

Learning to Detect Faces Rapidly and Robustly

- Hierarchy of Visual Attention Operators
- Automatic Selection of Discriminative Features

Paul Viola Mike Jones
MERL Compaq
Cambridge MA

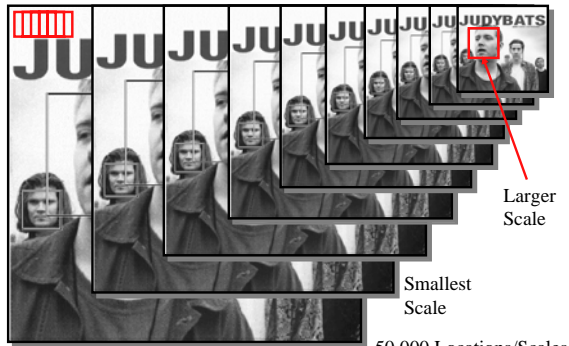
Viola 2001

Face Detection Example



Viola 2001

The Classical Face Detection Process



Viola 2001

Classifier is Learned from Labeled Data

- Training Data
 - 5000 faces
 - All frontal
 - 10^8 non faces
 - Faces are normalized
 - Scale, translation
- Many variations
 - Across individuals
 - Illumination
 - Pose (rotation both in plane and out)



Viola 2001

Key Properties of Face Detection

- Each image contains 10 - 50 thousand locs/scales
- Faces are rare 0 - 50 per image
 - 1000 times as many non-faces as faces
- Extremely small # of false positives: 10^{-6}

Viola 2001

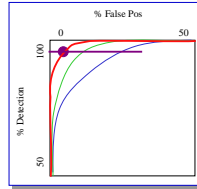
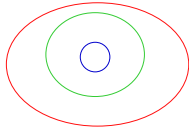
Overview

- Cascaded Classifier for rapid detection
 - Hierarchy of Attentional Filters
- Feature set (... is huge about 6,000,000 features)
- Efficient feature selection using AdaBoost
- New image representation: Integral Image

Viola 2001

Trading Speed for Accuracy

- Given a nested set of classifier hypothesis classes

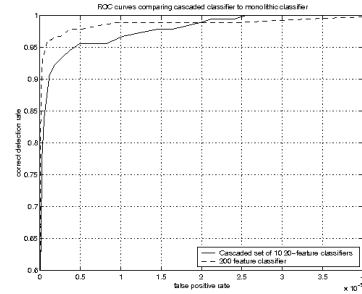


- Computational Risk Minimization



Viola 2001

Experiment: Simple Cascaded Classifier



Viola 2001

Cascaded Classifier



- A 1 feature classifier achieves 100% detection rate and about 50% false positive rate.
- A 5 feature classifier achieves 100% detection rate and 40% false positive rate (20% cumulative)
 - using data from previous stage.
- A 20 feature classifier achieve 100% detection rate with 10% false positive rate (2% cumulative)

Viola 2001

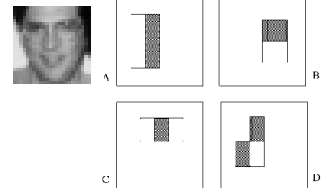
Image Features

“Rectangle filters”

Similar to Haar wavelets

Papageorgiou, et al.

Differences between sums of pixels in adjacent rectangles



$$h_t(x) = \begin{cases} +1 & \text{if } f_t(x) > \theta_t \\ -1 & \text{otherwise} \end{cases}$$

60,000 × 100 = 6,000,000
Unique Features

Viola 2001

Constructing Classifiers

- Feature set is very large and rich
- Perceptron yields a sufficiently powerful classifier

$$C(x) = \theta \left(\sum_i \alpha_i h_i(x) + b \right)$$

- 6,000,000 Features & 10,000 Examples
 - 60,000,000,000 feature values!
- Classical feature selection is infeasible
 - Wrapper methods
 - Exponential Gradient (Winnow - Roth, et al.)

