Face Detection using Independent Component Analysis

Aditya Rajgarhia
CS 229 Final Project Report

December 14, 2007

1 Introduction

A commonly used approach for detecting faces is based on the techniques of "boosting" and "cascading", which allow for real-time face detection. However, systems based on boosted cascades have been shown to suffer from low detection rates in the later stages of the cascade. Yet, such face detectors are preferable to other methods due to their extreme computational efficiency.

A given natural image typically contains many more background patterns than face patterns. In fact, the number of background patterns may be 1,000 to 100,000 times larger than the number of face patterns. This means that if one desires a high face detection rate, combined with a low number of false detections in an image, one needs a very specific classifier. Publications in the field often use the rough guideline that a classifier should yield a 90% detection rate, combined with a false-positive rate in the order of 10^{-6}.

In this project we introduce a novel variation of the boosting process that uses features extracted by Independent Component Analysis (ICA), which is a statistical technique that reveals the hidden factors that underlie sets of random variables or signals. The information describing a face may be contained in both linear as well as high-order dependencies among the image pixels. These high-order dependencies can be captured effectively by representation in ICA space [Barlow, 1989]. Moreover, it has been argued in [Bartlett and Movellan, 2002] that the metric induced by ICA is superior to other methods in the sense that it may provide a representation that is more robust to the effect of noise such as variations in lightening. We propose that features extracted from such a representation may be boosted better in the later stages of the cascade, thus leading to improved detection rates while maintaining comparable speed.

2 Robust Real-Time Face Detection

[Viola and Jones, 2001] described a face detection framework that is capable of processing images extremely rapidly while achieving high detection rates. There are three key contributions of this detection framework. The first is the introduction of a new image representation called the "Integral Image" which allows the features used by the detector to be computed very quickly. The second is a simple and efficient classifier which is built using the AdaBoost learning algorithm to select a small number of critical visual features from a very large set of potential features. 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 face-like regions.

2.1 Features

The detection procedure classifies images based on the value of simple features, as opposed to using the image pixels directly. The most common reason for doing so is that features can act to encode ad-hoc domain knowledge that is difficult to learn using a finite quantity of training data. For this system, there is also a second critical motivation for features: the feature-based system operates much faster than a pixel based system. The task is to find suitable features for detecting objects in images.

Viola and Jones use three kinds of features. The value of a *two-rectangle feature* is the difference between the sum of the pixels within the two rectangular regions. A *three-rectangle feature* computes the sum within two outside rectangles subtracted from the sum in a center rectangle. Finally, a *four-rectangle feature* computes the difference between diagonal pairs of rectangles.

2.2 Integral Image

Rectangle features can be computed very rapidly using an intermediate representation for the image that is called the integral image. The integral image at location x, y contains the sum of the pixels above and to the left of x, y inclusive:

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

where $ii(x, y)$ is the integral image and $i(x, y)$ is the original image. Using the following pair of recurrences:

$$s(x, y) = s(x, y - 1) + i(x, y)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y)$$

(where $s(x, y)$ is the cumulative row sum, $s(x, -1) = 0$, and $ii(-1, y) = 0$) the integral image can be computed in one pass over the original image. Using the integral image, any rectangular sum can be calculated in four array references. Clearly the difference between two rectangular sums can be calculated in eight references. Since the two-rectangle features defined above involve adjacent rectangular sums they

can be computed in six array references, and eight and nine references in the cases of three and four-rectangle features respectively.

2.3 Learning Classification Functions

There are 160,000 rectangle features associated with each image sub-window of 24 x 24 pixels, a number far larger than the number of pixels. Even though each feature can be computed efficiently, computing the complete set is prohibitively expensive. The hypothesis is that a very small number of these features can be combined to form an effective classifier. The main challenge, then, is to find these features. In this system, a variant of AdaBoost is used to select the features and to train the classifier. The formal guarantees provided by the AdaBoost learning procedure are quite strong. It has been proved in [Freund and Schapire, 1996] that the training error of the strong classifier approaches zero exponentially in the number of rounds. More importantly, a number of results were later proved about generalization performance.

