2008 IEEE International Conference on Signal Image Technology and Internet Based Systems

Applying Two-Pixel Features to Face Detection

Ido Nissenboim
Computer Science Department
University of Haifa
Haifa 31905, Israel
idoniss@gmail.com

Daniel Keren
Computer Science Department
University of Haifa
Haifa 31905, Israel
dkeren@cs.haifa.ac.il

Michael Werman
The Institute of Computer Science
The Hebrew University of Jerusalem
Jerusalem 91904, Israel
werman@cs.huji.ac.il

Abstract

We propose a "quick rejection" paradigm which is applied to face detection. The features we use are arguably the simplest possible: a threshold on the difference between the grey levels of two pixels. No negative examples are used for training; instead, we use a simple statistical model of natural images. The resulting features are easy to find, extremely fast to apply, and achieve a good detection rate.

1. Introduction and Previous Work

Recent years have witnessed a growing interest in "fast rejection" schemes, which very quickly reject the large majority of input images, and spend more time only on those inputs which survived the initial rejection. For example, if we seek to detect faces, we can make use of the fact that most inputs are "far" from resembling a human face, and very quickly reject them, spending more processing time only on images which are viable candidates to be a face. Some papers in this direction are [1, 2, 3, 4, 5, 6, 7, 8, 9].

The question suggests itself – just *how* fast can we make this fast rejection stage? Most algorithms require time which is proportional to the size of the image, and can usually be implemented by convolution if the task is to search for a face inside a large image. Typically, a *cascade* of "rejectors" is applied; if the image makes it through the first test, it is subjected to a second test, and if not, it is discarded. Those which make it through the second rejector are tested by a third rejector, etc. A typical rejection stage may consist of testing whether the inner product of the image with a certain "detector" is larger than a certain threshold [6].

In an effort to reduce running time by using ("touching") only a part of the image, the well-known algorithm in [5] tests only a few regions of the image, which capture the uniqueness of the grey-level structure of faces vs. background images. These features are combined using boosting to form a very fast and reliable detection scheme.

In [9], the number of pixels that are used is further reduced by searching for simple and local properties which characterize face images. These properties are expressed in terms of the mean and variance of local regions.

In [10], a very fast template matching algorithm using SPRT (Sequential Probability Ratio Test) is presented, however it does not seek to detect generic face images, but a specific template.

Here we attempt to further reduce the number of pixels which participate in the process, and to develop very simple features for detection. The main novelty lies in the application of a prior on the class of background images, which allows to quickly find features which reject the very large majority of background images.

Next we define the features and explain how they are derived and used. We shall hereafter speak of face detection, but the method can be applied to other image classes as well.

978-0-7695-3493-0/08 $25.00 © 2008 IEEE
DOI 10.1109/SITIS.2008.28

2. The Features

The approach described in this paper is very simple. We search for binary features, which for each image assume a value of "yes" (1) or "no" (0), and which satisfy the following properties:

1. Simplicity: the features should be as simple as possible to compute, involving simple operations on a minimal number of pixels.

2. Discrimination: each feature should obtain a value of "yes" with a much higher probability on face images than on background images.

3. Independence: since it is improbable that a single simple feature can successfully discriminate between faces and non-faces, we suggest to use a few features. For this to be effective and fast, these features have to be as independent as possible *over the set of background images*, since this will maximize the effectiveness of the rejection process.

We now elaborate on requirements 1-3 above.

2.1. Feature Simplicity

The features we use resemble those used in [5] and [9], but they are simpler – arguably, they are the simplest features that can be used, especially if we assume even a constant variation in the illumination. For every pair of pixel positions p_1, p_2 and threshold T, the feature $\{p_1, p_2, T\}$ obtains the value "1" for an image I iff $I(p_1) - I(p_2) \geq T$.

The intuition behind choosing such features (beyond the obvious one – their very simple form) is that they are immune to a constant change in illumination. They can also be thought of as a simplified version of the Viola-Jones detectors [5].

