MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

A Unified Learning Framework for Real Time Face Detection & Classification

Gregory Shakhnarovich *
Paul Viola †
Baback Moghaddam †

TR2002-23 May 2002

Abstract

This paper presents progress toward an integrated face detection and demographic analysis system that is robust and works in real-time. Faces are detected and extracted using the very fast algorithm recently proposed by Viola & Jones. Detected faces are passed to a novel demographics classifier which uses the same architecture as the face detector. This demographic classifier is extremely fast, yet delivers error rates slightly better than the best known classifiers. Demographics information, since it can be noisy in realistic situations, is integrated across time for each individual. The final demographic classification combines the estimates from many facial detections in order to significantly reduce error rate. The entire process runs faster than 10 fps on an 800 MHz Intel PIII.

International Conference on Automatic Face & Gesture Recognition (FG'02)
Washington D.C., May 2002

* MIT AI Laboratory
† MERL - Research Laboratory

Published in: *Int'l Conf. on Automatic Face & Gesture Recognition* (FG'02), May 2002.

A Unified Learning Framework for Real Time Face Detection and Classification

Gregory Shakhnarovich
AI Lab, MIT
gregory@ai.mit.edu

Paul A. Viola
MERL
viola@merl.com

Baback Moghaddam
MERL
baback@merl.com

Abstract

This paper presents progress toward an integrated, robust, real-time face detection and demographic analysis system. Faces are detected and extracted using the fast algorithm recently proposed by Viola and Jones [16]. Detected faces are passed to a demographics classifier which uses the same architecture as the face detector. This demographic classifier is extremely fast, and delivers error rates slightly better than the best known classifiers. To counter the unconstrained and noisy sensing environment, demographic information is integrated across time for each individual. Therefore, the final demographic classification combines estimates from many facial detections in order to reduce error rate. The entire system processes 10 frames per second on an 800 MHz Intel PIII.

Figure 1. Two detected faces and the associated gender (Male/Female) and ethnicity (Asian/Non-asian) estimates. Following the label is the classifi cation confi dence.

1 Introduction

Surveillance applications present an extreme challenge for the designers of vision algorithms. In many cases lighting is poor and cameras distant. Nevertheless, there is a clear need for systems which can record the arrival of people and divide them into demographic classes. This paper describes a system which automatically detects faces, tracks them across time, and classifies them as either male/female and asian/non-asian (see Figure 2 and 1). Each of the components of this system has been tested on a difficult dataset of faces from the world wide web. The overall system has been tested on video data recorded on a hand held video camera in a variety of office environments.

This unified system is designed to work online and in real-time. Faces are extracted using the algorithm described in [16], which operates at 15 frames per second. Detections are tracked across time using a technique reminiscent of particle filtering [7]. Detected faces are classified directly, without alignment or rectification, by an efficient demographics classifier.

The central contribution of this paper is a demographic classification scheme which is specifically designed to work in this real-time and real-world context. One key difference from previous work is the difficulty of the task, since faces are extracted from unconstrained video which can be of poor quality. Such faces are rarely completely frontal, are weakly aligned, and have extreme variations in lighting and image quality. The best published technique for gender classification is an SVM, which when tested on rectified FERET data yields an error rate of 3% [8, 10]. When trained and tested on a set of faces detected by our system, the SVM system yields an error rate of 23% and requires over 3000 support vectors (much of the increase can be attributed to the lack of rectification). In contrast our approach yields a classifier which attains 22% error and is approximately 1000 times faster to evaluate.

