Face detection and recognition

Bill Freeman, MIT

6.869 April 7, 2005

Today (April 7, 2005)

- Face detection
  - Subspace-based
  - Distribution-based
  - Neural-network based
  - Boosting based
- Face recognition, gender recognition

Some slides courtesy of: Baback Moghaddam, Trevor Darrell, Paul Viola

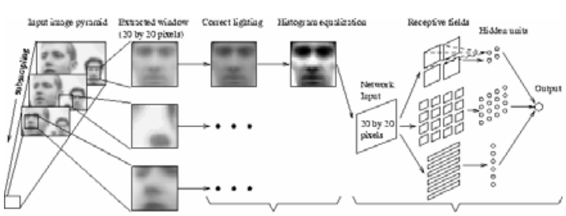
Readings

- Face detection:
  - Forsyth, ch 22 sect 1-3.
  - "Probabilistic Visual Learning for Object Detection," Moghaddam B. and Pentland A., *International Conference on Computer Vision*, Cambridge, MA, June 1995. ,(http://www-white.media.mit.edu/vismod/publications/techdir/TR-326.ps.Z)
- Brief overview of classifiers in context of gender recognition:
  - http://www.merl.com/reports/docs/TR2000-01.pdf, ***Gender Classification with Support Vector Machines*** Citation: Moghaddam, B.; Yang, M-H., "Gender Classification with Support Vector Machines", *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pps 306-311, March 2000
- Overview of subspace-based face recognition:
  - Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000 (Elsevier Science, http://www.merl.com/reports/docs/TR2000-42.pdf)
- Overview of support vector machines—Statistical Learning and Kernel MethodsBernhard Schölkopf, ftp://ftp.research.microsoft.com/pub/tr/tr-2000-23.pdf

Face detectors

- Subspace-based
- Distribution-based
- Neural network-based
- Boosting-based

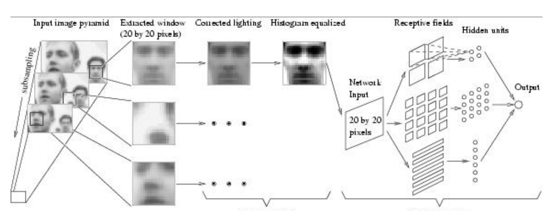
The basic algorithm used for face detection



From: http://www.ius.cs.cmu.edu/IUS/har2/har/www/CMU-CS-95-158R/

Neural Network-Based Face Detector

- Train a set of multilayer perceptrons and arbitrate a decision among all outputs [Rowley et al. 98]



From: http://www.ius.cs.cmu.edu/IUS/har2/har/www/CMU-CS-95-158R/

"Eigenfaces"

Figure 4: Standard Eigenfaces.

Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000

Computing eigenfaces by SVD

svd(X,0) gives $X = U\,S\,V^T$

Covariance matrix $XX^T = U\,S\,V^T\,V\,S\,U^T$
$= U\,S^2\,U^T$

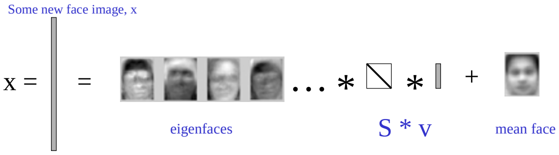
So the U's are the eigenvectors of the covariance matrix X

Computing eigenfaces by SVD

svd(X,0) gives $X = U\,S\,V^T$

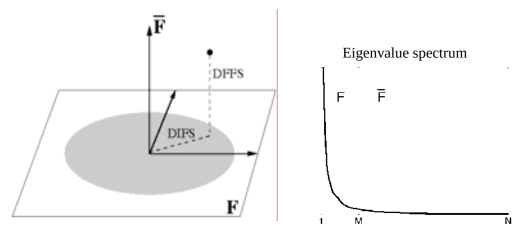
Covariance matrix $XX^T = U\,S\,V^T\,V\,S\,U^T$
$= U\,S^2\,U^T$

Some new face image, x



Subspace Face Detector

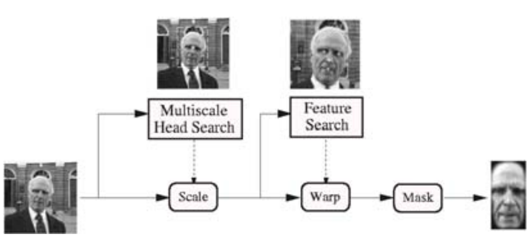
- PCA-based Density Estimation p(x)
- Maximum-likelihood face detection based on DIFS + DFFS



Moghaddam & Pentland, "Probabilistic Visual Learning for Object Detection," ICCV'95.