While such features may appear to be too unstable, they are surprisingly efficient, as will be demonstrated by experiments. One possible explanation is the following. Rejectors such as those used in [5, 9] use image regions which tend to be uniform. Pixels in the center of these regions serve as reliable representatives of the region's grey levels, and can be used instead of the region's mean.

Since background images are typically sampled from the class of natural images, and the latter tend not to exhibit large grey-level variations over short distances, it is clear that the closer p_1 is to p_2, and the larger T is, the feature will filter out more background images. We now formalize this point by using a very simple approximation to the distribution of natural images, which, while hardly complete, suffices for the task of selecting our simple features.

2.2. Discrimination

We seek features whose output on natural images differ greatly from their output on faces. This can be achieved in various ways, for example by computing the mutual information between a feature and the face class; if the feature has a value of 1 on f of the face images and b of the background images ($0 \leq f, b \leq 1$), the mutual information equals $f \log \left(\frac{2f}{f+b} \right)$. Other definitions for feature-category affinity exist, but experience indicates that it is not very crucial which one is used.

The grave difficulty when classifying faces vs. an arbitrary background lies in the estimation of b. Usually, it is estimated from a huge set of "non-face" images. Clearly this set is rather hard to describe and approximate by even a very big sample (and it is not clear how such a sample should be constructed), and if the space of possible features is large, the computational price of testing them may become prohibitively high. In [6, 11], the authors suggest to use a simple prior on the space of natural images in order to design rejectors which are constructed so as to accept the positive examples, while rejecting most natural images. In these two papers, a simple Gaussian prior was applied in the DCT domain; it assigned higher probabilities to smoother images. As reported in [11], this simple prior allowed to construct a linear classifier which did better than linear SVM, the improvement being more noticeable when a small training set was used. This suggests that when estimating the output of relatively simple (e.g. linear) operators on natural images, it is enough to use a simple prior on the space of natural images to obtain a good approximation to the operator's output.

The advantage of using such a prior is substantial; it allows to construct a rejector *without using any negative examples* which, as noted, is very important when the negative examples have to represent the space of all "non-faces". It also yields a vast reduction in computation.

2.3. The Prior

We have opted to use a very simple prior on the space of natural images. This prior – which was used to choose the features – is simply a Gaussian, studied from about 5,000 non-face images. While this prior is evidently far from providing a faithful description of natural images (see e.g.

[12, 13]), it does a decent enough job predicting the output of our simple rejectors (see Section 3.1), which suffices for our cause.

The great advantage in using the prior is that we can easily compute – in a constant number of operations – the rejection ratio not of just one feature, but of any combination of features, by applying standard probabilistic tools to the normal distribution used to approximate the probability distribution of natural images,

2.4. Feature Independence

After constructing a pool of individual features having high mutual information with respect to the face class (this is done by searching over the feature space, a relatively fast process, since we don't use any negative examples), the next task is too choose a subset of k (we used $k = 6$) features which are independent. This is a typical procedure when combining a few rejectors – the more independent they are, the better they will reject. Even with the very fast testing of independence enabled by the normal distribution model, it is still not possible to go over all subsets of 6 features. We therefore used the following sub-optimal algorithm:

1. Use exhaustive search to determine a pool of 100 individual features with the highest mutual information. Hereafter, the search will be restricted to this feature pool only.

2. For all the pairs f_1, f_2 of features from the pool, compute the independence measure $\frac{\Pr(f_1)\Pr(f_2)}{\Pr(f_1 \wedge f_2)}$, where $\Pr(f_1)$ etc. is the probability for a natural image to satisfy the feature f_1. Note that this computation, too, is very fast when using the normal distribution prior over the background class. Choose the 100 pairs with the highest independence measure.

3. Continue in the same fashion, constructing the 100 feature triplets with the highest independence measure by computing the independence measure of every individual feature with the 100 best pairs, etc.