The structure of the demographics classifier is a perceptron constructed from simple localized features. Learning and feature selection is performed using a variant of Adaboost. This process automatically identifies regions of the face whose relative luminance characteristics provide infor-

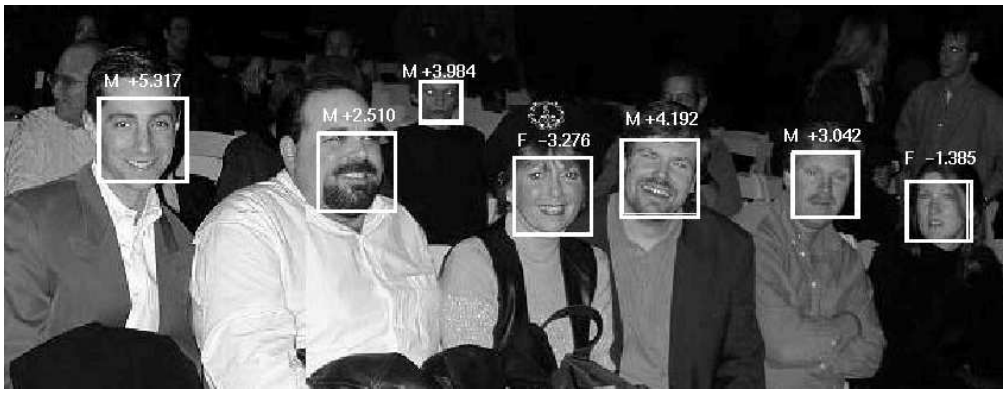

Figure 2. Seven examples of gender classification. Furthermore, all these faces were labeled as 'non-asian'. Observe the difficult conditions: non-frontal poses, in-plane rotations, and varied illumination, all encountered in this image.

mation for detection or classification. Unlike most existing systems we do not require full facial templates or explicit appearance models (AAMs) based on shape and texture.

A key distinguishing aspect of our work is the ability to perform analysis on *unaligned* faces. We forgo alignment for two reasons, robustness and efficiency. Alignment requires the automatic extraction of facial features, a process which is not entirely *robust* in the presence of significant pose, lighting, and quality variation. Alignment also requires significant time, since the features must first be found and then the face needs to be cropped, transformed, and resampled. To our knowledge there are few if any systems which knowingly attempt tasks such as gender or ethnicity classification on images with no precise alignment.

Another key aspect of this work is that both detection and classification are encompassed in one framework. In other words, the *same* architecture but *different* features are automatically derived for each task, without making any prior assumptions on what particular features should be used and the number (dimensionality) that is necessary.

1.1 Face Detection

Face detection has a long and rich history (see for example the survey on face detection by [19]). Since the same architecture is used both for face detection and demographic classification, key aspects of the technique are reviewed in Section 2.

Key competitors to the face detection approach of Viola and Jones include a neural network system by Rowley et. al. and a Bayesian system by Schneiderman and Kanade [11, 13]. While the neural network system is widely considered to be the fastest previous system, the Viola-Jones system is slightly more accurate and ten times faster. Though the Bayesian system has the highest reported detection rates, it is by far the slowest of the three.

One practical advantage of using the architecture proposed by Viola and Jones both for detection and classification, is that many pre-processing and bookkeeping calculations can be shared.

1.2 Gender Classification

In the early 1990s various neural network techniques were employed for classifying the gender of a (frontal) face. Gollomb, Lawrence and Sejnowski trained a fully connected two-layer neural network, SEXNET, to identify gender from 30-by-30 human face images [5]. Their experiments on a set of 90 photos (45 males and 45 females) show an average error rate of 8.1% compared to an average error rate of 11.6% from a study of five human subjects. Cottrell and Metcalfe also applied neural networks for emotion and gender recognition [3]. The dimensionality of a set of 160 64-by-64 face images is reduced from 4096 to 40 via an autoencoder network. These vectors are then given as inputs to another one layer network for training and recognition. Their experiments on gender classification report perfect results. Brunelli and Poggio [1] developed HyperBF networks for gender classification in which two competing RBF networks, one for male and the other one for female, are trained using 16 geometric features (e.g., pupil to eyebrow separation, eyebrow thickness, and nose width) as inputs. The results on a data set of 168 images (21 males and 21 females) show an average error rate of 21%. Similar to the methods by Golomb [5] and Cottrell [3], Tamura et al. [14] applied multilayer neural networks to classify gender from face images of multiple resolutions (from 32-by-32 to 16-by-16 and 8-by-8 pixels). Their experiments on 30 test images show that their network is able to determine gender from face images of 8-by-8 pixels with an average error rate of 7%. Instead of using a raster scan vector of gray levels to represent a face image, Wiskott et al. [18]

