

Multi-view Face Detection Using Support Vector Machines and Eigenspace Modelling

Yongmin Li, Shaogang Gong, Jamie Sherrah and Heather Liddell
Department of Computer Science, Queen Mary and Westfield College
University of London, London, E1 1NS, UK
{yongmin, sgg, jamie, heather}@dcs.qmw.ac.uk

Abstract

An approach to multi-view face detection based on head pose estimation is presented in this paper. Support Vector Regression is employed to solve the problem of pose estimation. Three methods, the eigenface method, the Support Vector Machine (SVM) based method, and a combination of the two methods, are investigated. The eigenface method, which seeks to estimate the overall probability distribution of patterns to be recognised, is fast but less accurate because of the overlap of confidence distributions between face and non-face classes. On the other hand, the SVM method, which tries to model the boundary of two classes to be classified, is more accurate but slower as the number of Support Vectors is normally large. The combined method can achieve an improved performance by speeding up the computation and keeping the accuracy to a preset level. It can be used to automatically detect and track faces in face verification and identification systems.

1 Introduction

Face recognition, as one of the popular issues in biometrics, has been addressed extensively over the past decade. For a real-time automatic face recognition system, faces should usually be detected as quickly as possible before identification is performed.

Most of the previous work in face detection is limited to the frontal view. Sung and Poggio proposed a Neural Network (NN) based approach which uses a set of face and non-face prototypes to build the hidden layers. The distances between a detected pattern and each of the prototypes are measured to synthesize the final output [11]. Another NN-based approach proposed by Rowley *et al.* can cope with rotation in the image plane by designing an extra NN to estimate the orientation of the face [9]. Osuna *et al.* presented an SVM-based approach for frontal view face detection [7]. If a single classifier cannot solve the problem efficiently, hierarchical methods can be applied. Obvious patterns are classified by simple and fast detectors at the low level(s), and the difficult ones are left to more complex classifiers at the high level(s). For example, Sung and Poggio used a first-stage classifier that is very fast to discard the obvious non-faces [11]. Osuna *et al.* used the similar strategy by

designing an extra SVM classifier in an initial step which can quickly discard most non-face patterns [6].

However, the problem of dealing with rotation in depth and hence being able to detect faces across multiple views remains difficult. Moghaddam and Pentland used an eigenface method for face detection and recognition across multiple views [3]. Their method is to build a set of parallel “observers”, each trying to detect faces in a distinct view, then to determine the location and orientation by a *maximum-likelihood* principle. Alternatively, an SVM-based multi-view face detection and pose estimation model was introduced in [2].

In this paper, we present an approach to multi-view face detection based on pose estimation. SVM Regression is employed to estimate the head pose. Using the pose information, the problem of face detection across large range of views is decomposed into a set of sub-problems, each of them for a small range of views. The task of detection is accomplished by exhaustively scanning and matching in sub-images which may contain faces. We present three methods for multi-view face detection: the eigenface and SVM-based methods extended to the multi-view case, and a novel combination of the two methods aiming to improve the overall performance in terms of speed and accuracy.

The paper is organized as follows: The systematic approach to multi-view face detection is described in Section 2. The issue of pose estimation using SVM Regression is briefly introduced in Section 3. Section 4 presents and compares the three methods for multi-view face detection. Section 5 provides the experimental results of applying those methods to multi-view face detection. The conclusions of the paper are drawn in Section 6.

2 Detecting Faces across Views Based on Head Pose Estimation

We adopt the 2D appearance of face images as the representation in the approach. One of the most challenging problems of this kind of representation is that faces look very different across multiple views. Thus the distribution of face patterns in feature space may be quite irregular. However, if the view information of faces is available, the problem of multi-view face detection can be decomposed into a set of sub-problems, for examples, face detection in up-frontal, up-profile, down-frontal and down-

profile views, which are much easier than the original problem. Based on this idea, we propose an approach based on head pose estimation. The process is described briefly as:

1. use motion estimation, skin colour detection, and background subtraction to segment sub-images containing faces;
2. scan the sub-images with different scales;
3. for each scan, estimate the likely “pose” (in tilt and yaw) of the image patch;
4. choose a appropriate face detector from a set of multi-view face detectors according to the estimated “pose” to determine whether the pattern is a face;
5. synthesize all detections to a single detection.

It is important to note that most computation comes from the exhaustive scanning, including pose estimation and face detection. This situation can be alleviated by

1. using a Kalman filter to track the position, scale, and pose of the detected face in successive frames;
2. performing pose estimation every $n(n > 1)$ frames instead of every frame.