Drawing an analogy between weak classifiers and features, AdaBoost is an effective procedure for searching out a small number of good "features" which nevertheless have significant variety. In support of this goal, the weak learning algorithm is designed to select the single rectangle feature which best separates the positive and negative examples. For each feature, the weak learner describes the optimal threshold classification function, such that the minimum number of examples are misclassified. A weak classifier $h(x, f, p, \theta)$ thus consists of a feature (f), a 24 x 24 pixel sub-window of the image (x), a threshold (θ) and a polarity (p) indicating the direction of the inequality:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } pf(x) < p\theta \\ 0 & \text{otherwise} \end{cases}$$

The weak classifiers used (thresholded single features) can thus be viewed as single node decision trees, and the final strong classifier takes the form of a perceptron (a weighted combination of weak classifiers followed by a threshold).

2.4 The Attentional Cascade

A cascade of classifiers is used, which achieves increased detection performance while radically reducing computation time. Simpler classifiers are used to reject the majority of sub-windows before more complex classifiers are called upon to achieve low false positive rates.

Stages in the cascade are constructed by training classifiers using AdaBoost. The overall form of the detection process is that of a degenerate decision tree, or cascade. A positive result from the first classifier triggers the evaluation of a second classifier which has also been adjusted to achieve very high detection rates. A positive result from the second classifier triggers a third classifier, and so on. A negative outcome at any point leads to immediate rejection of the sub-window. The structure of the cascade reflects the fact that within any single image, an overwhelming majority of sub-windows are negative.

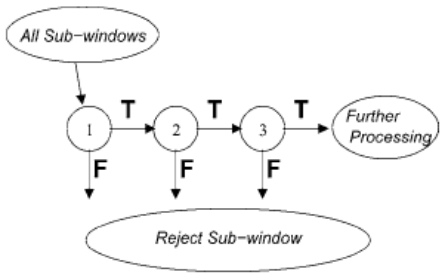

Figure 1: Schematic depiction of the attention cascade.

3 Boosting in ICA Feature Space

In Sec. 2.3, we described the boosting process in Haar-like feature space. The classification power of the described system is limited when the weak classifiers derived from simple local features become too weak to be boosted, especially in the later stages of the cascade training. Empirically, it has been observed in [Zhang, Li and Gatica-Perez, 2004] that when the discriminating power of a strong classifier reaches a certain point, e.g. a detection rate of 90% and a false alarm rate of 10^{-6}, non-face examples become very similar to the face examples in terms of the Haar-like features. The histograms of the face and non-face examples for any feature can barely be differentiated, and the empirical probability of misclassification for the weak classifiers approaches 50%. At this stage, boosting becomes ineffective because the weak learners are too weak to be boosted. This issue has been discussed in the past in [Valiant, 1984]. One way to address this problem is to use better weaker classifiers in a different feature space, which is more powerful. We propose to boost in ICA coefficient space. As we show, weak classifiers in this global feature space have sufficient classification power for boosting to be effective in the later stages of the cascade.

First, we shall explicate the two architectures for performing ICA for representing faces.

3.1 Architecture 1: Statistically Independent Basis Images

The goal here is to find a set of statistically independent basis images. We organize the face mixtures in matrix $\mathbf{x}$ so that the images are in rows and the pixels are in columns. In this approach, ICA finds a matrix W such that the rows of $\mathbf{u} = \mathbf{W}\mathbf{x}$ are as statistically independent as possible. The source images estimated by the rows of $\mathbf{u}$ are then used as basis images to represent faces. Face image representation consists of the coordinates of these images with respect to the image basis defined by the rows of $\mathbf{u}$, as shown in Fig. 3. These coordinates are contained in the mixing matrix $\mathbf{A} = \mathbf{W}^{-1}$.

The number of IC's found by the FastICA algorithm (describe in [Hyvarinen, 1999]) corresponds to the dimensionality of the input. In order to have control over the number

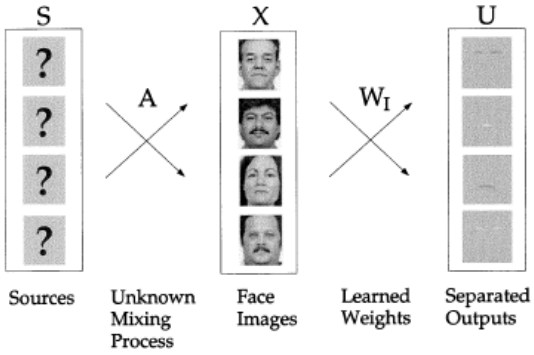

Figure 2: Image synthesis model for Architecture I.