Subspace Face Detector

- Multiscale Face and Facial Feature Detection & Rectification



Moghaddam & Pentland, "Probabilistic Visual Learning for Object Detection," ICCV'95.

Today (April 7, 2005)

- Face detection
  - Subspace-based
  - Distribution-based
  - Neural-network based
  - Boosting based
- Face recognition, gender recognition

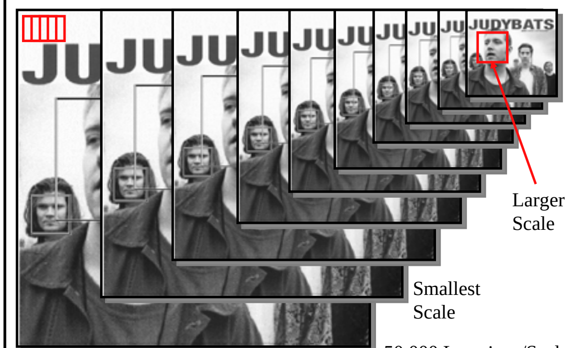
Some slides courtesy of: Baback Moghaddam, Trevor Darrell, Paul Viola

Rapid Object Detection Using a Boosted Cascade of Simple Features

Paul Viola Michael J. Jones
Mitsubishi Electric Research Laboratories (MERL)
Cambridge, MA

Most of this work was done at Compaq CRL before the authors moved to MERL

The Classical Face Detection Process



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 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 and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

What is novel about this approach?

- Feature set (… is huge about 16,000,000 features)
- Efficient feature selection using AdaBoost
- New image representation: Integral Image
- Cascaded Classifier for rapid detection
  - Hierarchy of Attentional Filters

The combination of these ideas yields the fastest known face detector for gray scale images.

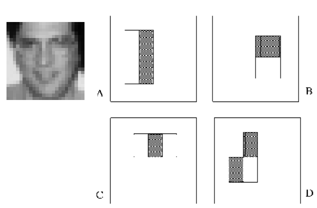
Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Image Features

"Rectangle filters"

Similar to Haar wavelets

Differences between sums of pixels in adjacent rectangles



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

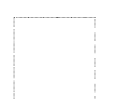
$160{,}000 \times 100 = 16{,}000{,}000$
Unique Features

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Integral Image

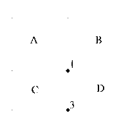
- Define the Integral Image

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



- Any rectangular sum can be computed in constant time:

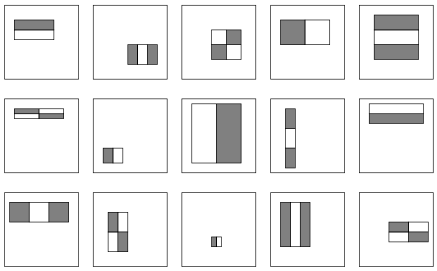
$$D = 1 + 4 - (2+3)$$
$$= A + (A+B+C+D) - (A+C+A+B)$$
$$= D$$



- Rectangle features can be computed as differences between rectangles

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Huge "Library" of Filters



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Constructing Classifiers

- Perceptron yields a sufficiently powerful classifier

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

- Use AdaBoost to efficiently choose best features

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Flavors of boosting

• Different boosting algorithms use different loss functions or minimization procedures
(Freund & Shapire, 1995; Friedman, Hastie, Tibshhirani, 1998).

• We base our approach on Gentle boosting: learns faster than others
(Friedman, Hastie, Tibshhirani, 1998;
Lienahart, Kuranov, & Pisarevsky, 2003).

