Advances in Component-based Face Detection

Stanley M. Bileschi[1] and Bernd Heisele[2]

[1] Center for Biological and Computational Learning, M.I.T., Cambridge, MA, USA
[2] Honda R&D Americas, Inc., Boston, MA, USA
bileschi@ai.mit.edu heisele@ai.mit.edu

Abstract. We describe a component based face detection system trained only on positive examples. On the first layer, SVM classifiers detect predetermined rectangular portions of faces in gray scale images. On the second level, histogram based classifiers judge the pattern using only the positions of maximization of the first level classifiers. Novel aspects of our approach are: a) using selected parts of the positive pattern as negative training for component classifiers, b) The use of pair wise correlation between facial component positions to bias classifier outputs and achieve superior component localization.

1 Introduction

Object recognition is a well-studied area of computer vision. Face detection, the act of finding the set of faces in an image, is perhaps the most common of visual object detection tasks. In the following we give a brief overview of face detection methods.

In [8, 5] neural networks are used to discriminate between face and non-face images. In [4] Support Vector Machines (SVMs) using polynomial kernels are trained to detect frontal faces. The above systems have in common that the whole face pattern is represented by a single feature vector which is then fed to a classifier. This global approach works well for frontal faces but is sensitive to rotations in depth and to partial occlusions. More recently, component based classification methods have been developed to overcome these problems. A naïve Bayesian approach for face detection is presented in [6]. In this system, the empirical probabilities of occurrence of small rectangular intensity patterns within the face are determined. Another probabilistic component based approach is proposed in [3]. Here, the geometrical configuration of the parts is matched to a model configuration by a conditional search.

Most of the above systems are trained on a positive training set of face images and a negative training set of background patterns (non-face images). The negative class contains all possible non-face objects in images, making it difficult to represent it with a reasonably sized training set. We propose a method that circumvents this problem completely by training a face detection system on positive examples only. As starting point, we use the face detection system developed in [2]. It consists of a two level hierarchy of SVM classifiers. On the first level, a set of 14 component classifiers are shifted over every position of a novel input image. Each classifier's maximum output is then propagated to a second level SVM, which makes decisions whether or not the pattern is a face.

Instead of training the SVM component classifiers on face (i.e. nose, mouth etc) and non-face patterns, we train them on face patterns only. As negative examples we extract other parts of the face (i.e. not the nose, or not the mouth). Not only do we avoid dealing with background patterns, we also improve the detection accuracy of the component classifier given a face as input image: shifting a component classifier over a face image it is more likely to peak at the correct location when other nearby parts of the face were included in the negative training examples. The second level SVM classifier in [2] bases its decision on the maximum outputs of the component classifiers within predefined search regions. In contrast, we propose a second level classifier which only uses the position of the detected components, i.e. the position of the peaks of the component classifiers within a 58×58 sub window. For training we determine the empirical distribution of the position of each component in the training data and assume that the positions of the components in non-face images will be approximately uniformly distributed. In a first experiment we implement a naïve Bayesian classifier, assuming that the positions of the components are independent. In a second experiment we develop a classification scheme that takes pair wise position statistics into consideration.

2 Procedure

The discussion of the architecture of our face detection system begins with a description of the processes involved in the system's construction. An outline of the data flow when detecting in a novel image follows.

2.1 Construction of the Face Detector

The construction of our component based face detector consists of three major parts. First we must acquire training images of faces, and from them extract training sets, both positive and negative, for our 14 components. From this data we must also build histograms representing the expected positions of said components. Histograms of expected relative positions must also be recorded at this point. Finally, we train the component classifiers.

The training data consists of 2,646 synthetic images of faces at a resolution of 100 x 100 pixels per image. The images were rendered from 21 different 3D head models under 63 conditions of rotation and illumination, as described in [1]. Figure 1 exhibits 4 example training images from the set. Because these images are projections of a 3D head model, dense correspondence in the form of the x y positions of many face sentinels is also available, as in Figure 1 at right.

The 14 components of our system are the same as those defined in [2]. Each component is centered at one of the 25 sentinels. The component is defined as a rectangle around the sentinel including all pixels within a fixed distance up, down, to the left and right of the center pixel. For instance, the mouth component

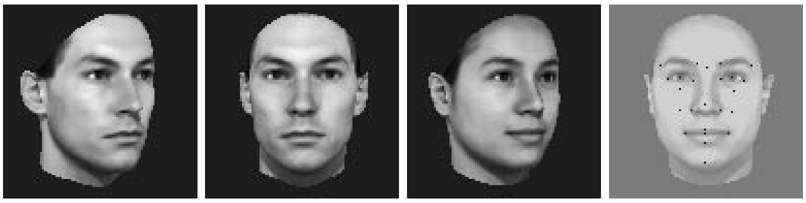

Fig. 1. Example images from the training data. The image on the far right has the 25 position sentinals visualized

includes the pixel defined as the center of the mouth, 15 pixels to the left, 15 pixels to the right, 7 pixels up, and 7 pixels down.

