Detecting and Reading Text in Natural Scenes

Xiangrong Chen, Alan L. Yuille
{xrchen, Yuille}@stat.ucla.edu

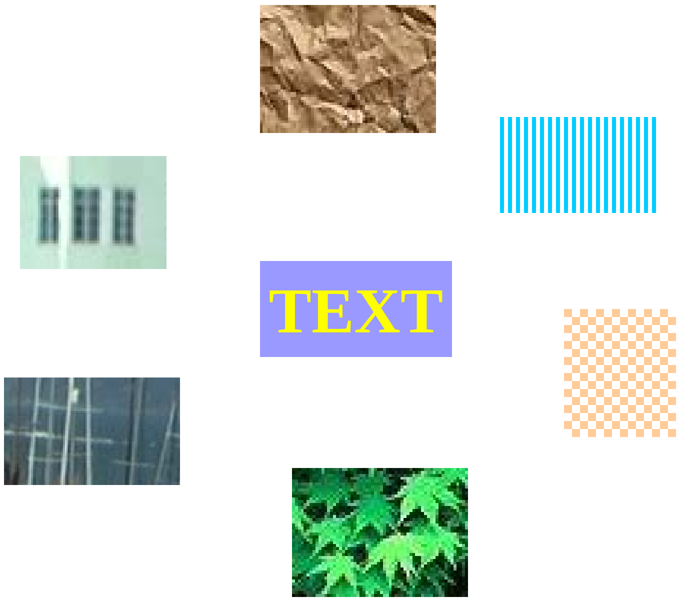
Statistics dept, UCLA

Outline

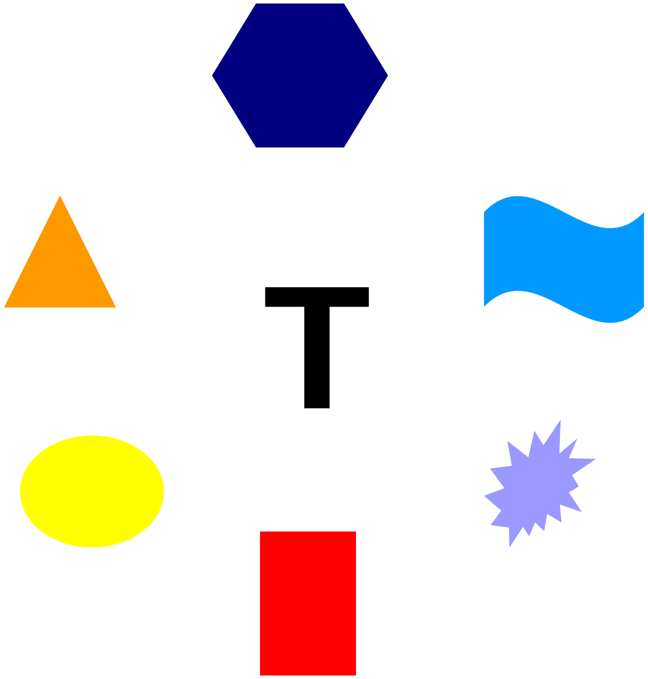
- Background
- Overview of our method
- Detecting text
- Reading text
- Experiments
- Summary