used labeled graphs of two-dimensional views to describe faces. The nodes are labeled with jets which is a special class of local templates computed on the basis of wavelet transform, and the edges are labeled with distance vectors similar to geometric features in [2]. They use a small set of controlled model graphs of males and females to encode the general face knowledge. It represents the face image space and is used to generate graphs of new faces by elastic graph matching. For each new face, a composite face resembling the original one is constructed using the nodes in the model graphs. If the majority of the nodes in the composite graph are taken from female models, it is believed the face image have the same gender. The error rate of their experiments on a gallery of 112 face images is 9.8%. Recently Gutta, Wechsler and Phillips [6] proposed a hybrid method which consists of ensemble of neural networks (RBFs) and inductive decision trees with Quinlan's C4.5 algorithm. Experimental results on a subset of FERET images of 384-by-256 (which were then manually located and normalized to 64-by-72 pixels) resulted in an average error rate 4% for gender classification.

In Moghaddam & Yang [8] 256-by-384 FERET "mugshots" were pre-processed and subsampled to 21-by-12 pixels for very low-resolution experiments. They used a total of 1,755 (1044 males and 711 females) FERET images with a 5-fold Cross Validation evaluation methodology. The best error rate reported was 3.4% using nonlinear Support Vector Machines.

1.3 Ethnicity Classification

There appears to be very little prior work in the area of facial ethnicity classification. Often it is simply an afterthought experiment conducted using one's representation for face recognition. There are certainly few if any statistically significant studies that have addressed this problem on the same scale as that of face recognition (ie. $O(10^3)$) individuals. One example is Gutta et al. [6] with the hybrid RBF/decision-trees. Using a similar architecture with Quinlan's C4.5 algorithm they were able to achieve a 6% error in ethnicity classification (consisting of four ethnic groups: Asian, Oriental, Caucasian, African).

Of course, categorization into these four groups is at times somewhat arbitrary and ambiguous (for example, from the paper the major distinctions between "Oriental" and "Asian" are not clear). Instead we consider a simpler and a somewhat more well-defined binary categorization into Asian and non-Asian.

2 Methodology

In recent work Viola and Jones have presented an efficient framework for face detection [16]. Three new insights where presented: (i) a new set of image features which are both efficient and effective for face analysis, (ii) a new feature selection algorithm based on Adaboost, and (iii) a cascaded architecture for learning and detection which accelerates performance significantly. In this paper we adopt both the image features and the Adaboost process for learning and feature selection.

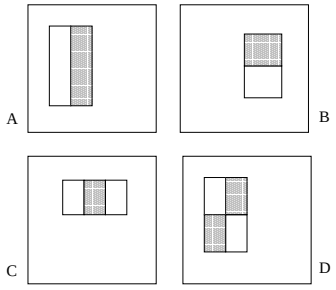

Figure 3. Example rectangle filters shown relative to the enclosing detection window. The sum of the pixels which lie within the white rectangles are subtracted from the sum of pixels in the gray rectangles.

2.1 Filters and Features

Following [16] image features are called *Rectangle Features* and are reminiscent of Harr basis functions (see [9] for the use of Harr basis functions in pedestrian detection). Each rectangle feature, $h_j()$ is binary threshold function constructed from a threshold θ_j and a *rectangle filter* $f_j()$ which is a linear function of the image:

$$h_j(x) = \begin{cases} 1 & \text{if } f_j(x) > \theta_j \\ 0 & \text{otherwise} \end{cases}$$

There are three types of rectangle filters. The value of a *two-rectangle filter* is the difference between the sum of the pixels within two rectangular regions. The regions have the same size and shape and are horizontally or vertically adjacent (see Figure 3 A and B). A *three-rectangle filter* computes the sum within two outside rectangles subtracted from the sum in a center rectangle (see C). Finally a *four-rectangle filter* computes the difference between diagonal pairs of rectangles (see D).