Given the face images, the positions of the sentinels, and the definitions of our 14 components, the extraction of the positive training set is simple. Each of the 2,646 images in the positive training set of a component is a cropped portion of a synthetic face image.

The negative training data for each component is extracted in a similar manner. Instead of cropping the portion defined to be the component, four random crops are taken from each face image. These cropped portions are of the same size as the positive component and are guaranteed by the extraction algorithm to overlap the defined position of the component by no more than 35% of the area. Thus, the negative training set of each component consists of 10,584 non-component patterns. These patterns are hereafter referred to as facial non-component patterns. Figure 2 depicts a typical mouth pattern, a typical facial non-mouth pattern, and also a non-face non-mouth pattern.

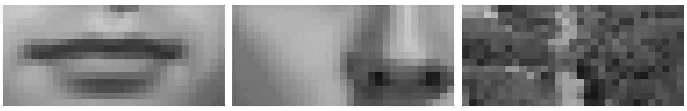

Fig. 2. Left: An example mouth pattern. Middle: An example non-mouth pattern extracted from the face. Right: An example non-mouth pattern extracted from a non-face

In order to construct our second level histogram classifier, we must construct position histograms of each classifier. The position histogram for component n is simply an image of the same scale as the synthetic input image, but whose pixel value at position i, j is representative of how many times the center of component n was at position i, j in the window. The 58×58 pixel image at the center of the histogram is then cropped out and the border discarded. This is done in order to more closely represent the part of the face we are interested in detecting, between the brow and chin but not including the hairline or much of the background. This 58×58 square becomes our definition of a face. In none of the training images do any of the sentinels fall outside this 58×58 window.

Figure 3 shows the position histogram for the mouth classifier. Darker areas are areas more likely to be the center of the mouth in a given training image.

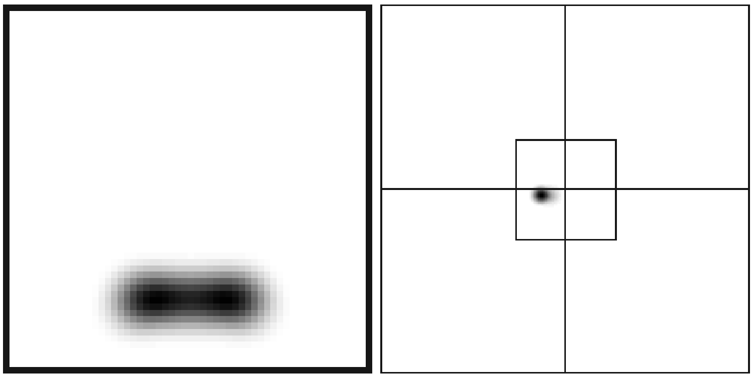

Fig. 3. Position histograms, darker areas are more likely to be the center of the component. Left: the expected position of the mouth component in any 58 x 58 window. Right: the expected position of the left eye given that the bridge of the nose is at the center of the crosshair. The inner square is 58 pixels in length

In our second experiment we will be using statistics of relative positions of pairs of components. During the training phase of the system, a pairwise position histogram is constructed for each pair of components. This histogram is a representation of where a component is likely to be centered given the center of another component. For instance, the left eye center is most likely approximately 15 pixels to the left and one pixel down from the bridge of the nose. Each first and second order position histogram is convolved with a small (5 pixel radius) blur in order account for slight rotations and variations in faces not well modeled by our training data. Also the histograms are linearly normalized before recording. The position histogram is normalized between one half and one, and the pair wise position histogram is normalized between the 13th root of one half and one. Figure 3 on the right shows a representation of the expected position of the left eye given the that the center of the bridge of the nose is mapped to the center of the crosshairs. For reference, the inner rectangle has a size of 58×58 pixels.

Finally, the component classifiers are trained using a linear kernel, and the construction of the system is complete. Before this training takes place however, each data point is histogram equalized in order to make the system more robust to variations in illumination.

2.2 Testing on a Novel Image

Given an image, which may or may not contain a face, one might ask how, given our 14 component classifiers and our position histograms, we detect faces. We begin the detection by compensating for the scale and position sensitivity via an exhaustive search through the test image at all positions and scales. This windowing method transforms our task to deciding face from non-face in each 58×58 pattern.