Text detection methods

Text as texture



Text as connected component



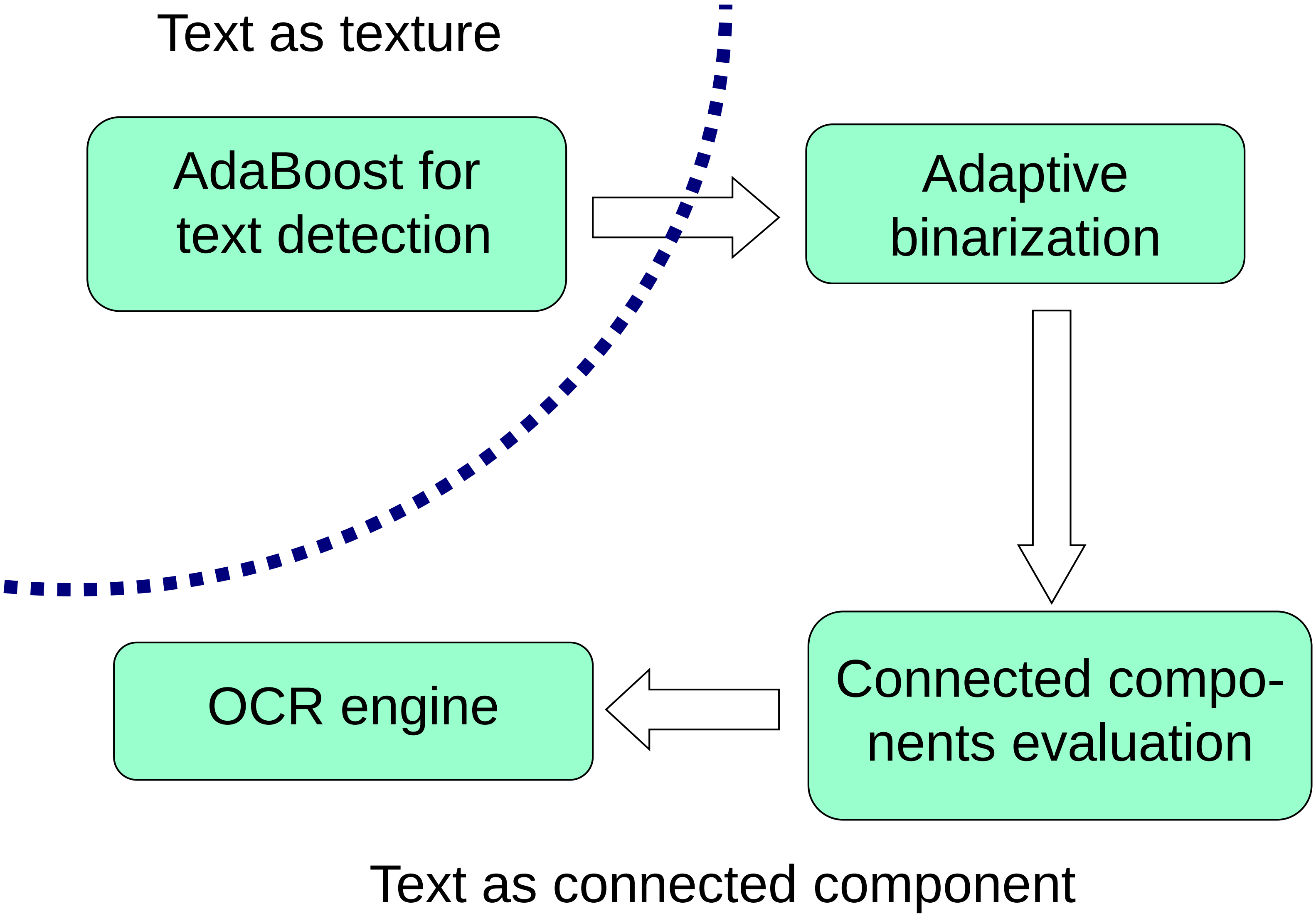
Comparison

Text as	texture	connected component
Feature	Texture analysis	Shape, structure and appearance analysis
Searching method	Scan the image using a small window in different scales	Enumerate all the CCPS; need image segmentation to obtain the CCPs
Pros	Easy to deal with scale and complex background; scan quickly	Easily lead to generative model and thus can guide recognition task
Cons	Discriminant model; a black box, not easy to guide recognition task	No good enough segmentation algorithm available to get CCPs

Combination

- Find candidate area using text as texture
- Verify using text as connected component

Proposed method



Why using AdaBoost

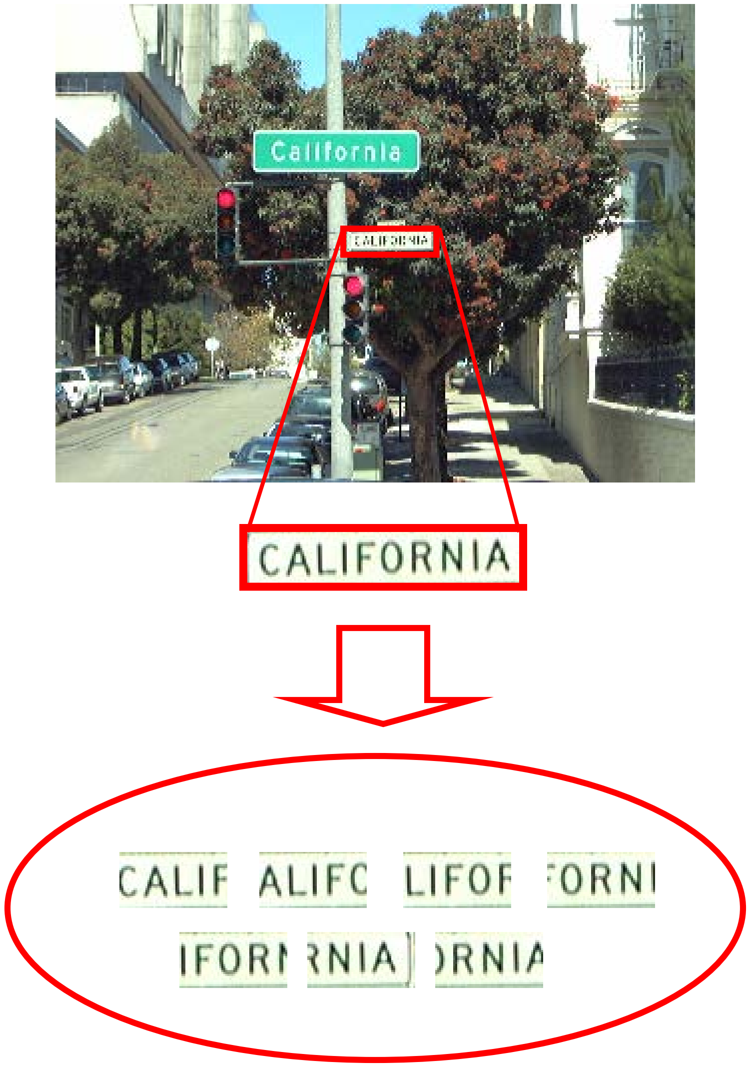
- Improves classification accuracy
- Can be used with many different classifiers
- Simple to implement
- Not prone to overfitting

