Weighted-PCANet for Face Recognition

Jiawen Huang(✉) and Chun Yuan

Graduate School at Shenzhen, Tsinghua University,
Shenzhen 518055, Guangdong, China
huangjw13@mails.tsinghua.edu.cn, yuanc@sz.tsinghua.edu.cn

Abstract. Weighted-PCANet, a novel feature learning method is proposed to face recognition by combining Linear Regression Classification model (LRC) and PCANet construction. The sample specific hat matrix is used to handle different images in feature extraction stage. After appropriate adaption, the performance of this new model outperform than various mainstream methods including PCANet for face recognition on Extended YaleB dataset. Particularly, various experiments testify the robustness of weighted-PCANet while dealing with less training samples or corrupted data.

Keywords: Convolutional neural network · Principal component analysis · Linear regression classification · Feature extractor · Face recognition

1 Introduction

Manifold learning methods are known to play an important role in face recognition systems [14]. The original image space is such a high-dimensional space that the feature extraction stage is necessary. Lower-dimensional vectors in the face space are the goal of many mainstream recognition algorithms. Over the last several decades, considerable efforts have been devoted to designing appropriate feature extractor, such as the Local Binary Patterns (LBP) [1] features, which use one of the best performing texture depicters in face recognition.

Furthermore, another kind of popular feature depicter is sparse coding. Sparse Representation-based Classification (SRC) [15], a powerful tool in distinguishing signal categories which lie on different subspaces, is based on a reconstructive perspective. An improved method, Superposed Sparse Representation based Classification (SSRC) [4], in which the intra-class differences are based on the P+V model, casts the recognition problem as finding a sparse representation of the test image in terms of a super-position of the class centroids. The Relaxed Collaborative Representation (RCR) [16] model, effectively exploits the similarity and distinctiveness of features, with each feature vector coded on its associated dictionary of coding vectors to address the similarity among features.

What's more, deep neural networks (DNNs), utilizing two or more successive layers of low-level feature extractors and a following supervised classifier to extract high-level features, is a representation of high-level feature descriptors. Particularly, Convolution Neural Networks (CNNs) [9], the key concepts

S. Arik et al. (Eds.): ICONIP 2015, Part IV, LNCS 9492, pp. 246–254, 2015.
DOI: 10.1007/978-3-319-26561-2_30

of which are local receptive fields and tied weights, has been the most popular model for face recognition because of its excellent ability in numeral complex datasets [6–8]. The utility of local receptive fields makes CNNs more efficient in training and more stable in transformation data and weight-tying observably reduces the number of learnable parameters.

However, despite the superior success of these deep network architectures, experience of weight-tuning and extra training tricks are necessary for these deep networks. Therefore, some researchers have exerted themselves to work on simplifying the training process.

Tiled Convolution Neural Networks (TCNN) [11] base on a statistical method Topographic Independent Component Analysis (TICA), obtains better robustness and competitive performance than CNNs. Its application in LFW dataset [18] even obtains the state-of-the-art record. Random weight-TCNN [12] analysis and demonstrate the importance of convolution pooling architectures, which can be inherently frequency selective and translation invariant, even with random weights. Even though the random-weight TCNN has already immensely simplified CNNs, the performance and efficiency cannot be ensured at the same time. Then another simple deep network appeared. PCANet [2] combines principal component analysis (PCA) with deep neural networks, earning new records for many classification tasks including several face recognition tasks.

Many variants of PCA have been presented since PCA is widely used for dimensionality reduction in computer vision fields, especially for face recognition technology, such as LRC [10], bases on a class-specific hat matrix.

Based on this concept, an efficient neural network model weighted-PCANet, attempting to more detail analyses in sample feature extraction is proposed for face recognition in this paper. When extracting features, the importance of different training examples is also considered because the information for face recognition in every training image is not the same and this may influence the performance of feature extracting. Taking the advantage of the workedout resolution matrix of LRC and the efficiency of cascading construction in PCANet, the weighted-PCANet improved the ability of handling small-amount of training set and corrupted inputs including many kinds of transformation, such as noises and shelters. After combining the LRC idea and the PCANet neural network model suitably, this new model outperforms many current methods including PCANet, in Extended YaleB dataset [5], a benchmark database in face recognition, reducing 20 % of the error rate.