4. Output the subset of 6 features with the highest independence measure.

3. Examples and Experiments

We now show some examples of the concepts defined in the previous Sections, as well as experimental results.

3.1. The Prior

To estimate the prior distribution of the background class, we computed the mean and covariance over a database of 5,500 19×19 natural images, taken from the Caltech database. The resulting covariance matrix is depicted in Fig. 1. Recall that the prior we use is a Gaussian

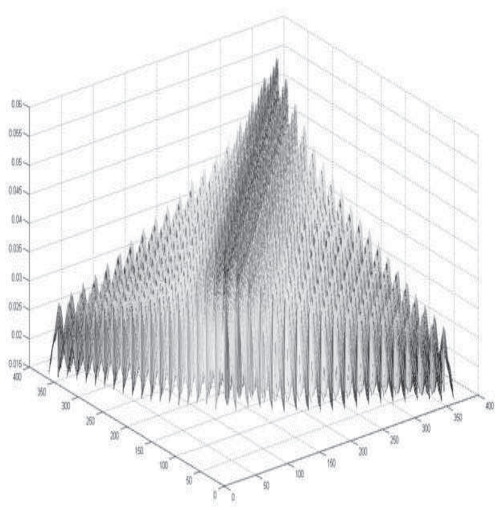

Figure 1. The prior's covariance matrix for 19×19 natural images (since the images are vectorized, the matrix is of size 361×361). The covariance matrix vividly demonstrates the well-known fact that correlations between pixels decrease as the distance between them increases. For the original color version, mail the second author at dkeren@cs.haifa.ac.il.

with the mean and covariance of the images taken from the Caltech database. We suggest that using this simple prior is good enough for our purpose, which is to predict the percentage of natural images that are rejected when using the features defined in Section 2.1. To demonstrate this, we have to compare the rejection probability of features as predicted by the prior (we call this the "model probability") with the rejection probability computed over many real images (we call this the "empirical probability"). Since we're interested in rejection, it suffices to look only at features for which the model probability of rejection is high. To do this, we randomly selected 100 pairs of pixel locations, and for each pair p_1, p_2 found the threshold T such that the model probability for rejection (that is, the probability that for a natural image I, $I(p_1) - I(p_2) < T$) is 0.97. This search is very fast, since the probability is monotone in T (incidentally, this property of the features can speed up the search for the feature pool described in Section 2.3). Then, we compared the model probability with the empirical probability for the 100 resulting features. The mean of the empirical probability over the 100 features equaled 0.9736, and its standard deviation was 0.0032. While the model probability differs from the empirical probability, this result is typical in the sense that the model probability is a rather accurate predictor for the feature's rejection rate on real images.

3.2. Features

In Fig. 2 three features from the pool are depicted, superimposed on a face image of the same resolution we use in the detection process (19×19). The two pixels which define each feature are identically colored; thus, for example, the "green feature" is defined by subtracting the pixel marked by a green minus sign from the pixel marked by a green plus sign, and checking whether the result is greater or equal than 86. The complete information on the three features is provided in Table 1.

Figure 2. Features super-imposed on face. For the original color version, mail the second author at dkeren@cs.haifa.ac.il.

3.3. Experimental Results

In order to make the detection process as fast as possible, we did not apply boosting, but instead searched for simple binary combinations of the values of the individual features which most successfully characterize face images. Using a single feature among the six optimal ones did a decent job, but a simple and highly effective combination was to declare an image to be a face iff it satisfied at least one of the six optimal features. The ROC curve for this combination, as well as those of three individual features, are depicted in Fig 3. 2,000 face images from the CBCL set were used for training, and 400 others for testing. The background set consisted of 18,000 images from the Caltech database. Various points along the ROC curve were created by varying the thresholds of the features (all thresholds were modified by the same amount).