However, the computation and efficiency in face detection are still the key factor to influence performance.

3 Head Pose Estimation Using SVM Regression

We adopt the method of SVM Regression [12, 1, 10] to perform the task of head pose estimation. Two pose estimators, one for yaw and the other for tilt, are constructed to estimate head pose. The image patterns are preprocessed first using background subtraction and colour normalization, then Principal Component Analysis (PCA) [8, 5] is performed to extract dominant features and reduce dimensionality. The obtained PCA vectors are taken as input, and the labelled pose angles (-90° -90° in yaw, and -30° -30° in tilt) as output to train the pose estimators. For a new face pattern, the estimators can give the angles in tilt and yaw directly from the face image (see [2] for more details).

4 Face Detection Methods

We present three methods for multi-view face detection: the eigenface method, the SVM-based method, and a combination of the two methods.

4.1 Eigenface Method

Moghaddam and Pentland introduced the eigenface method, where the confidence $P(\mathbf{x})$ of a pattern \mathbf{x} being a face is modelled by the “distance-in-feature-space” (DIFS) and “distance-from-feature-space” (DFFS) criteria,

$$P(\mathbf{x}) = \left[\frac{\exp(-\frac{1}{2} \sum_{i=1}^M \frac{u_i^2}{\lambda_i})}{(2\pi)^{M/2} \sum_{i=1}^M \lambda_i^{1/2}} \right] \left[\frac{\exp(-\frac{c^2(\mathbf{x})}{2\rho})}{(2\pi\rho)^{(N-M)/2}} \right] \quad (1)$$

where λ_i is the i th eigenvalue, u_i is the projection onto the i th eigenvector, N is the total number of eigenvectors,

M is the number of significant eigenvectors selected in the model, and ρ is an approximation factor [3, 4].

When detecting faces, a *maximum likelihood* strategy is used which takes the image patch with the maximal value of $P(\mathbf{x})$ as the final detection based on an assumption that only one face exists in the detection area. A more general statistical strategy for the classification problem can be described as finding the separating threshold (an optimal confidence value) for the two classes, for example, face and non-face. This strategy is illustrated in Figure 1. The two curves stand for the distributions of confidence values of the two classes. In a two-class classification problem such as face detection, since we are only interested in discriminating positive patterns from negative ones, the eigen-decomposition is performed on the positive class only. Thus the value of a positive pattern is more likely to be higher and the positive curve is usually located to the right of the negative curve. One can choose the optimal separating threshold t_0 as the confidence value of the intersection point of the two curves, assuming equal priors for the two classes.

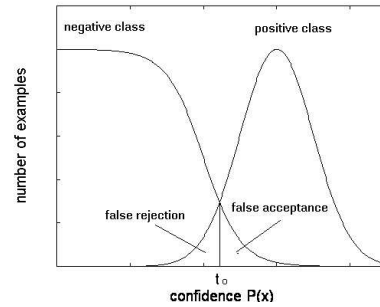


Figure 1: *Eigenface method for classification.*

4.2 SVM-Based Method

Alternatively, an SVM-based face detector can be designed. In this work, we extend the idea of [6, 7] to the multi-view situation based on pose estimation. It is interesting to notice that:

1. while the eigenface method models the probability density of face patterns, the SVM-based method only models the boundary between faces and non-faces;
2. by solving a quadratic programming problem, the SVM-based method is guaranteed to converge to the global optimum;
3. the solution is expressed directly by a subset of “important” training examples called Support Vectors.

4.3 A Combined Learning Approach Using Both eigenface and SVM

From the description above, one notices that the SVM method seeks to model the boundary of classes, thus it is more accurate when detecting faces. However, as the boundary is constructed by a subset of the examples in the training set, which are not necessarily the optimal ones, the

computation on those examples, i.e. SVs, may be expensive. On the other hand, the eigenface method is faster than the SVM method, but less accurate since the false rejection and false acceptance regions may be larger.

We propose a novel approach which combines the eigenface and the SVM methods to achieve improved overall performance in terms of both speed and accuracy. A schematic illustration of the classification criterion of the combined method is given in Figure 2.

The whole process consists of a coarse detection phase by the eigenface method followed by a fine SVM phase. In the first phase, the probability density of each class is estimated as simply as possible. Unlike the eigenface model shown in Figure 1, two thresholds, a rejection threshold (t_r) and an acceptance threshold (t_a), are defined. For a test sample \mathbf{x} , if the value of $P(\mathbf{x})$ given by Equation (1) is less than t_r , it is rejected as a negative example. If the value is larger than t_a , it is accepted as positive. Otherwise, if the value falls between t_r and t_a , it is considered as ambiguous and left to the SVM classifier in the next phase.

