Face Recognition Using AdaBoost Modular Locality Preserving Projections

Pengzhang Liu†, Kin-Man Lam†, Tingzhi Shen†

*Department of Electronic Engineering, School of Information Science and Technology, Beijing Institute of Technology, Beijing 100081, China
E-mail: lpz639@163.com Tel: +86-010-68912612
†Centre for Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong
E-mail: enkmlam@polyu.edu.hk

Abstract—Locality Preserving Projections (LPP) is a linear manifold learning method proposed for feature extraction which can optimally preserve the neighborhood structure of a data set. Although LPP has been widely applied in image recognition, working as a holistic approach, it is sensitive to variations caused by illumination and expression, and to inaccuracy in face localization. To alleviate these problems, in this paper, we propose to combine the block-based LPP with the AdaBoost algorithm to select the features to improve the accuracy of face recognition. This algorithm, namely AdaBoost Modular LPP (AMLPP), divides a facial image into many overlapping small blocks, and applies LPP to these blocks to form block features. By using 'pseudo-loss' and by updating the distribution of mislabelled samples in the AdaBoost algorithm, AMLPP selects adaptively those optimal block features from a huge set of potential block features to form a number of weak classifiers, which are then combined for the construction of a strong classifier for accurate and efficient face recognition. In each feature-selection process, optimal features are selected to generate weak classifiers, which emphasizes those hard-to-classify samples. Our AMLPP algorithm is compared with the LPP algorithm, the neighborhood preserving embedding (NPE), the discriminant locality preserving projections (DLPP), and the orthogonal locality preserving projections (OLPP), based on the Yale and YaleB face databases. Experimental results show a significant improvement when using our proposed algorithm.

I. INTRODUCTION

Face recognition is one of the most challenging problems in computer vision and pattern recognition. Numerous methods [1-2] have been proposed for face recognition over the past few decades due to its wide applications in many areas, such as intelligent surveillance, access control, and information security. Manifold learning is a popular and recent approach for dimensionality reduction which has been widely used in face recognition. These algorithms are based on the idea that the dimensionality of many data sets is only artificially high (each data point may consist of thousands of features), so they can possibly be described as a function of only a few underlying parameters. In other words, the data points are actually samples from a low-dimensional manifold that is embedded in a high-dimensional space. Manifold learning algorithms attempt to uncover these parameters so as to find a low-dimensional representation of the data [3]. Locality preserving projections (LPP) [4] and neighborhood preserving embedding (NPE) [5] are well-known linear manifold learning methods, which aim at representing data points in a low-dimensional space while preserving the corresponding local structure in the high-dimensional space. The LPP approach has also been extended to become a supervised learning method in [6-7], and some aspects of this extension, known as discriminant LPP (DLPP) [8] and orthogonal LPP (OLPP) [9], have also been introduced.

The AdaBoost algorithm, proposed by Freund et al. [10-11], can theoretically be used to significantly reduce the errors in any learning algorithm that consistently generates classifiers whose performances are slightly better than random guesses. Some improvements [12] have been proposed, namely adaboost.MH, adaboost.MO, and adaboost.MR.

In this paper, we propose an accurate block-based face recognition algorithm that is insensitive to varying illuminations and facial expressions. The LPP algorithm and its extensions are holistic approaches, which are usually more sensitive to variations in illumination and expression than local-feature-based approaches [2]. Therefore, a better performance is expected when a block-based approach is employed, and those reliable blocks are identified for feature selection. Motivated by Viola and Jones [13], the AdaBoost Modular Locality Preserving Projections, denoted as AMLPP, is proposed. In this algorithm, each face image is divided into many overlapping blocks. The blocks located at the same position on the training face images form a block set. The LPP algorithm is then applied to each block set to form corresponding block feature sets. The AdaBoost algorithm then selects efficient block features to form a number of weak classifiers, which are combined to build a simple and efficient strong classifier for face recognition.

II. ADABOOST MODULAR LOCALITY PRESERVING PROJECTIONS ALGORITHM

By considering the local patches in human faces, it is possible to identify those patches which are less affected by variations caused by lighting conditions and facial expressions. By extracting features from these patches, a more
A reliable face recognition algorithm can be derived. In this section, we will discuss the generation of the block features and the selection of the optimal block features which will be used to form the weak classifiers.

### A. Block features

As using block features can reduce the effect of different lighting conditions and facial expressions, each face image is therefore divided into a number of small blocks, and LPP is then applied to each of these blocks to generate the block features for face recognition.

