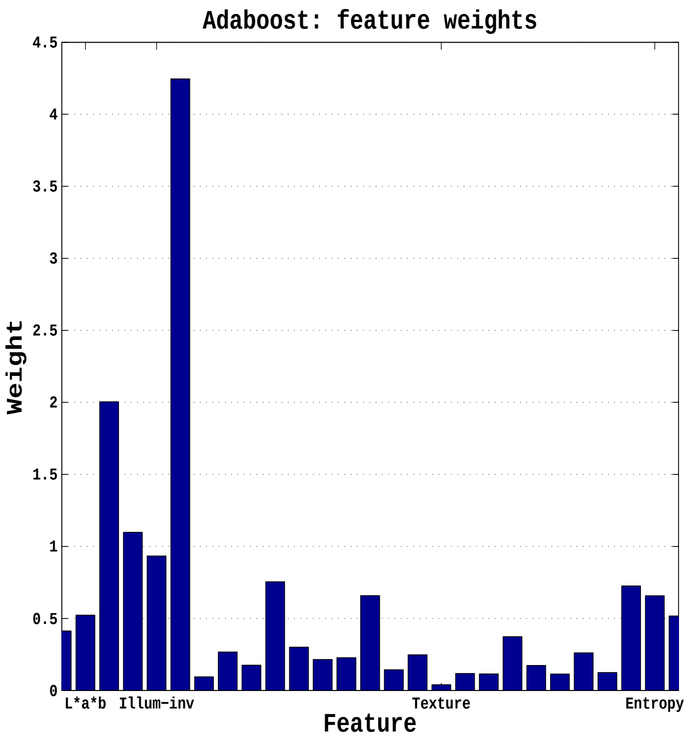
$$H(x) = \sum_{t=1}^{T} \alpha_t h_t(x),$$

where

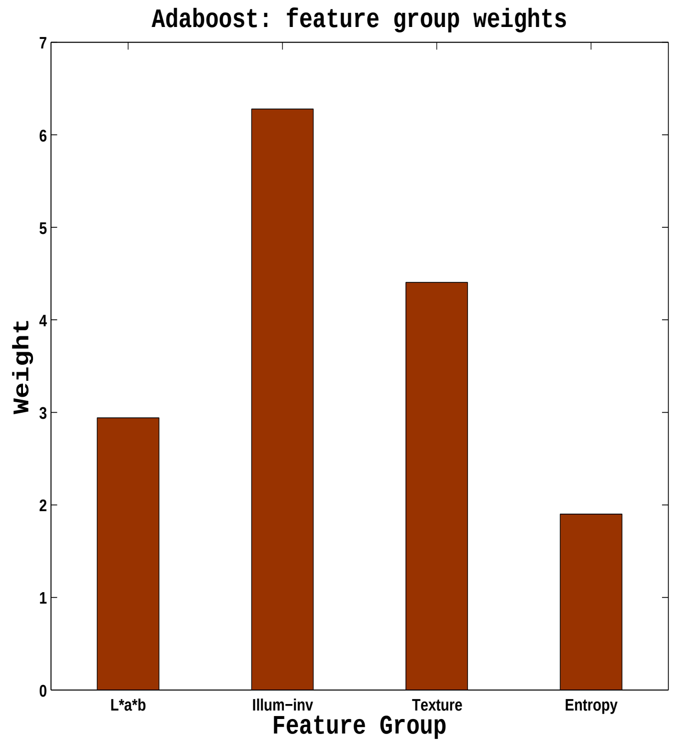
$$h_t(x) = \begin{cases} +1 & s x_{i(t)} > s\theta_t, \\ -1 & otherwise. \end{cases}$$

$$P_{tree}(x) = \frac{1}{1 + e^{-H(x)}}.$$

Pixel labeling: feature analysis



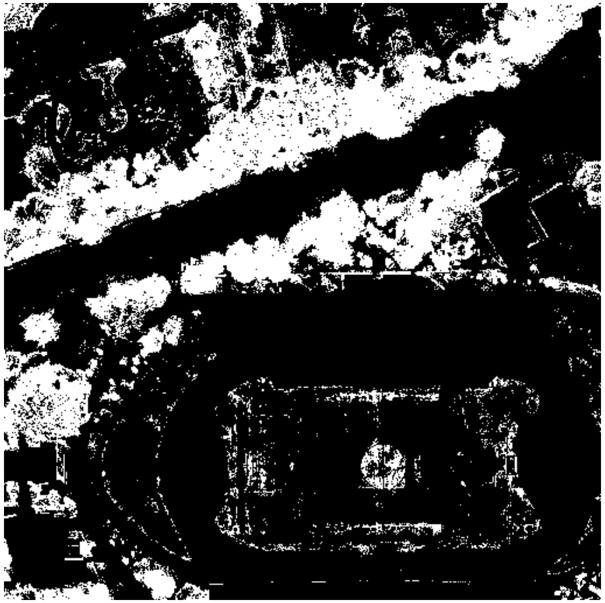
(c) Individual feature weights



(d) Grouped feature weights

Revisit: pixel labeling

- Assign a {tree, non-tree} label to each pixel.