Given that the base resolution of the classifier is 24 by 24 pixels, the exhaustive set of rectangle filters is quite large, 190,800, which is roughly $O(N^4)$ (i.e. the number of possible locations times the number of possible sizes). The actual number is a smaller since filters must fit within the classification window. Note that unlike the Haar basis, the set of rectangle features is overcomplete[1].

[1] A complete basis has no linear dependence between basis elements

Computation of rectangle filters can be accelerated using an intermediate image representation called the integral image. Using this representation any rectangle filter, at any scale or location, can be evaluated in constant time.

2.2 The Boosted Classifier

The demographic classifier is a perceptron constructed from binary rectangle features. The number of distinct rectangle features is extraordinarily large; for each filter there are potentially many distinct thresholds each resulting in a different feature. With respect training error the set of distinct thresholds is bounded by the number of training examples, since all thresholds that result in the same dichotomy of the data are equivalent (i.e. given a sorting of the examples by filter value, it is clear that any threshold that lies between two consecutive examples is equivalent). Given 190,800 filters and 1,000 examples there are 190,800,000 distinct binary features.

Based purely on machine learning considerations the final classifier cannot include every distinct rectangle feature. A stronger constraint is the computationally efficiency of the final classifier, which leads us to limit the number of features to at most a few hundred or thousand. The challenge is therefore to select a small set of features upon which a computational efficient and low error classifier can be constructed.

While there are many published feature selection algorithms, none scales to this sort of problem (see chapter 8 of [17] for a review). In the domain of image database retrieval Tieu and Viola proposed the use of Adaboost to select 20 features from a set of 45,000 [4, 12, 15]. [16] refined the approach for the problem of face detection.

Though it is not widely appreciated, AdaBoost provides a principled and highly efficient mechanism for feature selection[4]. If the set of weak classifiers is simply the set of binary features (this is often called boosting stumps) each round of boosting adds a single feature to the set of current features.

In practice Adaboost is run for a few hundred rounds, yielding a perceptron The final classifier is given by

$$f(\mathbf{x}) = \sum_{t=1}^{T} \alpha_t h_t(\mathbf{x}) \qquad (1)$$

with a few hundred features. Since the process is greedily optimal, a trained classifier can be truncated to form a smaller more efficient classifier at any time (albeit with higher error rate).

In practice no single feature can perform the classification task with low error. Features which are selected in early rounds of the boosting process had error rates between 0.3 and 0.4. Features selected in later rounds, as the task becomes more difficult, yield error rates above 0.4.

and has the same number of elements as the image space, in this case $24 \times 24 = 576$. The full set of filters is many times over-complete.

2.3 Gender and Ethnicity Classifiers

For both the gender and a 2-class ethnicity classifier, an input facial image $\mathbf{x}$ generates a scalar output $f(\mathbf{x})$ whose polarity – sign of $f(\mathbf{x})$ – determines class membership. The magnitude $\|f(\mathbf{x})\|$ can usually be interpreted as a measure of belief or certainty in the decision made. Nearly all binary classifiers can be viewed in these terms; for density-based classifiers (Linear, Quadratic and Fisher) the output function $f(\mathbf{x})$ is a log likelihood ratio, whereas for kernel-based classifiers (Nearest-Neighbor, RBFs and SVMs) the output is a "potential field" related to the distance from the separating boundary.

In our system, the classifier output $f(\mathbf{x})$ is a linear combination of simple (and weak) single-feature discriminants. The final (binary) class assignment can consist of simple sign-thresholding of this output or more complicated temporal fusion algorithms constructed specifically for tracking in video sequences (see Section 2.5).

Note: the features selected for gender or ethnicity can be different from each other and are very different from those selected for face detection.