Given one rescaling of the test image, we check for 58×58 faces in the following way. First, a result image is created for each component. Figure 4 shows the result images from the brow, nose, mouth classifiers. Each result image is the same scale as the rescaled input image, but has pixel value i, j equal to the value returned when the rectangle centered at position i, j is fed through the component classifier. Once the result images have been computed, we move a 58×58 window through the rescaled test image, at each translation recording the positions of maximization of each result image inside the 58×58 sub-window. In this way, we are able to find the position of the best example of each component in each sub-window of our test image. We refer to this set of positions and values of maximization of the components as the constellation of a sub-window. Ideally, when the sub window is one exactly surrounding a face, the constellation will correspond to the positions of the 14 components of the face, as defined.

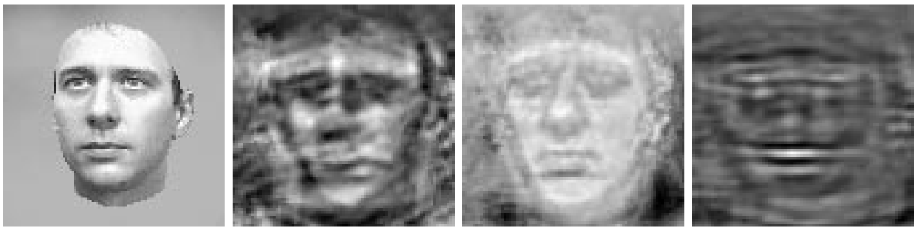

Fig. 4. Leftmost: A sample synthetic test image. Right: Three result images from the brow, nose, and mouth classifiers in order

In our first experiment, the value returned by the classifier for each sub window is the product of 14 values calculated by indexing into the position histograms. If classifier n maximizes at position x_n, y_n, then its contribution to the product will be the value in the position histogram stored at position x_n, y_n. Effectively, we are assuming that the position of each classifier is an independent random variable, and the probability of a given constellation, given that the image is a face, is simply the product of the probabilities of finding each component where the corresponding component classifier found it. Since we assume that the maximization of components in non face images is uniformly distributed, the probability of any constellation is equal, given the image is a non-face. This is analogous to a naïve Bayes formulation to this detection problem, to which the solution is always a likelihood ratio test. Since the likelihood of the pattern being a non-face is constant for all constellations, to decide face from non-face, given only the positions of maximization of our classifiers, we must only compare the product statistic to a threshold. To decide whether the entire image has a face in it or not, we only use the maximum product of all the 58×58 windows.

In the first experiment, the position of component i in a given sub-window was simply the maximum of result image i in that sub window. In our second experiment, after finding the positions of maximum stimulation of the classifiers, we employed the following greedy optimization rule to obtain a second estimate of the component position which utilizes both information from the image as well as information about the expected pair wise positioning of the components.

If classifier i maximizes at position x_i y_i, then for all result images n not equal to i we will multiply the result image n by the expected position of component n given the position of component i. This has the effect of biasing component n to maximize where it likely is in relation to component i. Ideally, if all the components are in their correct positions except one, then the 13 other components will all constructively bias the mistaken component at the point where we expect to find it. Since the minimum value returnable by the pair wise position histogram is the 13th root of one half, at worst any value in the result image will be cut in half. At best, if all the other components expect the component at the same location, the result image value will not change. After biasing the result image, the new maximum value of the result image in the sub window is recorded. Afterwards, given this new constellation, the computation continues as per experiment one. Figure 5 shows the same classifiers and the same data as Figure 4 except this time the results are subject to the greedy optimization rule.

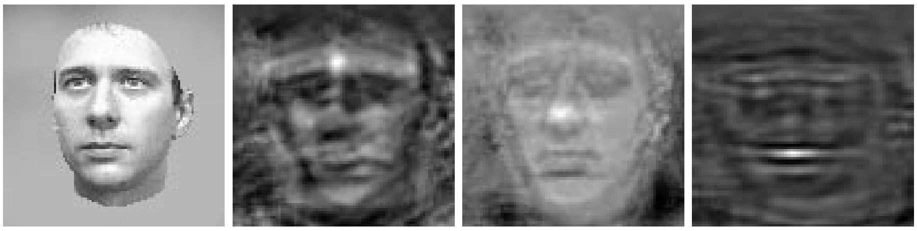

Fig. 5. Leftmost: A sample synthetic test image. Right: Three result images after greedy optimization (brow, nose, and mouth)

3 Results

In order to test the system, we reserved 1,536 artificial face images and 816 non face images. We chose to test our system on images of synthetic heads instead of real images of heads in order to save time. In the synthetic face images, we can be sure that the face is of the right size so we do not have search across scale. Also, with synthetic heads, we are able to extract the ground truth components in order to test the component classifiers individually and independent of the face detection task. The component vs. non-face non-component curves were trained on the same positive data, but used random extractions from 13,654 non-face images as the negative training set (vis. the rightmost image in Figure 2. Analogous to the classifiers defined in the procedure, the data for these classifiers were histogram equalized before training.