2 Algorithm

2.1 Sample Specific Hat Matrix

Since the information for feature extracting in every training sample is different, a sample specific hat matrix H is set to depict more details about the whole training set, which is denoted as follow,

$$H_i = S_i(S_i^T S_i)^{-1} S_i^T \tag{1}$$

where S_i is defined as the feature extracted from the i_{th} sample. The weight matrix is developed from LRC [10] algorithm, which is a modified feature extractor of PCA. Assume that there are N training samples. The matrix F contains all feature vectors from N samples $F = [S_1, S_2, ..., S_N]$. In order to adapt the LRC algorithm to unsupervised extractor, the hat matrix in LRC is changed in the light of feature extraction from all image samples no matter which class it came from. Assume that y_i is the feature matrix of S_i after projection.

$$y_i = \beta_i S_i + e \tag{2}$$

where $_i$ is the projection parameter and e is an error vector whose components are independent random variables with mean zero. The goal of feature extraction is to find a set of suitable $\widetilde{\beta_i}$ to minimize the residual errors e. In hence, the projection coefficients can be solved through the least-square estimation in LRC and can be written as a matrix form in (4)

$$\widetilde{\beta_i} = (S_i^T S_i)^{-1} S_i^T y \tag{3}$$

Then the feature matrix y_i and the sample specific matrix H_i can be denoted as

$$y_i = S_i(S_i^T S_i)^{-1} S_i^T x, H_i = S_i(S_i^T S_i)^{-1} S_i^T \tag{4}$$

where x is the input of previous layer. And y_i can be expressed as $y_i = H_i x$

There are two advantages of this adaptation:

1. The abandon of the idea of extraction features between different classes through class specific hat matrix in original LRC keeps the feature extraction stage an unsupervised process. The importance of unlabeled extractors is well-known to the explosive-growth data nowadays. At the same time, this convert keeps more information from the input data, and after training the filters becomes more suitable for the whole training set and more powerful for the testing set.
2. Since the neural network model of PCANet has already blocked the raw images, the grouping in original LRC is unnecessary in weighted-PCANet. Every training sample in weighted-PCANet is treated as a set of block-size input data, making this operation a kind of grouping of the raw samples.

2.2 Weighted-PCANet

Propose there are N image samples in the dataset for training. For randomness,these samples are out-of-order firstly. Every sample X, is taken into $b = k \times k$ overlapping patches, and then the mean of every patch is subtracted, denoted as $\bar{X}_i$. The constructed matrix of a training sample X is defined as

$$\bar{X} = [\bar{X}_1, \bar{X}_2, ..., \bar{X}_N] \tag{5}$$

The target of feature extractors is to find a suitable set of feature F, which is a concise representation of original data. So a family of orthonormal filters A is needed to minimize the reconstruction error as follows,

$$min\|\bar{X} - AF\|^2, s.t. D(F) = I_L \tag{6}$$

where $D(F)$ is the variance of F. As mentioned in Sect. 2.1, a modified method of PCA can be used to resolve this optimization problem. Since the raw image is blocked, the original LRC is already inappropriate for deep neural network models. Therefore, a sample specific hat matrix is designed in (7),

$$\bar{Y} = mean(\bar{X}) \tag{7}$$

where $\bar{Y}$ is the mean block of $\bar{X}$. So the optimization problem is changed into

$$min\|\bar{Y} - AF\|^2, s.t.D(F) = I_L \tag{8}$$

The solution is known as described in Subsect. 2.1. The filters are therefore expressed as

$$W = mat(q(\bar{Y}_i(\bar{Y}_i^T\bar{Y}_i)^{-1}\bar{Y}_i^T)) = mat(q(H)) \tag{9}$$

where $mat_{k_1,k_2}(x)$ is a function that maps $x \in R^{k_1,k_2}$ to a matrix $W \in R^{k_1 \times k_2}$ and q denotes the principal eigenvectors of the sample specific matrix. The leading principal eigenvectors capture the main variation of all the mean-removed training patches. Then, the output for this layer is a convolution result like this,

$$O_i^l = I_i^{l-1} * W_j^l \tag{10}$$

Of course, similar to CNNs, multiple layers of filters can be stacked to extract higher level features. Two layers are set to extract features after various experiments. A pooling layer is set to follow these convolution layers like the traditional CNNs. In pooling layer, the outputs from the last convolution layer are binarized by a Heaviside like function.

$$H_i^l = \begin{cases} 1 & \text{if}\, O_i^l > 0 \\ 0 & \text{else} \end{cases} \tag{11}$$

The order of eigenvalues is reflected by output value multiple a set of exponents of constant 2 because the value of H_i^l is 1 or 0.