2.4 Tracking

The analysis of surveillance video presents different problems and opportunities than the static analysis of mugshots or passport photos. As mentioned earlier surveillance video is low quality, has poor lighting, and may never capture a completely "frontal" facial image. These flaws are somewhat offset by the large number of face images available for any individual. Given 15 frames per second, it is not unreasonable to assume that 15 to 30 usable images of each individual will be available. Though no single image might be of high quality, the collection of images is more likely to yield confident predictions of gender or race. This sort of temporal integration requires that detected faces be tracked across time.

In this section we will briefly describe a new form of face tracking which has distinct advantages over previous approaches.

Most tracking algorithms implicitly assume that a complete "brute force" search of an image for the target of interest is prohibitively expensive. As a result location evidence from the previous frames is used to focus the search in subsequent images (see [7] for an example). Given strong prior assumptions about the movement of the target, each new frame can be processed very quickly. Another key component of tracking algorithms is disambiguation: given the

locations of several targets in one image and their location in the next image, the tracking algorithm can act to disambiguate the identities of the objects. Once again a prior model for target motion is required. One form of ambiguity is the absence of evidence for a particular tracked target. Prior knowledge can be used to compute target location as a combination of weak image evidence and strong prior expectations. The final property common to almost all tracking algorithms is the necessity of initialization. One either assumes a simple process is sufficient for initial detection or initialization is done by hand.

In the domain of face tracking, the appearance of a real-time algorithm like that described in [16] can radically change many of the classical motivations for tracking. The tracking process no longer needs to "focus" the search for faces (though focusing can lead to an improvement in computation time). The tracking process no longer requires initialization, since new faces will be automatically detected in each frame. The remaining issue is disambiguation: the process by which detections in one image are related to detections in subsequent images.

Given an extremely fast detector, a variety of tracking schemes can be constructed. The simplest approach is "discrete". A target track is continued if a face is detected in a location which is assigned some minimum likelihood by the prior model of motion. If no face is detected for a given target track it is terminated. After extension of existing target tracks, all unaccounted faces are used to create new target tracks.

While this process works well it can be brittle. Since face detection rates are just over 90% it is not unusual to loose track of a fully visible face after 10 or 20 frames. The discrete approach lacks the ability to make up for the lack of evidence in any particular image.

A better approach uses a variant of deterministic particle filtering [7]). Each track generates a set of hypothesis locations in the next image. The evidence at each location is then combined to yield an estimate for the conditional distribution of target tracks in this new image.

Particle filtering style approaches are probabilistic: the evidence measured in the image is interpreted as the conditional probability of the image given a location hypothesis. Since the [16] cascaded detector is discriminative, there is no immediate analogy with probabilities. Our approach is to relate the depth reached in the cascade with the probability of the image. In other words the probability of a image patch is higher if it progresses further through the cascade. This is exactly consistent with the detection process.

A target track is terminated when no particle is assigned a probability above a rejection threshold (this probability is the product of the observation probability, the dynamical model, and the current likelihood). A new target track is initialized if a face is detected in a location assigned zero probability by all current tracks.

The resulting tracker is more robust than the face detector alone because it can continue to track a face even if it would not have been detected by the detection cascade.

The dynamical model used in our experiments is a first order linear with a decay constant of 0.9. The deterministic particle proposal distribution explores all locations/scales which have at least 35% overlap with the current location. The image evidence function assigns probabilities to images patches as e^{n-N}, where N is the total number of cascade levels, and n is the number of levels before the patch is rejected.