Training data

- 162 Source images by normal and blind people

- Manually label text regions

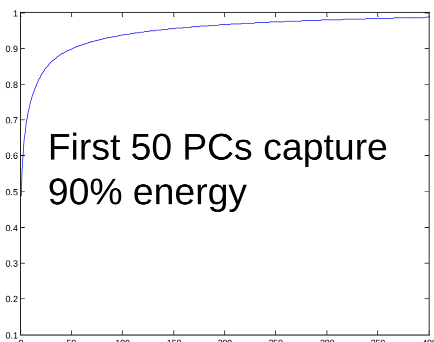
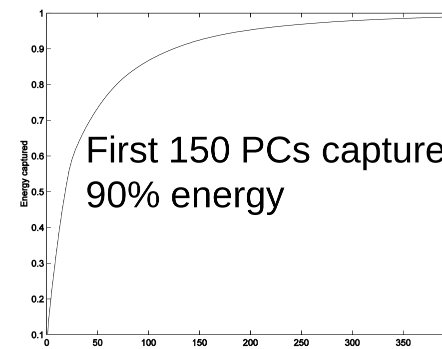
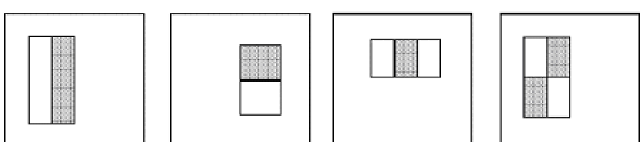
- Cut the text regions into overlapped training samples with fixed width-to-height ratio, 2:1



Features – Criterion

- Informative
  - Invariant for text regions
  - Discriminating between text and non-text regions
- Cost
  - Computation

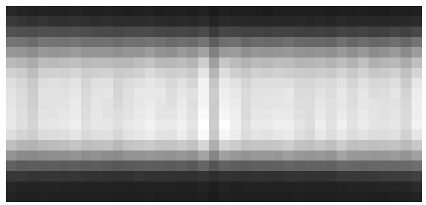
Features-Training samples

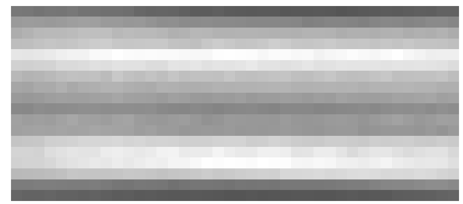
	Face		Text
Raw data			
Align, Crop & Scale		4,000 faces 32 × 32	4,000 patches 20 × 40 
PCA		Mean face 	Mean patch  
Features			?