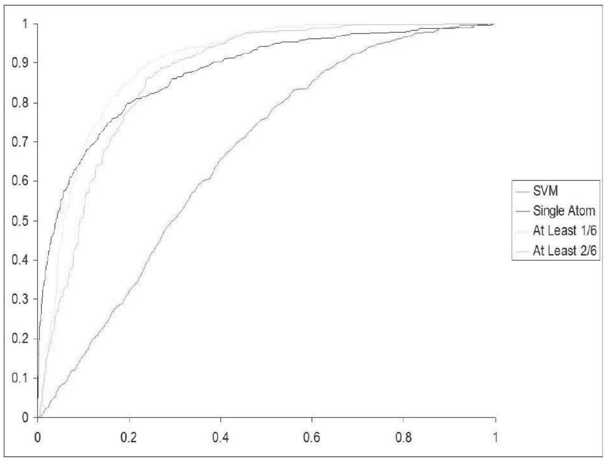

Figure 3. ROC curves of linear SVM, the optimal unique feature, at least one feature of the optimal six, and at least two features of the optimal six. For the original color version, mail the second author at dkeren@cs.haifa.ac.il.

4. Conclusions

A family of very simple features for face detection, which can be applied with minimal space-time requirements, was presented. The features are defined by thresholding the difference among the grey levels of two pixels. A simple prior on the background class allows to train the detectors without using any negative examples. Future work will address textured regions and more general illumination changes.

feature “color”	pixel locations	threshold	face rejection rate	non-face rejection rate
green	(10,5) (+) , (4,7) (-)	86	0.089	0.965
red	(2,10) (+) , (5,5) (-)	93	0.0873	0.971
blue	(7,10) (+) , (5,16) (-)	91	0.093	0.965

Table 1. Three of the features in the pool.

5 Acknowledgement

This work was partially supported by the Israeli Ministry of Science.

References

[1] Elad, M., Hel Or, Y., Keshet, R.: Rejection based classifier for face detection. Pattern Recognition Letters **23** (2002) 1459–1471

[2] Hel Or, Y., Hel Or, H.: Real-time pattern matching using projection kernels. IEEE Trans. on Pattern Analysis and Machine Intelligence **27** (2005) 1430–1445

[3] Sun, J., Rehg, J., Bobick, A.: Automatic cascade training with perturbation bias. In: IEEE Conference on Computer Vision and Pattern Recognition. (2004) II: 276–283

[4] Sahbi, H., Geman, D.: A hierarchy of support vector machines for pattern detection. Journal of Machine Learning Research **7** (2006) 2087–2123

[5] Viola, P., Jones, M.: Robust real-time face detection. International Journal of Computer Vision **57** (2004) 137–154

[6] Keren, D., Osadchy, M., Gotsman, C.: Antifaces: A novel, fast method for image detection. IEEE Trans. on Pattern Analysis and Machine Intelligence **23** (2001) 747–761

[7] Baker, S., Nayar, S.: Pattern rejection. In: IEEE Conference on Computer Vision and Pattern Recognition, San-Francisco (1996) 544–549

[8] Romdhani, S., Torr, P., Schoelkopf, B., Blake, A.: Efficient face detection by a cascaded support-vector machine expansion. Proceedings of The Royal Society A **460** (2004) 3283–3297

[9] Avidan, S., Butman, M.: The power of feature clustering: An application to object detection. In: NIPS. (2004)

[10] Werman, M., Pele, O.: Robust real time pattern matching using bayesian sequential hypothesis testing. IEEE Trans. on Pattern Analysis and Machine Intelligence (To appear)

[11] Osadchy, M., Keren, D.: Incorporating the boltzmann prior in object detection using svm. In: IEEE Conference on Computer Vision and Pattern Recognition. (2006) II: 2095–2101

[12] Lee, A., Pedersen, K., Mumford, D.: The nonlinear statistics of high-contrast patches in natural images. International Journal of Computer Vision **54** (2003) 83–103

[13] Weiss, Y., Freeman, W.: What makes a good model of natural images? In: IEEE Conference on Computer Vision and Pattern Recognition. (2007) 1–8