Each image, positive or negative, is first resized to 100 x 100 pixels. As per the discussion of experiment one in the procedure, the best result of all the 58 x 58 windows is the one that is recorded for the entire image. After every image has been evaluated we compute the ROC curve. Figure 6 shows the ROC curves for both the new system as discussed, and a similar system using component classifiers as trained in [2], vis. using non faces as the negative training data. Figure 7 also shows two ROC curves, this time comparing the result of our system both using and not using the pair wise position histograms to bias the output. Again, it is worth emphasizing that every test image, positive or negative, is handled identically by the system. Our final set of results, Figure 9 compares the results of our pair wise position utilizing system on synthetic and real data. The real data, 100 images from the CMU PIE face detection set [7], does not have the correspondence in position and scale that the synthetic data has. In order to compensate, our system was made to search for faces at 10 different scales between 60×60 pixels up to 100×100 pixels. Figure 8 is an example image from this test database.

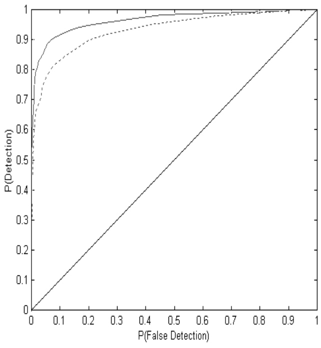

Fig. 6. A comparison of the complete system using classifiers trained on facial negatives (solid line) and non-facial negatives (dotted line)

4 Conclusions and Future Work

By using the remainder of the face as the negative training data for a component classifier, we aimed to engineer a classifier at least on par with the corresponding component classifier of [2]. We can say that the first level classifiers built in this

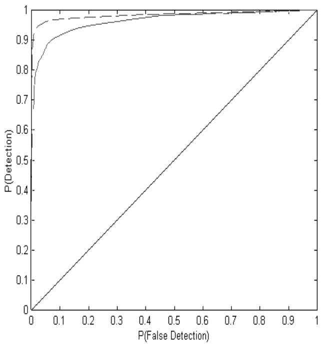

Fig. 7. A comparison of the complete system without (solid line) and with (dashed line) greedy optimization of position data

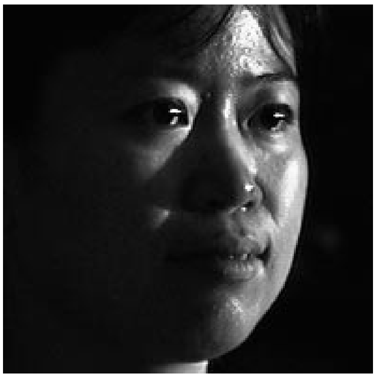

Fig. 8. An example face from the CMU PIE database

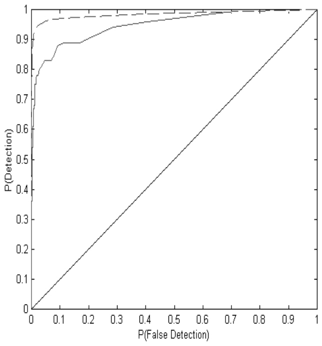

Fig. 9. A comparison of the complete system, using greedy optimization, on the synthetic test set (dashed line) and the CMU PIE face test set(solid line)

project are at least as powerful as those trained with the method outlined in [2], even with fewer training examples. Replacing the second level SVM constellation classifier with a classifier based on multiplying the outputs of histograms of positions, we were able to build a face detecting classifier using only faces as training data. Our system generalizes to images of real faces even though the training data was synthetic. Also we show that we can improve the accuracy of our system by utilizing information in pair wise position statistics.

As the system is currently designed, the data returned from each component is utilized equally. Empirical evidence suggests, however, that certain classifiers are more robust than others. We will overlay a weighting scheme on the processing of the constellations to give more weight to the components that have more credence.

References

1. V. Blanz and T. Vetter. A morphable model for synthesis of 3D faces. In *Computer Graphics Proceedings SIGGRAPH*, pages 187–194, Los Angeles, 1999.
2. B. Heisele, T. Serre, M. Pontil, and T. Poggio. Component-based face detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 657–662, Hawaii, 2001.
3. T. K. Leung, M. C. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *Proc. International Conference on Computer Vision*, pages 637–644, Cambridge, MA, 1995.
4. E. Osuna. *Support Vector Machines: Training and Applications.* PhD thesis, MIT, Department of Electrical Engineering and Computer Science, Cambridge, MA, 1998.
5. H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.
6. H. Schneiderman and T. Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 45–51, Santa Barbara, 1998.
7. T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression (PIE) database of human faces. Computer Science Technical Report 01-02, CMU, 2001.
8. K.-K. Sung. *Learning and Example Selection for Object and Pattern Recognition.* PhD thesis, MIT, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Cambridge, MA, 1996.