Features – Set I

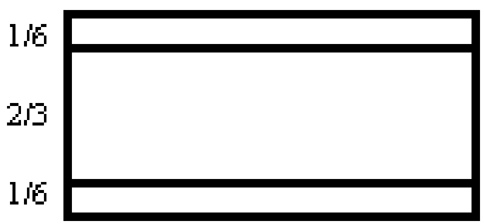
- 1st order derivatives

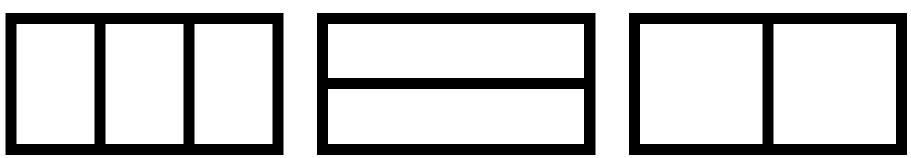
Mean of $\left|\frac{dI}{dx}\right|$





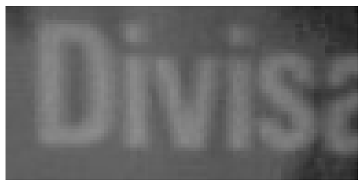
Mean of $\left|\frac{dI}{dy}\right|$

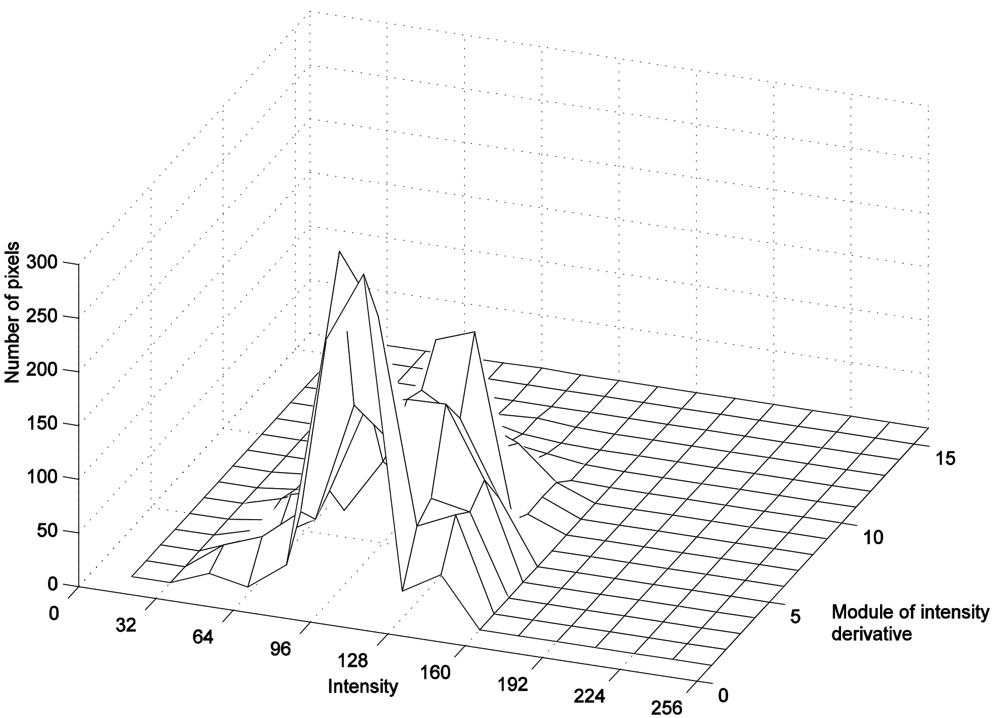


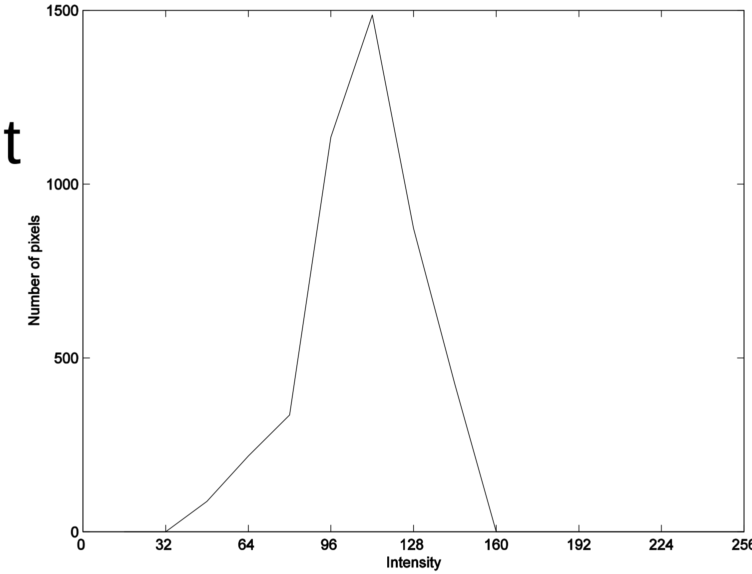


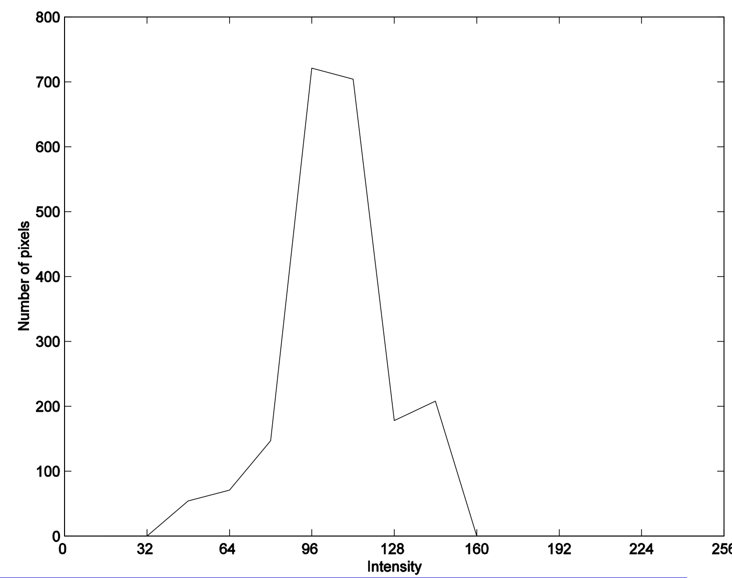
Features – Set II

- Histogram of Intensity and gradient









Features – Set III

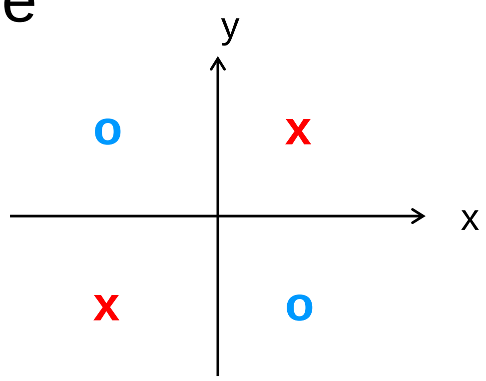