In our AMLPP algorithm, each face image is divided into a number of small blocks which can overlap with each other, and the blocks at the same position in all the training faces form a block set. Each face image is normalized to a size of 128×128, and the images are aligned based on the position of the two eyes, as in [14]. For the block size of 16×16, the number of block sets is 113×113, or 12,769. The position of the block on the \(i^{th}\) row and the \(j^{th}\) column is denoted as \((i, j)\), where \(i = 1, \ldots, H-S_H+1\), \(j = 1, \ldots, W-S_W+1\), \(S_H\) and \(S_W\) are the height and width, respectively, of the blocks under consideration, and \(H\) and \(W\) represent the height and the width, respectively, of the face images. A block of the \(k^{th}\) image \(I_k\) at block position \((i, j)\) can be represented as follows:

\[
I_k^{(i,j)}(m, n) = I_k(i-1 + m, j - 1 + n),
\]

where \(m = 1, \ldots, S_H\), \(n = 1, \ldots, S_W\), \(k = 1\) to \(M\), and \(M\) is the number of images in the training set. All the blocks at position \((i, j)\) form the block set \( \{I_k^{(i,j)}\}_{k=1}^{M} \). In this case, all these block sets of the same size are denoted as \(BS = \{I_{x,y}^{(i,j)}\}_{i,j} \), where \(x = 1, \ldots, S_H\), \(y = 1, \ldots, S_W\), \(i = 1, \ldots, H-S_H+1\), and \(j = 1, \ldots, W-S_W+1\) is the number of blocks in an image.

Supervised LPP is then applied to these blocks to generate a huge number of block features, which will be selected using the AdaBoost algorithm, to form weak classifiers. In the AMLPP algorithm, when the LPP algorithm is applied to different training block sets, or when a training block set is divided into different subspaces of different dimensions, a different hypothesis in Step 2 will be produced.

### B. AdaBoost Modular Locality Preserving Projections

In our algorithm, the AdaBoost.M2 algorithm [11] is employed for feature selection to form weak classifiers, which are then combined to construct a strong classifier. With features of different block sizes, the number of block features available is very large. Thus the AMLPP algorithm selects a critical and small set of block features from the huge number of available block features. In each iteration of boosting, one block feature is selected by using the AdaBoost.M2 algorithm to generate a weak classifier. The process of our proposed AMLPP algorithm is listed in Fig. 1.

### C. Analysis of the AMLPP

Suppose that the block size used is \(S_H \times S_W\), the training image set \(\{I_1, \ldots, I_M\}\) will produce \(N_B\) training block sets, where \(N_B = (H-S_H+1)(W-S_W+1)\). In the AMLPP algorithm, a
To extract the optimal features, we should select the appropriate hypotheses from all the available hypotheses. In AMLPP, each pair of block \(b\) and dimension \(d\) corresponds to a hypothesis, so we compute the ‘pseudo-loss’ \(e_{i}(b, d)\) for each of the pairs, using (2). The ‘pseudo-loss’ is computed with respect to the mislabel distribution \(D_t\), over the set of all sample and incorrect label pairs. By manipulating this distribution, the boosting algorithm can select weak classifiers, which not only emphasize the hard-to-classify examples but, more specifically, the incorrect labels, which are the hardest to discriminate. The generated \(e_{i}(b, d)\) with different values of \(b\) and \(d\) are arranged in increasing order. The block feature with the smallest ‘pseudo-loss’ is selected to form a weak classifier. Only one block feature corresponding to the smallest ‘pseudo-loss’ \(e_{i}(b, d)\) will be selected in each iteration, where \(b_i\) and \(d_i\) are the values corresponding to the \(e_i\) with a minimum in the \(i^{th}\) boosting process. The weak classifiers generated are then combined to construct the final strong classifier.

According to Krogh and Vedelsby [15], a better performance is expected when the selected block features have little correlation as possible to each other. However, when AMLPP is applied to face images, it is found that those blocks corresponding to the selected features are always concentrated in a small region, such as around the eyes. This means that the weak classifiers generated by the selected block features are highly correlated to each other. Thus, in our algorithm, the selected blocks are not allowed to overlap with each other by more than a certain percentage. A threshold, denoted as \(R_t\), is defined as the largest overlapped ratio between the selected blocks. The ‘block overlapped ratio’ for two block features, denoted as \(R_{xy}\), is computed as the ratio between the area overlapped by the corresponding blocks of the two block features and the total area of the blocks.

In each boosting process, we arrange the block features in increasing order according to their ‘pseudo-loss’, and process the ordered block features one by one, by computing all the ratios \(R_{xy}\) between the block feature under consideration and all the already selected block features. The first block feature, whose generated block overlapped ratios \(R_{xy}\) are all smaller than the threshold \(R_t\) will be selected. The AMLPP algorithm will be stopped when all the block features have been processed. In this way, the number of weak classifiers \(T\) to be used can be determined automatically.