2.5 Temporal Integration for Classification

The ability to track an individual throughout a video sequence allows us to design fusion strategies for combining classifier outputs at each time frame in order accumulate evidence and thus form more confident and reliable decisions. It should be noted that in our system the computational load of classifying a single detected facial image is minor compared to the processing load of the face detection itself. Therefore, we can afford to make evaluate the demographics classifier $f(\mathbf{x})$ at each time frame t. A final decision statistic can be made using the following (causal) formula

$$D(t) \;=\; \frac{1}{T}\sum_{i=0}^{T} e^{-\alpha i}\, V(f(\mathbf{x}_{t-i}))\, Q(\mathbf{x}_{t-i}) \qquad (2)$$

This formula allows for a decision criterion $D(t)$ which is an exponentially weighted sum of the past T classifier outputs $f(\mathbf{x}_t)$ passed through a fixed "voting" function $V()$ and an input-dependent "quality" function $Q(\mathbf{x})$. The latter Q function measures the quality of the input irrespective of the confidence of the classifier ($||f(\mathbf{x}_t)||$) and may take into account factors such as overall input image contrast, lighting gradients or pose variation (degree of mis-alignment). The simplest integration scheme has no time decay $\alpha = 0$, a linear ramp $V(f) = f$ and no quality modulation $Q(\mathbf{x}) = 1$, corresponding to a simple running average of the classifier outputs $f(\mathbf{x}_t)$. The voting function, however, can be hard-limited (a "sign" function, for example) in which case the result is the average class membership vote in the last T frames. Setting $T = 0$ or equivalently $\alpha \rightarrow \inf$ would disable temporal fusion altogether, resulting in "snapshot" decisions in time with $D(t)$. We are currently in the process of investigating the effectiveness of this technique.

3 Experimental Results

In order to estimate the performance of the proposed framework, we collected a set of images of human faces

from the World Wide Web. The images were fetched using "crawling" software", and the faces were detected automatically. We then manually labeled the detections, removing false positives and faces more than $30\mathrm{deg}$ off frontal orientation, as well as those for which it was impossible to establish the ground truth regarding the gender and ethnicity of the subjects. The face detector was trained on a basic scale of size 24x24 pixels, to which size the test images were scaled as well. All the results reported in this section are obtained using 5-fold cross-validation. The images were randomly split to 5 subsets, with equal balance between the two classes, and in each trial one of the subsets was used as test set for a classifier trained on the rest of the data. In the end, the results were averaged over the 5 trials. The reported total error is the average of the errors on the two classes, assuming equal priors.

Naturally, the images are very diverse in their resolution, lighting conditions, age, gender and ethnicity of the subjects. To assess the degree of alignment in the data, we also marked the location of 5 landmarks (eyes, nose and mouth corners). Figure 4 shows the plot of the distribution of the marked eye centers. The standard deviation of the eye locations is about 20% of the inter-ocular distance and over 20% of the images were "poorly" aligned: the location of the eyes with respect to the detected face box was farther than two standard deviations from the mean eye location. This "poor" alignment as caused by non-frontal views, in-plane rotation, and poor face localization by the detector.

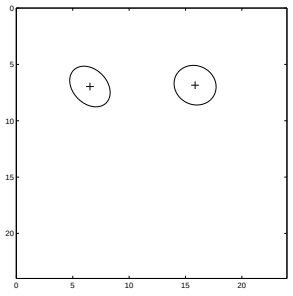

Figure 4. Illustration of the alignment properties of our Web images. The distribution of the eye centers are displayed as a mean and an ellipse at 1 standard deviation.

3.1 Classification Performance

3.1.1 Gender

The task in this experiment was to find the gender of the subject. Table 1 describes the results.

For comparison, we trained an SVM-based classifier with radial basis function kernel (kernel functions and parameters were manually tuned to obtain the best performance on this problem). SVMs have achieved the best published results of 3.5% error on good quality and properly aligned data. On this more realistic data SVMs yield an average error of 24.5%. Note: there is a notable bias towards males in classification (28% error on female faces).