- Edge linking features

  edge map → thinning → linking

  Using statistics of the length of the linked edges

Weak learners

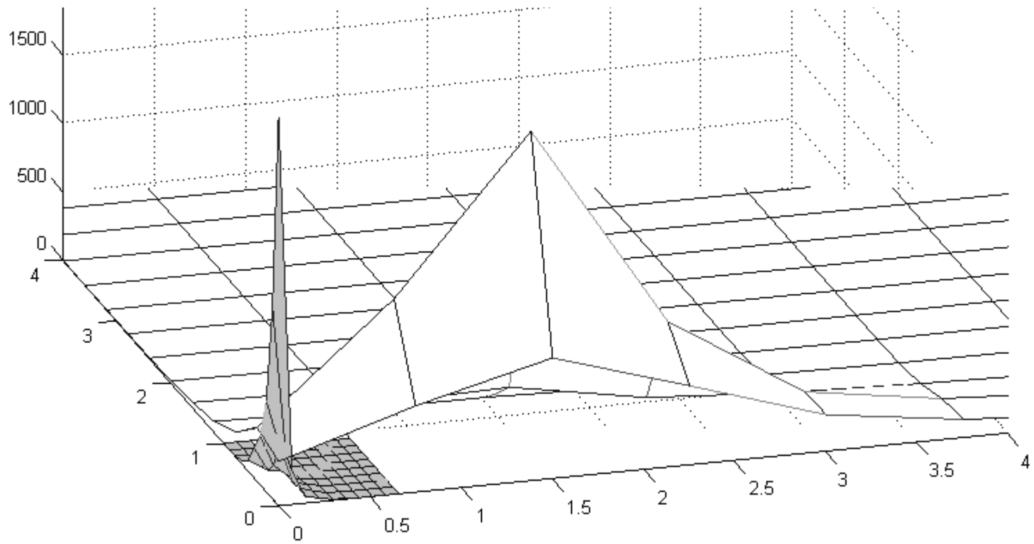
- Ability of the strong classifier is determined by the ability of the weak learners
- Strong classifier with 1D stub weak learners can't deal with the example



- We use log-likelihood ratio test on distributions of both single features and pairs of features as weak learners (Konishi and Yuille, 2003)