Additive models for classification, "gentle boost"

$$H(v,c) = \sum_{m=1}^{M} h_m(v,c)$$

classes

+1/-1 classification

feature responses

(in the face detection case, we just have two classes)

(Gentle) Boosting loss function

We use the exponential multi-class cost function

classes

$$J = \sum_{c=1}^{C} E\left[e^{-z^c H(v,c)}\right]$$

cost function

membership in class c, +1/-1

classifier output for class c

Weak learners

At each boosting round, we add a perturbation or "weak learner":

$$H(v_i, c) := H(v_i, c) + h_m(v_i, c)$$

Use Newton's method to select weak learners

Treat h_m as a perturbation, and expand loss J to second order in h_m

$$J(H+h_m) \approx E(e^{-z^c H(v,c)})[2-2z^c h_m + (z^c)^2 h_m^2]$$

$$\underbrace{\arg\min_{h_m} J}_{\text{cost function}}(\underbrace{H+h_m}_{\text{classifier with perturbation}}) \simeq \arg\min_{h_m} \sum_{c=1}^{C} E\left[\underbrace{e^{-z^c H(v,c)}}_{\text{reweighting}} \underbrace{(z^c - h_m)^2}_{\text{squared error}}\right]$$

Gentle Boosting

Replacing the expectation with an empirical expectation over the training data, and defining weights $w_i^c = e^{-z_i^c H(v_i,c)}$ for example i and class c, this reduces to minimizing the weighted squared error:

$$\underbrace{J_{wse}}_{\text{Weight squared error over training data}} = \sum_{c=1}^{C} \sum_{i=1}^{N} \underbrace{w_i^c}_{\text{weight}} \underbrace{(z_i^c - h_m(v_i,c))^2}_{\text{squared error}}.$$

Good reference on boosting, and its different flavors

- See Friedman, J., Hastie, T. and Tibshirani, R. (Revised version) "Additive Logistic Regression: a Statistical View of Boosting" (http://www-stat.stanford.edu/~hastie/Papers/boost.ps) "We show that boosting fits an additive logistic regression model by stagewise optimization of a criterion very similar to the log-likelihood, and present likelihood based alternatives. We also propose a multi-logit boosting procedure which appears to have advantages over other methods proposed so far."

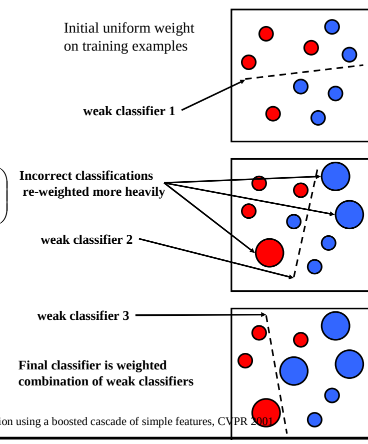
AdaBoost

(Freund & Shapire '95)

$$f(x) = \theta\left(\sum_t \alpha_t h_t(x)\right)$$

$$\alpha_t = 0.5 \log\left(\frac{error_t}{1 - error_t}\right)$$

$$w_t^i = \frac{w_{t-1}^i e^{-y_i \alpha_t h_t(x_i)}}{\sum_i w_{t-1}^i e^{-y_i \alpha_t h_t(x_i)}}$$



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

AdaBoost (Freund & Shapire 95)

- Given examples $(x_1, y_1), \ldots, (x_N, y_N)$ where $y_i = 0,1$ for negative and positive examples respectively.
- Initialize weights $w_{t=1,i} = 1/N$
- For t=1, …, T
  - Normalize the weights, $w_{t,i} = w_{t,i} / \sum_{j=1}^{N} w_{t,j}$
  - Find a weak learner, i.e. a hypothesis, $h_t(x)$ with weighted error less than .5
  - Calculate the error of h_t : $e_t = \sum w_{t,i} \,|\, h_t(x_i) - y_i \,|$
  - Update the weights: $w_{t,i} = w_{t,i} B_t^{(1-d_i)}$ where $B_t = e_t / (1- e_t)$ and $d_i = 0$ if example x_i is classified correctly, $d_i = 1$ otherwise.
- The final strong classifier is

$$h(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) > 0.5 \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$$

where $\alpha_t = \log(1/ B_t)$

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 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.)

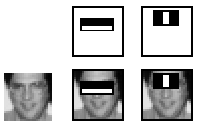
Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

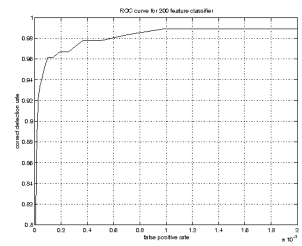
Example Classifier for Face Detection

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...