Classifier	Female	Male	Total error
50 features	24.7%	24.3%	24.5%
100 features	23.7%	23.1%	23.4%
300 features	23.7%	21.2%	22.4%
500 features	21.9%	20.1%	21.0%
SVM, rbf kernel	28.1%	20.9%	24.5%

Table 1. Gender results - 5-fold CV

Classifier	Non-Asian	Asian	Total error
50 features	24.2%	24.4%	24.3%
100 features	22.7%	24.7%	23.7%
300 features	20.6%	22.2%	21.4%
500 features	19.7%	21.8%	20.8%
SVM rbf kernel	20.3%	24.9%	22.6%

Table 2. Ethnicity classification results - 5-fold CV

Though similar in error rate, SVM's and boosted feature classifiers are radically different in computation time. The learned SVMs often utilized over 2500 support vectors out of a total training set size of 3500 (each fold of cross-validation yielded a slightly different number of support vectors). Evaluating the classifier therefore involves the computation of 2500 image dot products. In contrast, the boosted classifier achieves better performance with 100 features – each requiring at most 10 additions and a thresholding operation. Moreover, the face detection preprocessing described in Section 2 and which is performed before applying the SVM as well, sets the stage for the demographic classifiers, so that no additional preprocessing is required. The resulting speedup factor is about 1000-fold.

3.1.2 Ethnicity

The task in this experiment was to classify the subject as an Asian or non-Asian[2]. Table 2 shows the performance of our classifier, compared to that of the SVM. While the test error of the boosted classifier continues to decrease after the first 300 features, the incremental value of additional features is low. Note that for all classifiers there is a significant bias towards one class (non-Asians). Again, the SVM found a very large number of support vectors (2300, out of 3250 training images in each fold of cross-validation).

[2] almost all the non-Asian faces in our data belong to Caucasians

3.2 Performance for Image Retrieval

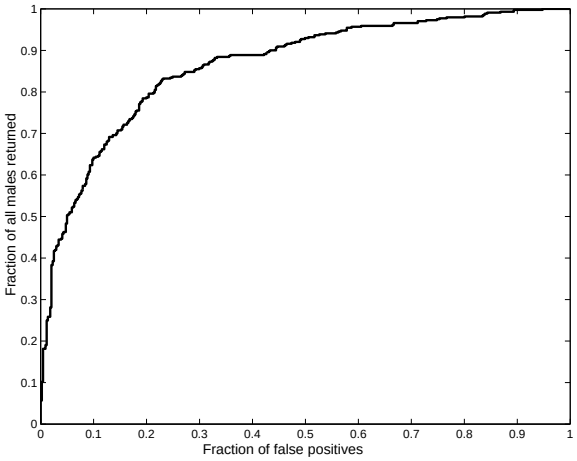

Figure 5. An ROC curve for the detection of males in a database of images. Vertically is the proportion of male images which are returned (returning 100% is the goal). Vertically is the percentage of females retrieved at that threshold (return 0% is the goal).

As noted in Section 2.3, our classifiers (1) provide a measure of confidence. In order to evaluate the usefulness of this measure as a decision threshold, we cast the male vs. female classification as a detection problem. Specifically, suppose we are interested in retrieving all images of males from a collection of photographs. Then by controlling the threshold on the value of $f(x)$ that is deemed high enough for classifying x as a male, we can control both the rejection and the false alarm fraction. The ROC curve for one such test is given in Figure 5. It was computed on one of the test sets from the cross-validation partition described in the section regarding gender experiments. Out of 882 images, half were of males, half of females.

3.3 Demographic Classification from Video

In a separate experiment, we evaluated the performance of the combined face detection and classification system on video tape containing 30 subjects. Each person appears in about 1 minute of video, filmed at 30 fps, and is captured speaking, with natural variations in pose and expression, under natural lighting conditions of indoor office environment. The tracking algorithm described in Section 2.4 was applied to the image sequences to locate and track faces. The demographic classifiers trained on the Web data described above were applied to subwindows where faces had been detected and tracked. The confidence of demographic classification (ethnicity and gender) is updated through the track, using the strategy described in 2.5.