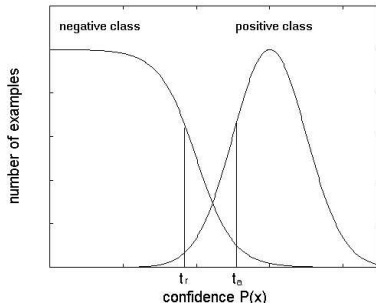


Figure 2: A combined eigenface and SVM model.

An SVM-based classifier is trained on the examples in the middle region of Figure 2. The classifier is only activated when an ambiguous pattern emerges. In most cases, the SVM-based classifier is computationally more expensive than the eigenface method, but more accurate. However, since the proportion of the examples in the ambiguous region is relatively small, a significant improvement of the classification speed can be achieved.

Furthermore, due to the fact that the SVM classifier is trained only on the examples in the ambiguous region and not on the whole training set, the SVM classification problem is simplified to some degree. A more precise and also compact set of SVs are obtained.

Suppose \mathcal{C}_1 and \mathcal{C}_2 are the positive and negative classes to be classified, and n_1 and n_2 are the number of examples in the two classes, as illustrated in Figure 2, the two classification criteria, t_r and t_a , define the false rejection rate r_{fr} and false acceptance rate r_{fa} respectively in the training example set:

$$r_{fr} = \frac{|\{\mathbf{x} : P(\mathbf{x}) < t_r, \mathbf{x} \in \mathcal{C}_1\}|}{n_1 + n_2} \quad (2)$$

$$r_{fa} = \frac{|\{\mathbf{x} : P(\mathbf{x}) > t_a, \mathbf{x} \in \mathcal{C}_2\}|}{n_1 + n_2} \quad (3)$$

One can determine the expected values of r_{fr} and r_{fa} *a priori* through the training data set. For example, one can train the SVM on a small sample of the training data set and yield r_{fr} and r_{fa} . Then the expected r_{fr} and r_{fa} can be set correspondingly. A conservative approach of determining r_{fr} and r_{fa} is to make them small enough so that more examples are handled by the SVM-based classifier. In this case, the final performance in terms of error rate and speed is close to the SVM method.

After calculating the expected number of false rejections $n_{fr} = r_{fr} \cdot (n_1 + n_2)$ and the number of false acceptances $n_{fa} = r_{fa} \cdot (n_1 + n_2)$, t_r is set to the n_{fr} th smallest value of $P(\mathbf{x})$ in \mathcal{C}_1 , and respectively, t_a is the n_{fa} th largest value of $P(\mathbf{x})$ in \mathcal{C}_2 .

5 Experiments

We used 2660 face images taken from 20 subjects, with changes from -90° to 90° in yaw and from -30° to 30° in tilt. The non-face images were collected as the “false positive” detections when bootstrapping the SVM face detector on a set of scenery pictures which do not contain any faces [11]. All images were scaled to 20x20 pixels. Figure 3 shows the representation of the multi-view faces.

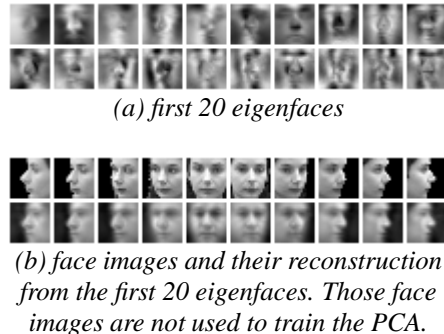


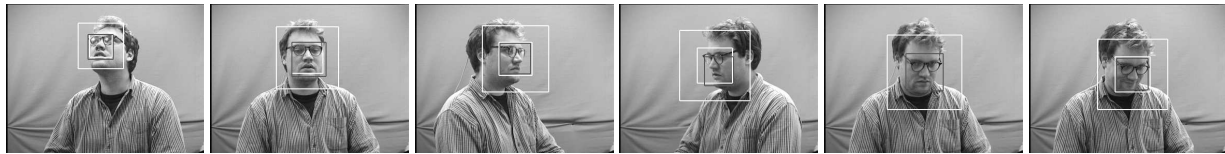
Figure 3: Representation for multi-view face detection.