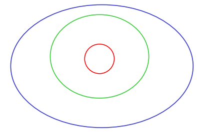


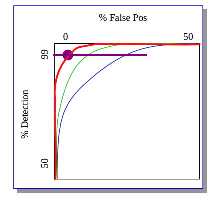
ROC curve for 200 feature classifier

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

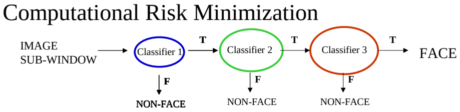
Trading Speed for Accuracy

- Given a nested set of classifier hypothesis classes



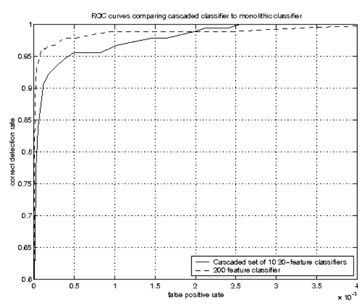


- Computational Risk Minimization



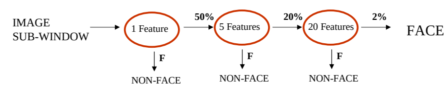
Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Experiment: Simple Cascaded Classifier



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 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 and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

A Real-time Face Detection System

Training faces: 4916 face images (24 x 24 pixels) plus vertical flips for a total of 9832 faces

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

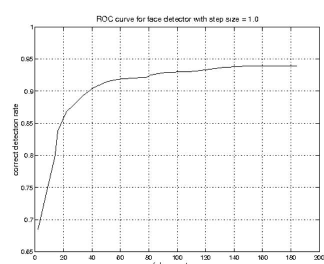
Final detector: 38 layer cascaded classifier
The number of features per layer was 1, 10, 25, 25, 50, 50, 50, 75, 100, …, 200, …

Final classifier contains 6061 features.

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Accuracy of Face Detector

Performance on MIT+CMU test set containing 130 images with 507 faces and about 75 million sub-windows.



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Comparison to Other Systems

Detector \ False Detections	10	31	50	65	78	95	110	167
Viola-Jones	76.1	88.4	91.4	92.0	92.1	92.9	93.1	93.9
Viola-Jones (voting)	81.1	89.7	92.1	93.1	93.1	93.2	93.7	93.7
Rowley-Baluja-Kanade	83.2	86.0				89.2		90.1
Schneiderman-Kanade				94.4				

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Speed of Face Detector

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

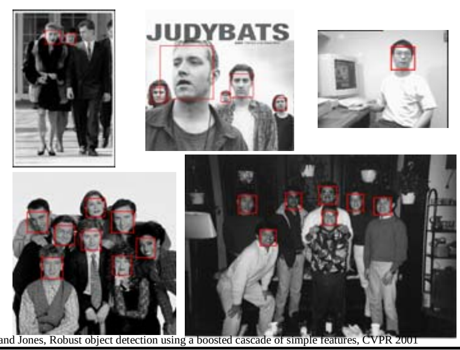
On the MIT+CMU test set, an average of 9 features out of a total of 6061 are computed per sub-window.

On a 700 Mhz Pentium III, a 384x288 pixel image takes about 0.067 seconds to process (15 fps).

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

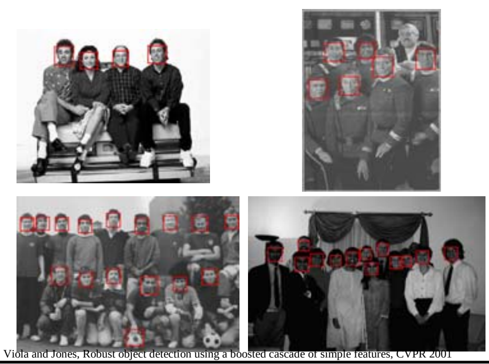
Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Output of Face Detector on Test Images



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

More Examples



Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Single frame from video demo

From Paul Viola's web page

We have created a new visual object detection framework that is capable of processing images extremely rapidly while achieving high detection rates. There are three key contributions.

The first is the introduction of a new image representation called the ``Integral Image'' which allows the features used by our detector to be computed very quickly.

The second is a learning algorithm, based on AdaBoost, which selects a small number of critical visual features and yields extremely efficient classifiers.

The third contribution is a method for combining classifiers in a ``cascade'' which allows background regions of the image to be quickly discarded while spending more computation on promising object-like regions.

A set of experiments in the domain of face detection are presented. The system yields face detection performace comparable to the best previous systems. Implemented on a conventional desktop, face detection proceeds at 15 frames per second.