Viola 2001

AdaBoost

$$f(x) = \theta \left(\sum_i \alpha_i h_i(x) \right)$$

$$\alpha_i = 0.5 \log \left(\frac{\text{error}_i}{1 - \text{error}_i} \right)$$

$$w_i^j = \frac{w_{i-1}^j e^{-y_i \alpha_i h_i(x_i)}}{\sum_i w_{i-1}^j e^{-y_i \alpha_i h_i(x_i)}}$$

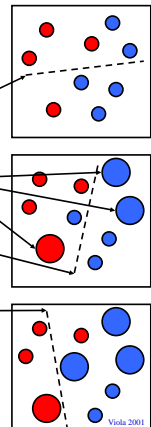
Initial uniform weight on training examples

weak classifier 1

Incorrect classifications re-weighted more heavily

weak classifier 2

Final classifier is weighted combination of weak classifiers



Viola 2001

Beautiful AdaBoost Properties

- Training Error approaches 0 exponentially
- Bounds on Testing Error Exist
 - Analysis is based on the Margin of the Training Set
- Weights are related the margin of the example
 - Examples with negative margin have large weight
 - Examples with positive margin have small weights

$$f(x) = \sum_i \alpha_i h_i(x) \quad \min \sum_i e^{-y_i f(x_i)} \geq \sum_i (1 - y_i C(x_i))$$

$$C(x) = \theta(f(x))$$

Viola 2001

AdaBoost for Efficient Feature Selection

- Our Features = Weak Classifiers
- For each round of boosting:
 - Evaluate each rectangle filter on each example
 - Sort examples by filter values
 - Select best threshold for each filter (min error)
 - Sorted list can be quickly scanned for the optimal threshold
 - Select best filter/threshold combination
 - Weight on this feature is a simple function of error rate
 - Reweight examples
 - (There are many tricks to make this more efficient.)

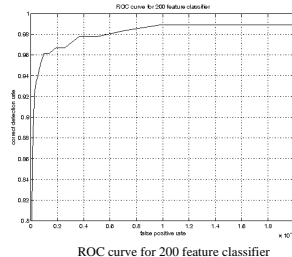
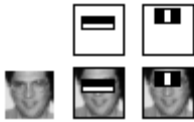
Viola 2001

Example Classifier for Face Detection

A classifier with 200 rectangle features was learned using AdaBoost

95% correct detection on test set with 1 in 14084 false positives.

Not quite competitive...



ROC curve for 200 feature classifier

Viola 2001

Training the Cascade

Training faces: 5000 manually cropped faces from web images (24 x 24 pixels)

Training non-faces: 350 million sub-windows from 9500 non-face images

Cascaded classifier with 32 layers was trained. The number of features per layer was 1, 5, 20, 20, 50, 50, 100, ..., 200, ...

Each layer was trained on false positives of previous layers (up to 5000 non-face sub-windows)

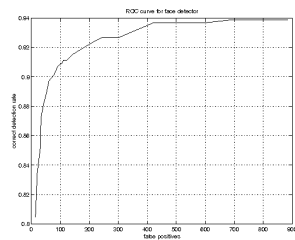
Final classifier contains 4297 features.



Viola 2001

Accuracy of Face Detector

MIT+CMU test set: 130 images, 507 faces and 75,000,000 subwindows



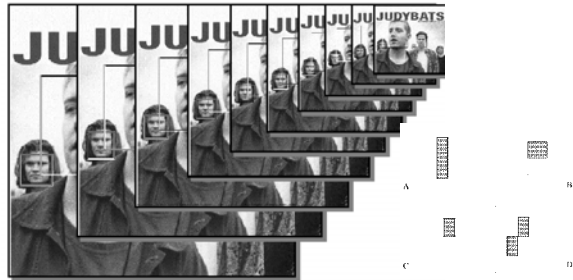
Viola 2001

Comparison to Other Systems

False Detections	10	31	50	65	78	95	110	167	422
Detector									
Viola-Jones	78.3	85.2	88.8	89.8	90.1	90.8	91.1	91.8	93.7
Rowley-Baluja-Kanade	83.2	86.0				89.2		90.1	89.9
Schneiderman-Kanade				94.4					
Roth-Yang-Ahuja					(94.8)				