The video of each subject was divided into three subsequence of 3 seconds. Each subsequence was independently tracked and labeled. This results in 90 experiments. In each video subsequence classification confidences were combined as a "running average" (see Section 2.5). The error rate of gender classification after temporal fusion was reduced to 10%. While the error rate of ethnicity classification was reduced to 17%. Note that unlike the web data, the total number of video participants was very small and was heavily biased toward non-asian males.

4 Discussion

The task of demographic classification on low quality surveillance video is a difficult one, and the error rates of 21% is far from perfect. In order to calibrate the difficulty of the task, a state of the art SVM-based classifier was trained on the same data. While error rates of the two classifiers were similar, with slight superiority of the boosted classifiers, computation times at the classification step were starkly different, with our classifiers running about 1000 times faster. Trained on 3500 images, the SVM used about 2500 support vectors, which means a three orders of magnitude advantage of the boosted classifiers that used 300 features, each requiring 8-10 additions and a thresholding. For any application, but especially for an integrated real-time system, the time it takes to classify a new image is of critical importance.

The low error rates shown in the anecdotal experiments in Figures 2 and 1 are not atypical. These images are actually of much higher quality than many examples from our web test set. In a number of applications where image quality is higher than web data, lower error rates can be assumed.

A real-time system provides an additional opportunity for reduction in classification error, the integration of classifications across time. Given many images of the same individual classifications can be averaged across time to provide a significant reduction in error rate.

References

[1] R. Brunelli and T. Poggio. Hyperbf networks for gender classification. In *Proceedings of the DARPA Image Understanding Workshop*, pages 311–314, 1992.

[2] R. Brunelli and T. Poggio. Face recognition : Features vs. templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10), October 1993.

[3] Garrison W. Cottrell. Empath: Face, emotion, and gender recognition using holons. In *Advances in Neural Information Processing Systems*, pages 564–571, 1991.

[4] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Com-*

putational Learning Theory: Eurocolt '95, pages 23–37. Springer-Verlag, 1995.

[5] B. A. Golomb, D. T. Lawrence, and T. J. Sejnowski. Sexnet: A neural network identifies sex from human faces. In *Advances in Neural Information Processing Systems*, pages 572–577, 1991.

[6] S. Gutta, H. Wechsler, and P. J. Phillips. Gender and ethnic classification. In *Proceedings of the IEEE International Automatic Face and Gesture Recognition*, pages 194–199, 1998.

[7] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. In *Int. J. Comp. Vis.*, volume 29, pages 5–28, 1998.

[8] B. Moghaddam and Ming-Hsuan Yang. Gender classification with support vector machines. In *Proc. of Int'l Conf. on Automatic Face and Gesture Recognition*, pages 306–311, Grenoble, France, March 2000.

[9] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *International Conference on Computer Vision*, 1998.

[10] P. J. Phillips, H. Moon, P. Rauss, and S. Rizvi. The FERET evaluation methodology for face-recognition algorithms. In *IEEE Proceedings of Computer Vision and Pattern Recognition*, pages 137–143, June 1997.

[11] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *IEEE Patt. Anal. Mach. Intell.*, volume 20, pages 22–38, 1998.

[12] Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. In *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997.

[13] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Computer Vision and Pattern Recognition*, 2000.

[14] S. Tamura, H. Kawai, and H. Mitsumoto. Male/female identification from 8×6 very low resolution face images by neural network. *Pattern Recognition*, 29(2):331–335, 1996.

[15] K. Tieu and P. Viola. Boosting image retrieval. In *International Conference on Computer Vision*, 2000.

[16] Paul Viola and Michael J. Jones. Robust real-time object detection. In *Proc. of IEEE Workshop on Statistical and Computational Theories of Vision*, 2001.

[17] J.A. Webb and J.K. Aggarwal. Shape and correspondance. *Computer Vision, Graphics, and Image Processing*, 21:145–160, 1983.

[18] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Face recognition and gender determination. In *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, pages 92–97, 1995.

[19] Ming-Hsuan Yang, David Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *Pattern Analysis & Machine Intelligence*, to appear.