Conclusions

- We [they] have developed the fastest known face detector for gray scale images
- Three contributions with broad applicability
  - Cascaded classifier yields rapid classification
  - AdaBoost as an extremely efficient feature selector
  - Rectangle Features + Integral Image can be used for rapid image analysis

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Today (April 7, 2005)

- Face detection
  - Subspace-based
  - Distribution-based
  - Neural-network based
  - Boosting based
- Face recognition, gender recognition

Some slides courtesy of: Baback Moghaddam, Trevor Darrell, Paul Viola

Bayesian Face Recognition

Moghaddam et al (1996)

Intrapersonal Ω_{I}

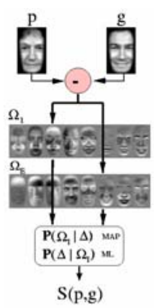
Extrapersonal Ω_{E}

$$\Omega_{\mathrm{I}} \equiv \{\Delta = x_i - x_j : L(x_i) = L(x_j)\}$$

$$\Omega_{\mathrm{E}} \equiv \{\Delta = x_i - x_j : L(x_i) \neq L(x_j)\}$$

$$\boxed{S = \frac{P(\Delta \mid \Omega_{\mathrm{I}})P(\Omega_{\mathrm{I}})}{P(\Delta \mid \Omega_{\mathrm{I}})P(\Omega_{\mathrm{I}}) + P(\Delta \mid \Omega_{\mathrm{E}})P(\Omega_{\mathrm{E}})}}$$

$P(\Delta \mid \Omega) \longrightarrow$ [Moghaddam ICCV'95]



Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000

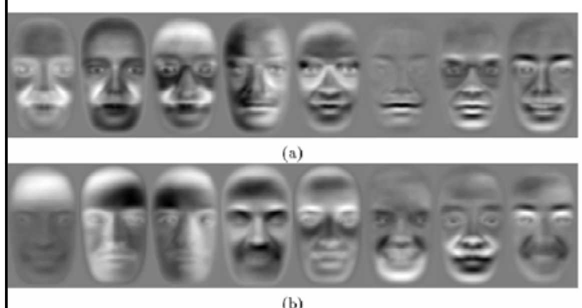


Figure 6: "Dual" Eigenfaces: (a) Intrapersonal, (b) Extrapersonal

Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000

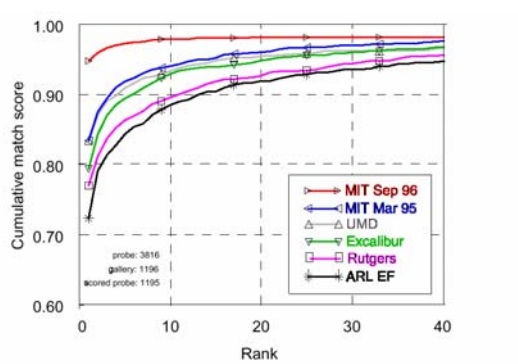
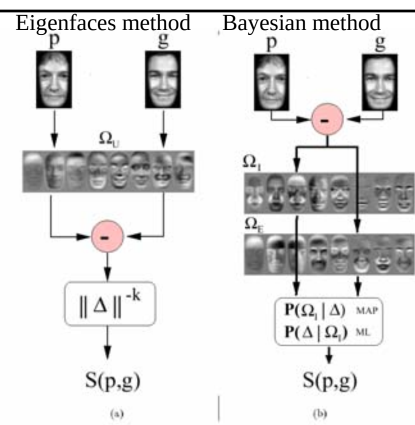


Figure 7: Cumulative recognition rates for frontal FA/FB views for the competing algorithms in the FERET 1996 test. The top curve (labeled "MIT Sep 96") corresponds to our Bayesian matching technique. Note that second placed is standard eigenface matching (labeled "MIT Mar 95").

Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000



Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000

Figure 9: Operational signal flow diagrams for (a) Eigenface similarity and (b) Probabilistic similarity.

Face Recognition Resources

Face Recognition Home Page:
- * http://www.cs.rug.nl/~peterkr/FACE/face.html

PAMI Special Issue on Face & Gesture (July ‘97)

FERET
- * http://www.dodcounterdrug.com/facialrecognition/Feret/feret.htm

Face-Recognition Vendor Test (FRVT 2000)
- * http://www.dodcounterdrug.com/facialrecognition/FRVT2000/frvt2000.htm

Biometrics Consortium
- * http://www.biometrics.org

Moghaddam, B.; Jebara, T.; Pentland, A., "Bayesian Face Recognition", *Pattern Recognition*, Vol 33, Issue 11, pps 1771-1782, November 2000

Gender Classification with Support Vector Machines

Baback Moghaddam

MITSUBISHI ELECTRIC RESEARCH LABORATORIES

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Support vector machines (SVM’s)

- The 3 good ideas of SVM’s

Good idea #1: Classify rather than model probability distributions.