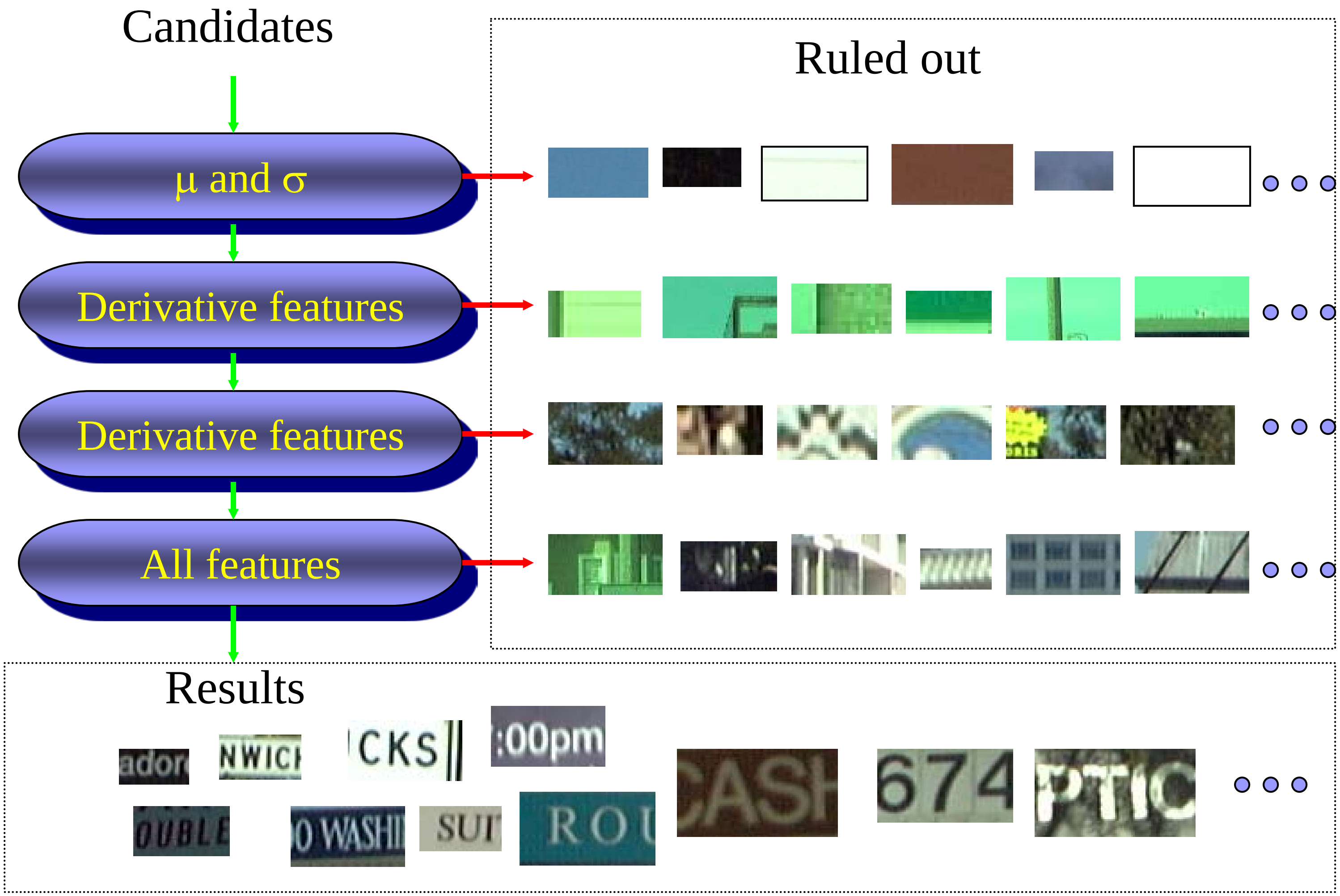
An example of Weak learners

- Joint distribution of a pair of features form the first weak learner AdaBoost selected