III. EXPERIMENTS

A. Databases

The Yale and YaleB databases were used to evaluate the performance of AMLPP. The images in these two databases have a large variation in illuminations and facial expressions. The Yale database contains 165 images with 15 distinct subjects. These images have different facial expressions: neutral, sad, happy, sleepy, surprised, and winking.

The YaleB database contains 5,760 faces with 10 distinct subjects. The images are under 576 viewing conditions (9 poses, 64 illumination conditions). In our experiments, we used the frontal-view face images under the 64 different illumination conditions. Fig. 2 shows some sample face images from the Yale and YaleB databases.

B. Experimental results

In the experiment, in order to balance accuracy and computational complexity, we select block features with a block size of 16×16. However, in order to avoid losing global information, we also consider block features based on a block size of 32×32. The two respective strong classifiers are combined, and a query image \(x_q\) will be classified by using the following combined classifier:

\[
C(x_{\text{test}}) = \arg \max_{y \in Y} \sum_{i=1}^{N_{16}} (\log \frac{1}{\beta_{16}} y_{16}(x_{x}, y) + \sum_{i=1}^{N_{32}} (\log \frac{1}{\beta_{32}} y_{32}(x_{x}, y)),
\]

where \(N_{16}\) and \(N_{32}\) are the numbers of block features selected for block sizes of 32×32 and 16×16, respectively. When the \(i^{th}\) selected block feature of a query image is used for classification, its probabilities of belonging to class \(y\) are \(h_{16}(x_{x}, y)\) and \(h_{32}(x_{x}, y)\) for block sizes of 32×32 and 16×16, respectively.

In the matching step, a local search is performed when comparing two faces so as to alleviate the effect of imperfect alignment and local distortions. At the position of each selected block feature, the block feature of each face in the training set is compared to the corresponding block features within a neighborhood at the corresponding position in the query image, and the differences are computed. The block feature of the query image with the smallest difference will then be selected for classification.

In our experiment, the proposed AMLPP algorithm was compared with the supervised LPP (SLPP), supervised NPE (SNPE), DLPP, and OLPP on the Yale and YaleB databases. The Yale database was used to evaluate the performance of AMLPP on images with varying facial expressions, and the YaleB database was used for evaluating the performance under illumination variations. In order to avoid any bias caused by choosing the training and testing sets, 10-fold cross-validation was used in the experiments. The respective performances of the algorithms are shown in Tables 1 and 2, with different numbers of training samples selected for each subject. For the SLPP, SNPE, DLPP, and OLPP algorithms, the numbers in brackets represent the dimension where the corresponding algorithms achieved the best performances in the Tables 1 and 2.

It can be seen from Tables 1 and 2 that our proposed algorithm performs much better than the other algorithms. For the Yale database, the OLPP algorithm outperforms the other

Fig. 2 Sample face images from the Yale and YaleB databases.
three algorithms, but the best recognition rate achieved by AMLPP is still about 2%-3% higher than that of the OLPP, and 4%-5% higher than that of supervised LPP. With the YaleB database, the improvement is even larger. The higher recognition rate achieved is due to the fact that the AMLPP algorithm divides the facial images into small blocks, so that the effect of illumination and facial expression variations on face recognition can be alleviated. In the AMLPP algorithm, the optimal block features which are robust to illumination and facial expression variations are selected to train the weak classifiers. To construct effective strong classifier, the weak classifiers with high performance will be assigned a higher weight, and a lower weight will be assigned to those with lower performance in AMLPP algorithm.

IV. Conclusion

In this paper, a novel supervised manifold learning method, namely Adaboost Modular Locality Preserving Projections (AMLPP), is proposed for face recognition by combining the block-based LPP algorithm and the AdaBoost algorithm. The AMLPP algorithm extracts block features from face images to alleviate the effect of illumination or facial expression variations on face recognition. The AdaBoost algorithm is used so that efficient features are selected and a strong classifier can be constructed. In our algorithm, the optimal block features generated by the block-based LPP are selected to train the weak classifiers according to the ‘pseudo-loss’. A better performance can be achieved by assigning a higher weight to those weak classifiers with a higher accuracy level. The number of block features to be selected, and the dimensionality of the reduced subspace where the blocks are mapped to, are both determined automatically by the AMLPP algorithm. By updating the mislabel distribution in each boosting process, the AMLPP can focus the weak classifiers not only on hard-to-classify examples but, more specifically, on the incorrect labels, which are the hardest to discriminate. Experimental results on the Yale and YaleB databases show a significant improvement over the supervised LPP, the supervised NPE, the DLPP, and the OLPP algorithms for face recognition under different illuminations and/or facial expressions.

ACKNOWLEDGEMENT

The work described in this paper was supported by an internal grant from the Centre for Signal Processing, The Hong Kong Polytechnic University, Hong Kong (Project No. 1-BB88).

REFERENCES