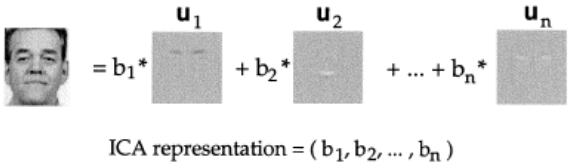

Figure 3: The independent basis image representation consists of the coefficients, $\mathbf{b}$, for the linear combination of independent basis images, $\mathbf{u}$, that comprised each face image $\mathbf{x}$.

of ICs extracted, instead of performing ICA directly on the n_r original images, we perform ICA on first m PC eigenvectors of the images set, where $m < n_r$. Recall that the ICA model assumes that the images in $\mathbf{x}$ are a linear combination of a set of unknown statistically independent sources. Thus, the ICA model is unaffected by replacing the original images with a linear combination of those images.

3.2 Architecture II: A Factorial Face Code

The goal in Architecture 1 was to find a set of spatially independent basis images. Now, although the basis images obtained in that architecture are approximately independent, the coefficients that code each feature are not necessarily independent. Architecture II uses ICA to find a representation in which the coefficients used to code images are statistically independent, i.e., a factorial face code. [Barlow, 1989] and [Atick, 1992] have discussed the advantages of factorial

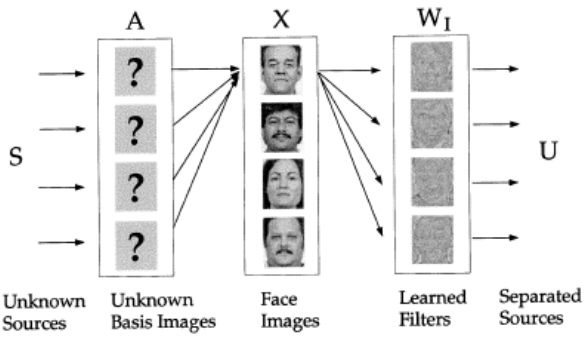

Figure 4: Image synthesis model for Architecture II.

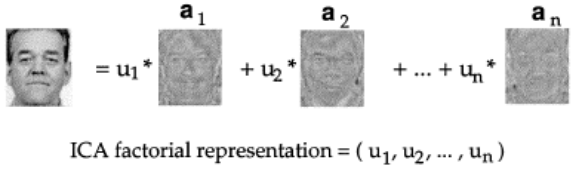

Figure 5: The factorial code representation consisted of the independent coefficients, $\mathbf{u}$, for the linear combination of basis images in $\mathbf{A}$ that comprised each face image $\mathbf{x}$.

codes for encoding complex objects that are characterized by high-order combinations of features.

We organize the data matrix $\mathbf{x}$ such that rows represent different pixels and columns represent different images. This corresponds to treating the columns of $\mathbf{A} = \mathbf{W}^{-1}$ as a set of basis images (see Fig. 4). The ICA representations are then in the columns of $\mathbf{u} = \mathbf{W}\mathbf{x}$. Each column of $\mathbf{u}$ contains the coefficients of the basis images in $\mathbf{A}$ for reconstructing each image in $\mathbf{x}$ (see Fig. 5). ICA attempts to make the outputs, $\mathbf{u}$, as independent as possible.

4 Boosting ICA Features

In AdaBoost learning, each weak classifier is constructed based on the histogram of a single feature derived from ICA coefficients $(b_1, b_2, ..., b_m)$. At each round of boosting, one ICA coefficient, the one which is most effective for discriminating between face and non-face classes, is selected by AdaBoost.