- Advantages:
  - Focuses the computational resources on the task at hand.
- Disadvantages:
  - Don’t know how probable the classification is
  - Lose the probabilistic model for each object class; can’t draw samples from each object class.

Good idea #2: Wide margin classification

- For better generalization, you want to use the weakest function you can.
  - Remember polynomial fitting.
- There are fewer ways a wide-margin hyperplane classifier can split the data than an ordinary hyperplane classifier.

Too weak

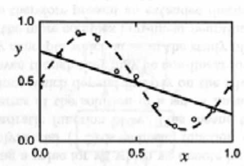


Figure 1.6. An example of a set of 11 data points obtained by sampling the function $h(x)$, defined by (1.4), at equal intervals of x and adding random noise. The dashed curve shows the function $h(x)$, while the solid curve shows the rather poor approximation obtained with a linear polynomial, corresponding to $M = 1$ in (1.2).

Bishop, neural networks for pattern recognition, 1995

Just right

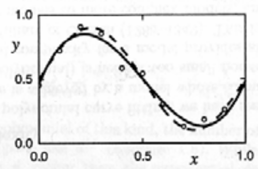


Figure 1.7. This shows the same data set as in Figure 1.6, but this time fitted by a cubic ($M = 3$) polynomial, showing the significantly improved approximation to $h(x)$ achieved by this more flexible function.

Bishop, neural networks for pattern recognition, 1995

Too strong

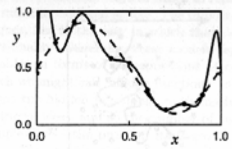


Figure 1.8. The result of fitting the same data set as in Figure 1.6 using a 10th-order ($M = 10$) polynomial. This gives a perfect fit to the training data, but at the expense of a function which has large oscillations, and which therefore gives a poorer representation of the generator function $h(x)$ than did the cubic polynomial of Figure 1.7.

Bishop, neural networks for pattern recognition, 1995

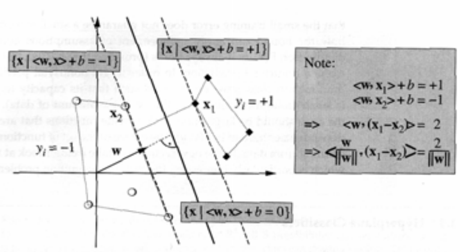


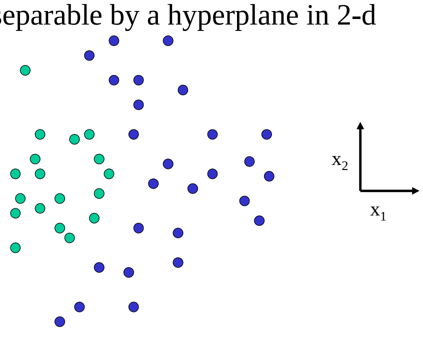
Figure 1.5 A binary classification toy problem: separate balls from diamonds. The *optimal hyperplane* (1.23) is shown as a solid line. The problem being separable, there exists a weight vector $\mathbf{w}$ and a threshold b such that $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) > 0$ ($i = 1, \ldots, m$). Rescaling $\mathbf{w}$ and b such that the point(s) closest to the hyperplane satisfy $|\langle \mathbf{w}, \mathbf{x}_i \rangle + b| = 1$, we obtain a *canonical* form $(\mathbf{w}, b)$ of the hyperplane, satisfying $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1$. Note that in this case, the *margin* (the distance of the closest point to the hyperplane) equals $1/\|\mathbf{w}\|$. This can be seen by considering two points $\mathbf{x}_1, \mathbf{x}_2$ on opposite sides of the margin, that is, $\langle \mathbf{w}, \mathbf{x}_1 \rangle + b = 1$, $\langle \mathbf{w}, \mathbf{x}_2 \rangle + b = -1$, and projecting them onto the hyperplane normal vector $\mathbf{w}/\|\mathbf{w}\|$.