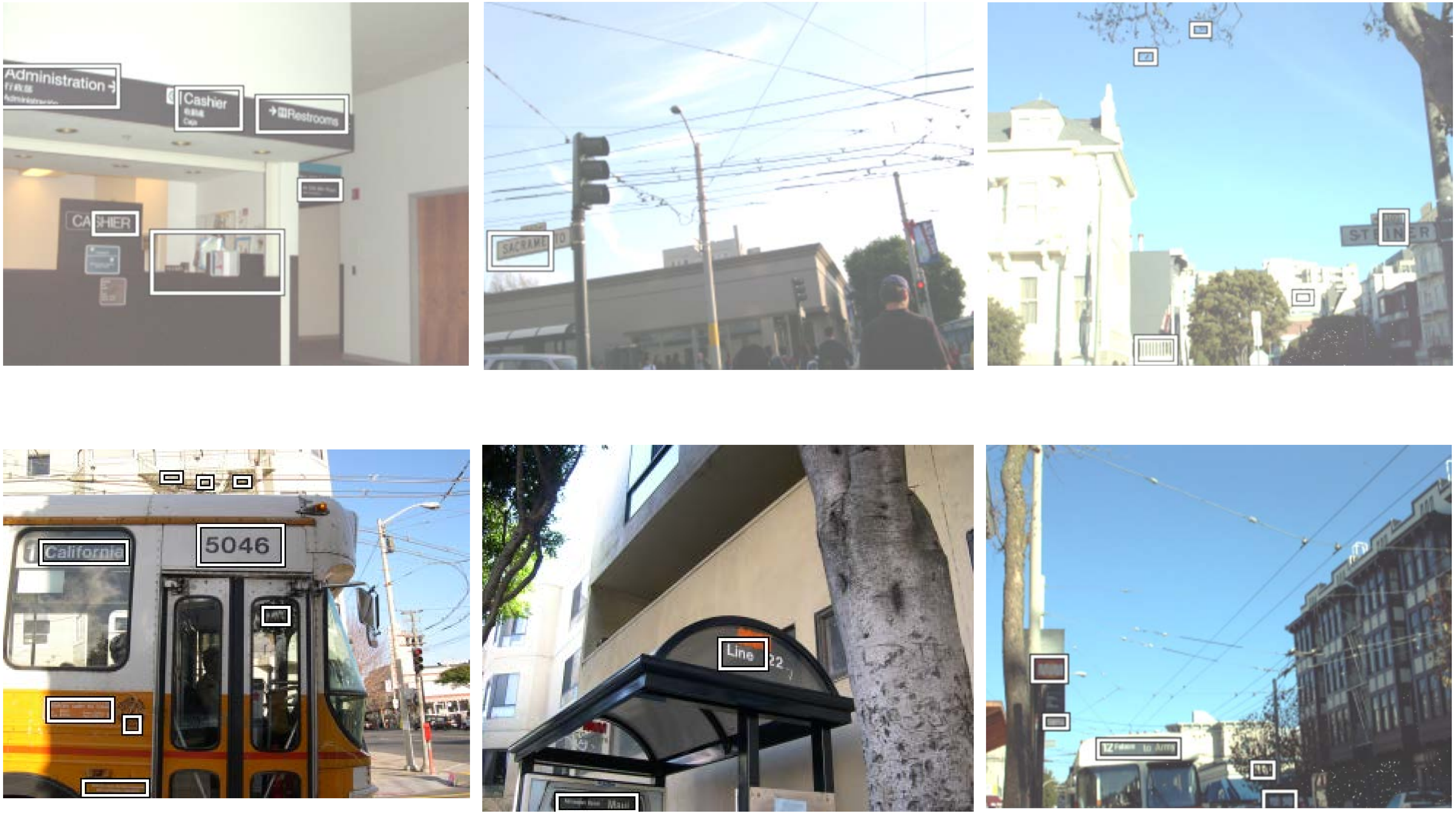


Text distribution is shaded.