As stated earlier, the distributions of the two classes in the Haar-like feature space almost completely overlap in the later stages of the cascade training. In that case, we propose to switch feature spaces and construct weak features in the ICA space. We do need to address the question of which stage in the cascade we should switch from the Haar-like features to the ICA features. It is quite evident that ICA features are much more computationally expensive than Haar-like features. Now, if we used ICA features in early stages of boosting, we would have to extract ICA features from a very large number of sub-windows, and the speed of the face detection system would be too slow for real-time performance. On the other hand, if we used ICA features in very late stages of boosting, the performance improvement gained from their superiority would be limited. Therefore, we shall determine the switching stage based on the trade off between speed and performance improvement.

5 Experimental Results

First, we provide the implementation details for our system. The discussion includes details on the structure and training of the detector, as well as results on large real-world testing sets. We also consider the importance the size and quality of the training data set towards creating an accurate classifier, and present results for two training sets of different sizes. Due to time limitations, we were unable to train a cascade of

Figure 6: Example face images from the training set.

classifiers. However, we did implement separate AdaBoost classifiers based on Haar and ICA features.

5.1 Training Datasets

The training data set we used is the the publicly available MIT-CBCL face database. This data set is not ideal for the purpose of training a classifier due a low resolution of 19 x 19 pixels. In fact, [Viola and Jones, 2001] report that an increased resolution of 24 x 24 pixels results in much higher accuracy of the face detector. However, the data set will serve our purpose of comparing our detection system with their original system, which we shall train using the same training set.

The original MIT-CBCL training set contains 2,429 face images and 4,548 non-face images in 19 x 19 grayscale PGM format images. The training faces are only roughly aligned, i.e., they were cropped manually around each face just above the eyebrows and about half-way between the mouth and the chin.

We also created an extended version of the MIT-CBCL data set by randomly mirroring, rotating, translating and scaling the original images by small amounts to obtain a set of 17,495 faces and 113,939 non-face images. Although the additional images are just variants of the original ones, the performance of the classifier is affected significantly, as shown subsequently.

All face and non-face images in the training set were histogram equalized to increase the local contrasts of the images. This allows for areas of lower local contrast to gain a higher contrast without affecting the global contrast.

5.2 ICA and Haar Features

Two face detection systems were trained: One using Haar features (we call this *H-Boost*) and the other using Architecture I ICA features (we call this *I-Boost*). We trained both types of classifiers with several different numbers of features ranging from 50 features to 350 features. In the following sections, we shall present the results for H-Boost and I-Boost classifiers trained using 200 Haar-like and ICA features respectively, since this choice resulted in the highest detection rates.

For training the I-Boost system, we first extracted the ICA features from the 2,429 face images in the MIT-CBCL training set. Next, all the 2,429 face images and the 4,548 non-face images from the training set were projected onto the set of ICA features to obtain the ICA coefficients of these images. AdaBoost was performed on the coefficients of these 6,977 training images to produce the strong classifier. The

Figure 7: Example face images from the testing set.

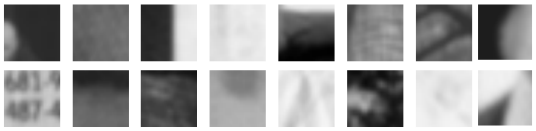

Figure 8: Example non-face images from the testing set.

extended training set of 17,495 faces and 113,939 non-faces was similarly projected onto the ICA basis extracted from the 2,249 face images to produce another strong classifier.

While experimenting with different numbers of faces from which we extract the ICA features, we found that larger numbers of faces result in better performance of the detector. However, extracting the independent components is a very memory-intensive task, and our memory limitations did not allow us to use more than 5,000 images. In the future, we would like to use the 17,495 face images from the extended training set to extract the ICA features as opposed to just projecting them onto the basis extracted using less features.

During testing, a given image is similarly projected on the above-mentioned ICA features to obtain the ICA coefficients for that image. The AdaBoost classifier then uses these coefficients to predict the class of the test image.

For training the H-Boost system, we first created the integral image representation for the training set, and then performed AdaBoost on the Haar-like features that are obtained using this integral image. We were unable to train the Haar classifier on the extended data set due to memory limitations.