Viola 2001

Pyramids are not free



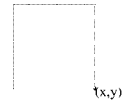
Takes about 0.06 seconds per image
(Detector takes about 0.06 secs!)

Viola 2001

Integral Image

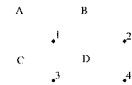
- Define the Integral Image

$$I'(x, y) = \sum_{x' \leq x} \sum_{y' \leq y} I(x', y')$$



- Any rectangular sum can be computed in constant time:

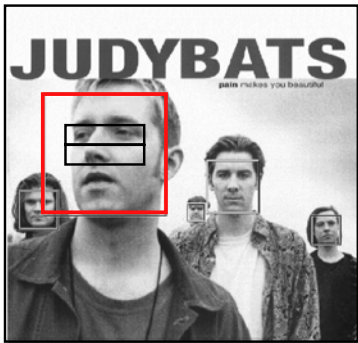
$$\begin{aligned} D &= 1 + 4 - (2 + 3) \\ &= A + (A + B + C + D) - (A + C + A + B) \\ &= D \end{aligned}$$



- Rectangle features can be computed as differences between rectangles

Viola 2001

Scale the Detector, not the Image



Viola 2001

Speed of Face Detector

Speed is proportional to the average number of features computed per sub-window.

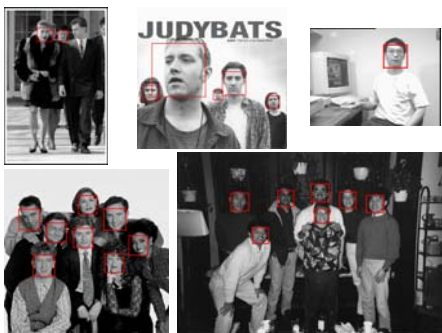
On the MIT+CMU test set, an average of 8 features out of a total of 4297 are computed per sub-window.

On a 700 Mhz Pentium III, a 384x288 pixel image takes about 0.063 seconds to process (15 fps).

Roughly 15 times faster than Rowley-Baluja-Kanade and 600 times faster than Schneiderman-Kanade.

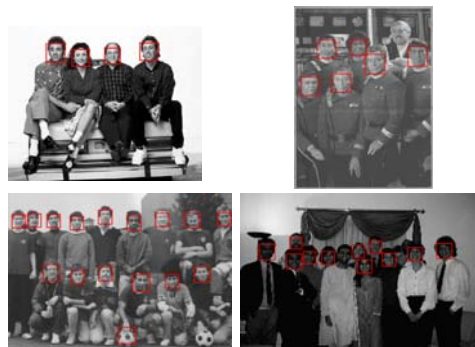
Viola 2001

Output of Face Detector on Test Images



Viola 2001

More Examples



Viola 2001

Results



Viola 2001

Video Demo



Viola 2001

Conclusions

- 3.5 contributions
 - Cascaded classifier yields rapid classification
 - AdaBoost as an extremely efficient feature selector
 - Rectangle Features + Integral Image can be used for rapid image analysis

Viola 2001

Related Work

- Romdhani, Torr, Scholkopf and Blake
 - Accelerate SVM by approximating decision boundary one vector at a time.
 - Introduce a threshold to reject an example after each additional vector
- Amit & Geman and Fleuret & Geman
- Our system:
 - Simplicity, Uniformity, and Interpretability
 - Directly construct discriminative features
 - Rather than density estimation based learning
 - Highly efficient features and lower false positives

Viola 2001