The comparing results of the three methods on a same sequence are illustrated in Figure 4. The SVM method is the most accurate in terms of error in detection scale and location, but also the slowest. The eigenface method is the fastest, but less accurate in many frames. The combined method demonstrates the best overall performance; it is almost as accurate as the SVM method and not significantly slower than the eigenface method in most frames.

6 Conclusions

We present an approach to multi-view face detection based on pose estimation. SVM Regression is employed to perform pose estimation and face detection is based on exhaustive scanning of sub-images containing faces.

To improve the overall performance of face detection in terms of both speed and accuracy, we implemented and compared three methods for multi-view face detection. The eigenface method and the SVM method were extended to



(a) Detection results of the combined method, where the outer boxes indicate results of segmentation, the small white boxes are the ground-truth positions of faces, and the dark boxes are detections.

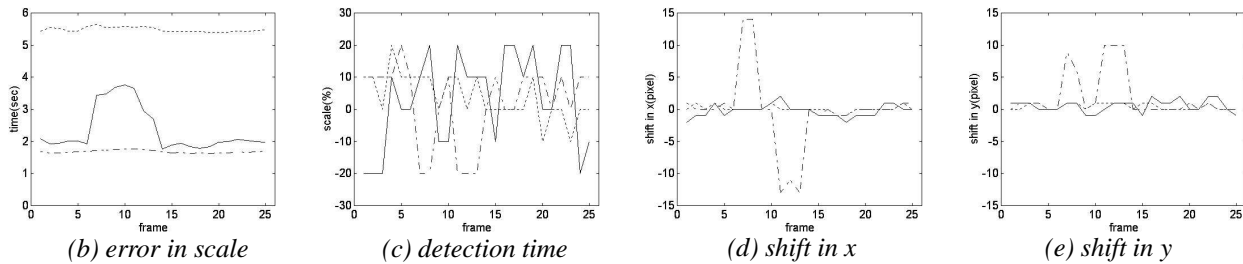


Figure 4: Comparison of SVM, eigenface and combined methods for multi-view face detection. Only the results from the first 25 frame are shown in (b)-(e), where dotted lines for the SVM method, dash-dotted lines for the eigenface method, and solid lines for the combined method. (b) shows the detection time in second for each frame. (c) is the difference in percentage between the detection box and the ground-truth box. (d) and (e) are the shift in pixel of the two boxes.

the case of multi-view face detection. Experimental results show that the eigenface method is faster but less accurate as there is a relatively big overlap between the confidence distributions of face and non-face classes, while the SVM method is more accurate but slower since the number of SVs cannot be efficiently controlled to a small level. By combining the two methods together, we propose a novel method which keeps the advantages and suppresses the disadvantages of both methods. The properties of the combined method lie in:

1. most “obvious” patterns are determined by the eigenface method which is fast;
2. the ambiguous patterns are classified by the SVM method which is accurate;
3. the acceptance and rejection thresholds are calculated based on a preset detection accuracy which guarantees the final accuracy is in an acceptable level;
4. the SVM classifier is trained only on a small set of ambiguous patterns, thus it is more accurate and faster.

The method can be employed to detect and track faces in automatic face verification and identification systems.

References

- [1] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*. MIT Press, Cambridge, MA, 1997.
- [2] Y. Li, S. Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In *IEEE International Conference on Automatic Face & Gesture Recognition*, pages 300–305, Grenoble, France, 2000.
- [3] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):137–143, 1997.
- [4] B. Moghaddam and A. Pentland. Beyond eigenfaces: probabilistic matching for face recognition. In *The 3rd IEEE Int’l Conference on Automatic Face & Gesture Recognition*, Nara, Japan, 1998.
- [5] H. Murase and S. K. Nayar. Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14:5–24, 1995.
- [6] E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. Technical report, Massachusetts Institute of Technology, 1997. AI Memo 1602.
- [7] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. In *Proc. Computer Vision and Pattern Recognition ’97*, pages 130–136, 1997.
- [8] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *IEEE CVPR*, pages 84–91, Seattle, 1994.
- [9] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1), 1998.
- [10] A. Smola, B. Scholkopf, and K.-R. Muller. General cost functions for support vector regression. In T. Downs, M. Frean, and M. Gallagher, editors, *Proc. of the Ninth Australian Conf. on Neural Networks*, pages 79–83, Brisbane, Australia, 1998.
- [11] K. Sung and T. Poggio. Example-based learning for view-based human face detection. Technical report, Massachusetts Institute of Technology, 1994. A.I. MEMO 1521.
- [12] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, 1995.