Learning with Kernels, Scholkopf and Smola, 2002

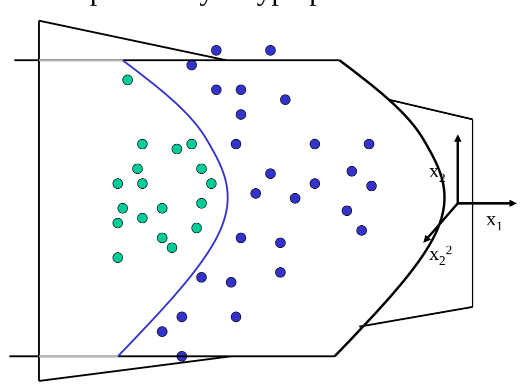
Finding the wide-margin separating hyperplane: a quadratic programming problem, involving inner products of data vectors

Good idea #3: The kernel trick

Non-separable by a hyperplane in 2-d



Separable by a hyperplane in 3-d



Embedding

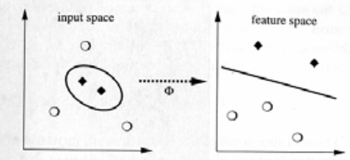


Figure 1.6 The idea of SVMs: map the training data into a higher-dimensional feature space via Φ, and construct a separating hyperplane with maximum margin there. This yields a nonlinear decision boundary in input space. By the use of a kernel function (1.2), it is possible to compute the separating hyperplane without explicitly carrying out the map into the feature space.

Learning with Kernels, Scholkopf and Smola, 2002

The idea

- There are many embeddings were the dot product in the high dimensional space is just the kernel function applied to the dot product in the low-dimensional space.
- For example:
  - $K(x,x') = (<x,x'> + 1)^d$
- Then you "forget" about the high dimensional embedding, and just play with different kernel functions.

Example kernel functions

- Polynomials
- Gaussians
- Sigmoids
- Radial basis functions
- Etc…

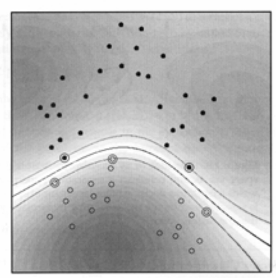
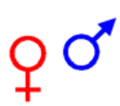


Figure 1.7 Example of an SV classifier found using a radial basis function kernel $k(x,x') = \exp(-\|x - x'\|^2)$ (here, the input space is $\mathcal{X} = [-1,1]^2$). Circles and disks are two classes of training examples; the middle line is the decision surface; the outer lines precisely meet the constraint (1.25). Note that the SVs found by the algorithm (marked by extra circles) are not centers of clusters, but examples which are critical for the given classification task. Gray values code $|\sum_{i=1}^{m} y_i \alpha_i k(x, x_i) + b|$, the modulus of the argument of the decision function (1.35). The top and the bottom lines indicate places where it takes the value 1 (from [471]).

Learning with Kernels, Scholkopf and Smola, 2002

Gender Classification with Support Vector Machines



Baback Moghaddam

MITSUBISHI ELECTRIC RESEARCH LABORATORIES

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Gender Prototypes

Images courtesy of University of St. Andrews Perception Laboratory

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

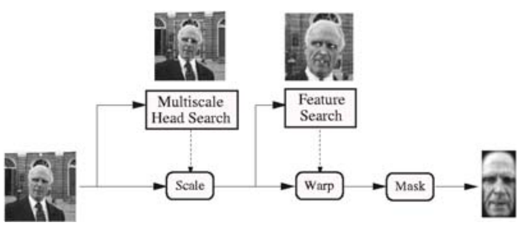
Gender Prototypes

Images courtesy of University of St. Andrews Perception Laboratory

Classifier Evaluation

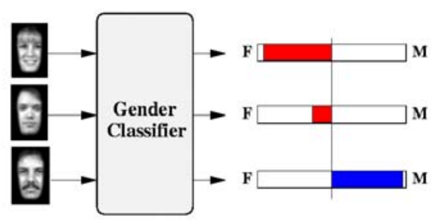
- Compare "standard" classifiers
- 1755 FERET faces
  - 80-by-40 full-resolution
  - 21-by-12 "thumbnails"
- 5-fold Cross-Validation testing
- Compare with human subjects

Face Processor

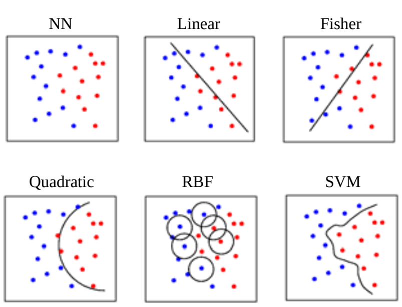


[Moghaddam & Pentland, PAMI-19:7]