5.3 Experiments on Real-World Test Sets

A number of experiments were performed to evaluate the system. We tested our system on the MIT-CBCL face test set, which consists of 472 faces and 23,573 non-faces. The testing images are of the same size as the training images, and are also cropped similarly. Considerable pose and lightning variations are represented by the test set, as can be seen in Fig. 8. The test face images are clearly more challenging to identify as compared to the training ones seen in Fig. 6, even for a human.

Fig. 9 shows the performance of our detection system (I-Boost) as well as that of a detector based on Haar-like features (H-Boost). Note that the H-Boost detector used is not the same as the Viola-Jones detector, since it is not cascaded. Clearly, the I-Boost detector performs better than H-Boost for all false positive rates. Moreover, using the extended training set significantly improves the accuracy.

	False Positive Rate (%)				
Detector	1.00	2.00	3.00	5.00	10.00
I-Boost (original training set)	3.0 %	16.7 %	28.6 %	38.6 %	50.4 %
I-Boost (extended training set)	12.7 %	26.5 %	38.6 %	62.7 %	75.0 %
H-Boost (original training set)	2.0 %	5.7 %	9.1 %	14.8 %	29.2 %

Figure 9: Detection rates for various numbers of false positives on the MIT-CBCL test set containing 472 faces and 23,573 non-faces.

6 Conclusions

In this project we introduced a novel algorithm for detecting faces, based on features derived from Independent Component Analysis. Motivated by the fact that the weak learners based on the simple Haar-like features are too weak in the later stages of the cascade, we propose to boost ICA features in the later stages. The global ICA feature space complements the local Haar-like feature space. The algorithm selects the most effective features from ICA features using AdaBoost.

Various experiments were performed to show the advantage of using ICA features for face detection. The results can be stated as follows:

- ICA features are better at discriminating between face and non-face images as compared to Haar-like features.
- Increasing the size of the training set as well as the size of images for ICA feature extraction significantly improves the detection rate for a given false positive rate.

Although we have not yet implemented the cascaded detector, the results from the AdaBoost classifier show that our system achieves high accuracy on the MIT-CBCL test set. Most importantly, though, we have showed that ICA features are, in fact, better than Haar-like features at discriminating between faces and non-faces. Hence, we are optimistic that a cascaded detections system which combines Haar-like and ICA features would demonstrate higher accuracy than a detector based only on Haar-like features. The computational efficiency of FastICA, coupled with the fact that the majority of images are rejected in the early stages of the cascade, should ensure that performance is not affected ostensibly.

7 Future Work

A larger training set would be essential for the detector to be of practical use. In particular, the number of non-face images would have to be drastically increased in order to decrease false positives. Moreover, as mentioned earlier, using a larger number of face images to extract ICA features would also improve the accuracy.

Implementing the cascade is required in order to achieve the ultimate aim of our work, i.e., to improve the accuracy of the Viola-Jones detector while maintaining real-time detection speed. We would also like to compare our system with other state-of-the-art detection systems such as those based on Neural Networks and Support Vector Machines.

It was mentioned in Sec. 5.2 that we have used the Architecture I ICA features in the I-Boost classifier. Another task in the future would be to implement Architecture II features as described in Sec. 3.2 and to compare the results.

References

P. Viola and M. Jones. Robust Real-time Object Detection. In *International Journal of Computer Vision*, pages 137-154, 2001.

H.B. Barlow. Unsupervised learning. In *Neural Computation*, page 295-311, 1989.

M.S. Bartlett, J.R. Movellan, T.J. Sejnowski. Face Recognition by Independent Component Analysis. In *IEEE Trans. on Neural Networks*, pages 1450-1464, November 2002.

Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning: Proceedings of the Thirteenth International Conference*, pages 148-156, 1996.

A. Hyvarinen. Fast and Robust Fixed-Point Algorithms for Independent Component Analysis. In *IEEE Transactions on Neural Networks*, pages 626-634, 1999.

D. Zhang, S. Z. Li, and D. Gatica-Perez. Real-Time Face Detection Using Boosting Learning in Hierarchical Feature Spaces. In *Proceedings of International Conference on Pattern Recognition*, Cambridge, August 2004.

L. Valiant. A Theory of the Learnable. In *Communications of ACM*, 1984.

J.J. Atick. Could information theory provide an ecological theory of sensory processing? In *Network*, page 213-251, 1992.