Cascade of strong classifiers



Text detection examples



Fail to detect

- Vertically aligned text
- Individual letters
- Extreme cases

Adaptive binarization

- Ni'Black's method

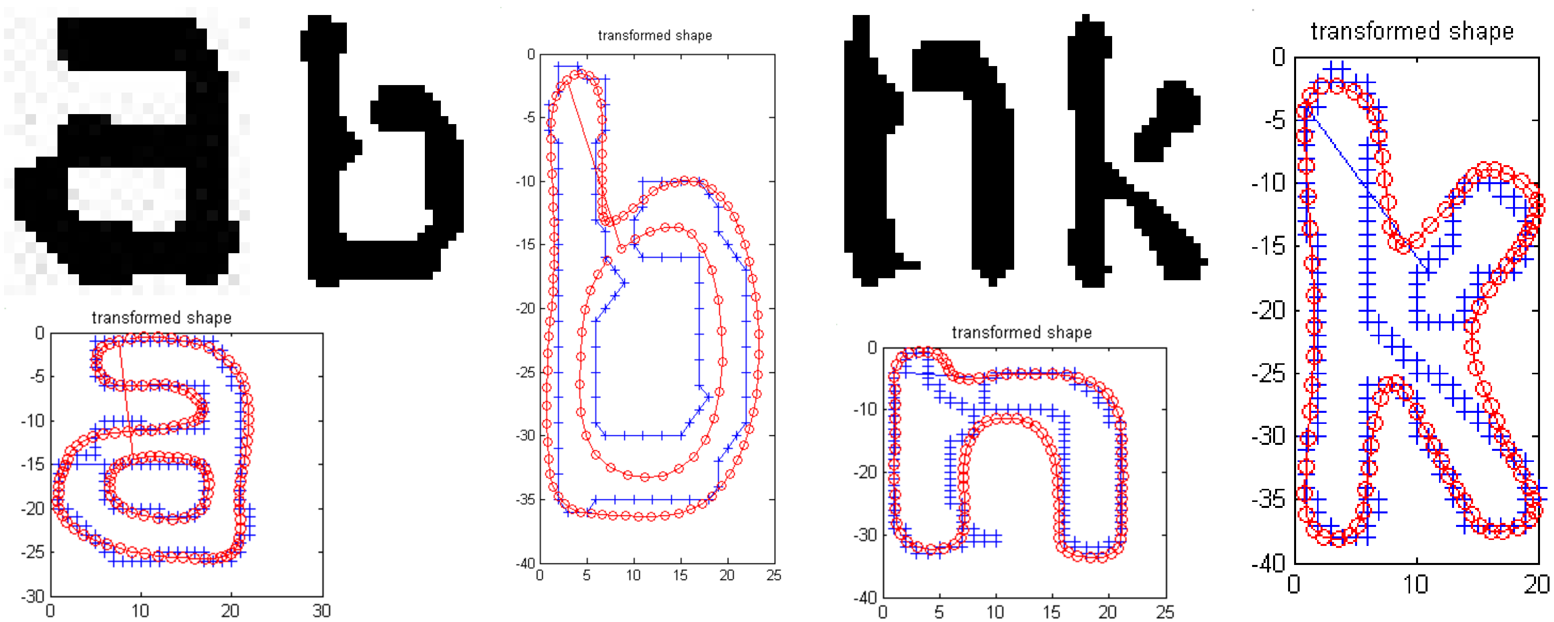
$$T_r(x) = \mu_r(x) + k \bullet \sigma_r(x)$$

- Determine range of neighborhood size
  - Relative to the sub-window height h

$$r(x) = \min_{r \subset R(h)} \{\sigma_r(x) > T_0\}$$

OCR engine

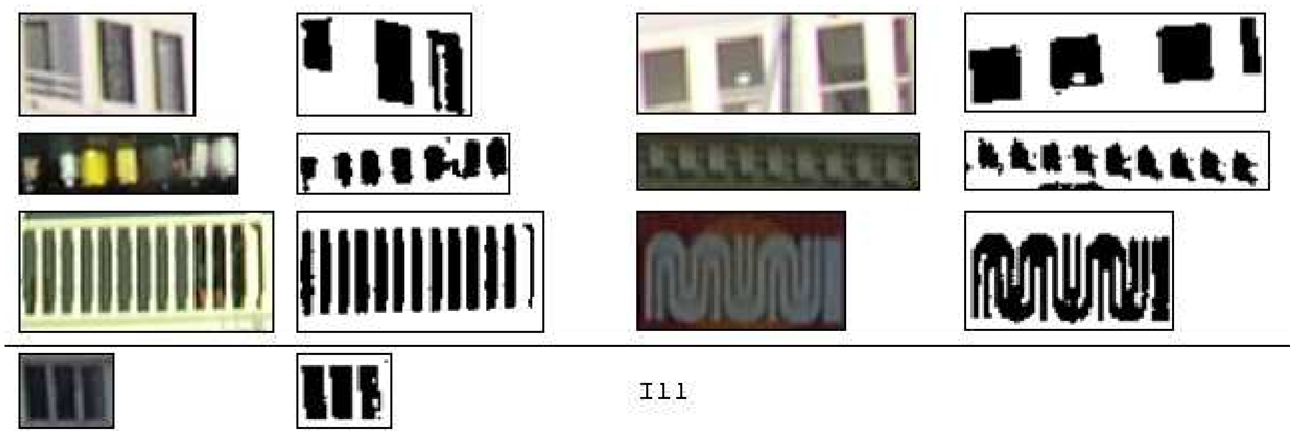
- Currently we use a commercial OCR engine
- A generative model for reading text is under developing



Text reading examples

False positives

- Building structures
- Signs or icons
- Tree leaves and branches



Results

- Accuracy
  - False Negative for detection 2.8%
  - False Positive for detection ~ 1/200,000
  - False Negative for reading 7%
  - False Positive for reading 10% (1% w/ constraint to form coherent word)
- Speed
  - 3 Seconds for 2,048*1536 image ~ 15fps for 320*240 video frames

Summary

- Using Adaboost to learn a strong classifier for detecting text in unconstrained scenes
- Selection of informative features with consideration of computation cost
- Detecting and reading over 90% text regions in our database
- Real-time (15fps) for video quality images (320 * 240)

ICDAR's competition

- Database