(e) after classification

(f) after refinement

Pixel labeling: refinement

- Graph-cuts

$$E = \sum_{p \in \mathcal{P}} D(p, L(p)) + \sum_{p,q \in \mathcal{N}} V(L(p), L(q)),$$

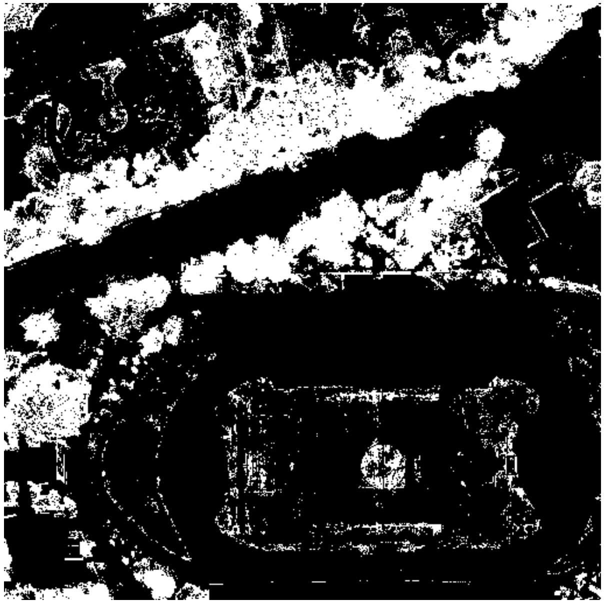
where

$$D(p, L(p)) = \begin{cases} log(1 - P_{tree}(p)) & L(p) = tree, \\ log(P_{tree}(p)) & otherwise, \end{cases}$$

$$V(L(p), L(q)) = \begin{cases} 0 & L(p) = L(q), \\ \beta & otherwise, \end{cases}$$

Revisit: pixel labeling

- Assign a {tree, non-tree} label to each pixel.



(g) after classification

(h) after refinement

Tree localization

- Template matching
  - Match templates on sampled tree regions given by Graph cuts.
  - Compute matching scores as normalized correlation on 4 channels: R, G, B and the tree probability given by Adaboost.
- Greedy selection
  - Sort matched templates in descending order of matching scores.
  - Iteratively select the top match and remove all the overlapped matches.

Sample result 1

Sample result 2

Sample result 3

Sample result 4

Tree detection for the globe

- Scalability: a bottleneck for many GIS applications
- An even bigger bottleneck when supervised training is involved (Adaboost)
- Solution: clustering regions of similar geographic features so that Adaboost can achieve reasonable accuracy with tiny training data.

Aerial image clustering

- Encode aerial images by Gist descriptors.
  - Gist [Oliva and Torralba 2001] computes a low dimensional representation for the scene structure of an image.
  - Gist has been shown to convey higher level semantics of images in the computer vision community [Hays and Efros, 2007].
- Reduce the dimensionality of Gist descriptors.
  - Principal Component Analysis (PCA) is applied to reduce the dimensionality of Gist descriptors from 512 to 12.
  - The 12 projected descriptors preserve about 95% of the eigenvalues of the original 512 descriptors.

Performance evaluation

- Evaluation on the New York data set (4500 tiles)

Training: 1% (45 tiles)		Baseline		Cluster-I		Cluster-II	
		Tree	NonT.	Tree	NonT.	Tree	NonT.
Ground-Truth Labels	Tree	5.6%	0.5%	5.4%	0.7%	5.1%	0.9%
	NonT.	12.0%	81.0%	10.3%	83.7%	8.4%	86.6%
Accuracy		86.6%		89.1%		91.7%	

  - Baseline: training tiles are uniformly sampled in the bounding region.
  - Cluster-I: k-means are applied to the Gist image descriptors to divide the tiles into 45 clusters, and the training tiles are selected as cluster centroids.
  - Cluster-II: two-level clustering is applied to the Gist image descriptors. The first level k-means divides the tiles into 4 clusters. Within each cluster, the method of Cluster-I is used to select 1% as training tiles. The training tiles of each cluster are used to train a **separate** classifier dedicated to the tiles in that cluster.

Summary

- Automatic approach to tree detection from aerial imagery
- Simple and extensible algorithm
  - Relies on image features only.
  - Open for incorporation of NIR and LiDAR data.
- Efficient and scalable to the globe
  - Takes around 30 seconds to process a 512×512 tile.
  - Takes around 1% training data to achieve 90% accuracy in pixel classification.

- Lin Yang, Xiaqing Wu, Emil Praun and Xiaoxu Ma. Tree detection from aerial imagery. *ACM SIGSPATIAL GIS 2009*, to appear.