Gender (Binary) Classifier



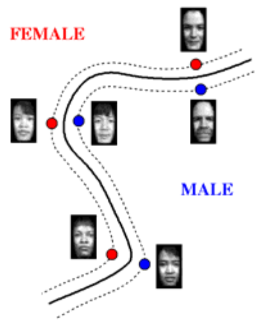
Binary Classifiers



Linear SVM Classifier

- Data: $\{\mathbf{x}_i , y_i\}$ $i = 1,2,3 \ldots N$ $y_i = \{-1,+1\}$
- Discriminant: $f(\mathbf{x}) = (\mathbf{w} \cdot \mathbf{x} + b) > 0$
- minimize $\|\mathbf{w}\|$
- subject to $y_i\,(\mathbf{w} \cdot \mathbf{x}_i + b) > 1$ for all i
- Solution: QP gives $\{\alpha_i\}$
- $\mathbf{w}_{opt} = \Sigma\, \alpha_i\, y_i\, \mathbf{x}_i$
- $f(\mathbf{x}) = \Sigma\, \alpha_i\, y_i\, (\mathbf{x}_i . \mathbf{x}) + b$

“Support Faces”



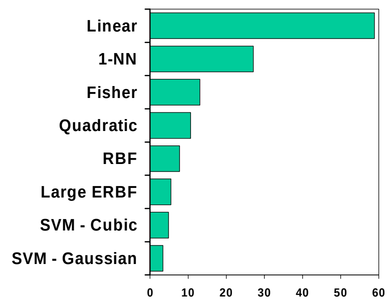
Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Classifier Performance

Classifier	Error Rate		
	Overall	Male	Female
SVM with RBF kernel	**3.38%**	**2.05%**	**4.79%**
SVM with cubic polynomial kernel	**4.88%**	**4.21%**	**5.59%**
Large Ensemble of RBF	5.54%	4.59%	6.55%
Classical RBF	7.79%	6.89%	8.75%
Quadratic classifier	10.63%	9.44%	11.88%
Fisher linear discriminant	13.03%	12.31%	13.78%
Nearest neighbor	27.16%	26.53%	28.04%
Linear classifier	58.95%	58.47%	59.45%

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Classifier Error Rates



Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Gender Perception Study

- **Mixture**: 22 males, 8 females
- **Age**: mid-20s to mid-40s
- **Stimuli**: 254 faces (randomized)
  - low-resolution 21-by-12
  - high-resolution 84-by-48
- **Task**: classify gender (M or F)
  - forced-choice
  - no time constraints

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

How would you classify these 5 faces?

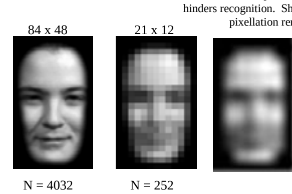
Top five human misclassifications

True classification: F, M, M, F, M

Human Performance

But note how the pixellated enlargement hinders recognition. Shown below with pixellation removed

Stimuli →



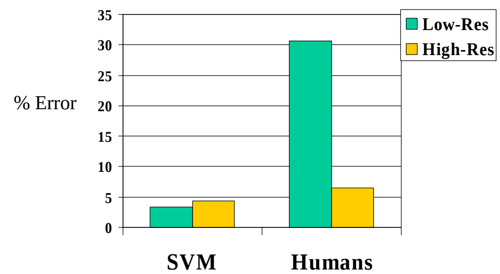
Results →

High-Res	Low-Res
6.54%	30.7%

σ = 3.7%

Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

Machine *vs.* Humans



Moghaddam, B.; Yang, M-H, "Learning Gender with Support Faces", *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, May 2002

end

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

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

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

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001

Ada-Boost Tutorial

- Given a Weak learning algorithm
  - Learner takes a training set and returns the best classifier from a weak concept space
    - required to have error < 50%
- Starting with a Training Set (initial weights 1/n)
  - Weak learning algorithm returns a classifier
  - Reweight the examples
    - Weight on correct examples is decreased
    - Weight on errors is decreased

$$\boxed{\sum_{i \in Errors} w_i = \sum_{j \in Correct} w_j}$$

- Final classifier is a weighted majority of Weak Classifiers
  - Weak classifiers with low error get larger weight

Viola and Jones, Robust object detection using a boosted cascade of simple features, CVPR 2001