$$T_i^l = \sum_{j=1}^{L} 2^{j-1} H_i^l \tag{12}$$

Each of the L_1 images T, $l = 1, ..., L_1$, is partitioned into B blocks. For the feature of face image is not the same in every position of the picture, the blocks is set to be non-overlapping. The histogram of the decimal values in each block is computed, and then concatenated into one vector, denoted as $BHist(T_i^l)$. After this process, the feature of the input sample X is then defined to be the set of block-wise histograms,

$$TF = [BHist(T_i^1), BHist(T_i^2), ..., BHist(T_i^L)] \tag{13}$$

The whole feature extraction procedure of weighted-PCANet is shown in Sect. 2.2.

Algorithm 1. Weighted-PCANet

Input: The training sample x
OutPut:Feature vector as (13)
Step 1: Data normalization: block the images and subtract the block mean. $\bar{X} = [\bar{X}_1, \bar{X}_2, ..., \bar{X}_N]$ by (5)
Step 2: Convolution layer. Change $\bar{X}$ to the image mean block $\bar{Y}$ by (10). Utilize sample specific matrix (1) and (9) to compute the l_{th} matrix W, and the output of this layer is $O_i^l = \{I_R^{l-1} * W_j^l\}$ by (10)
Step 3: Convolution layer. Repeat Step 2.
Step 4: Pooling layer. Binarized the output by (11) and hist them as (12) and (13).

2.3 Comparison with CNNs and PCANet

Weighted-PCANet shares the main construction characteristics With classical CNNs [9] as a cascading neural network, including convolution layers and pooling layers. The advantage of CNNs is taken in the design of weighted-PCANet, such as the utility of local vision fields. However, the traditional back propagation (BP) algorithms in CNNs are replaced with a solved optimization problem. There are two parts in BP algorithm: information transmission and forward error back-propagation, which is extensively utilized in traditional Artificial Neural Networks (ANNs). The common point in all ANN models is to find the optimal solution for the loss function iteratively. Especially in CNNs, the kernel filters are randomly set at the very first time and fine-tuned over and over again by stochastic gradient descent method in the process of forward error back-propagation. Not only does the iteration refinement last so long, but also there are many problems about the gradient descent method such as gradient diffusion and convergence to the local optimum which need abundant training experience. Instead, there isnt any forward error back-propagation process in weighted-PCANet. Moreover, the information transmission in it is only conducted for one time. Since the weight matrix of filters is set at a time based on sample specific matrix for all training data. Last but not the least, a worked-out optimization problem in convolution layer simplifies the theoretical analysis of deep neural networks which is an urgent need when the performance of it refreshed new record again and again and the application of it is more and more common.

While comparing with PCANet, in feature extracting stage, the significance of every training sample is different in weighted-PCANet. As shown in Fig. 1, even from the same individual, the global information for face recognition in two samples is far different. Furthermore, LRC is an improved feature depicter of PCA especially in face recognition. So after being elaborately designed to adapt block-calculation in cascading neural network, the performance of this model will outperform the original PCANet without the increase of time.

Fig. 1. Different training samples from Extended YaleB database

3 Experiments

3.1 Dataset and Experiment settings

The Extended Yale B dataset consists of 2414 frontal-face images of 38 human subjects, respectively 64 images for every individual. The cropped and normalized 192×168 face images were captured under various laboratory-controlled lighting conditions. The images are resized to 32×32 pixels. 8 samples are randomly picked for training and the rest for testing. The parameters of weightedPCANet and PCANet-2 [2] are set as follow, 2 convolution layers and 8 filters for each layer. The overlapping block size in convolution layers is 5×5, and the non-overlapping block size in pooling layer is 4×3. The whole model is followed by a linear-SVM classifier [3]. One layer weight-PCANet is also set to prove the excellent ability of multi-layer learning. The setting of CNNs-2 is the same as the famous LeNet-2 [9], 2 layers with filter size 5×5; 20 channels for the first convolution layer and 50 channels for the second convolution layer. Each convolution layer is followed by a 2×2 max-pooling layer. The output layer is a softmax classifier and the whole model is iterated for 500 times. In random-weight TCNNs [12], 20 maps is set for the first layer and 50 maps for second, window size is 5×5 and untied pooling size is 1×1. The model also uses a linear-SVM for classification [3].

3.2 Experimental Results and Analysis

A summary of results is reported in Table 1.Weighted-PCANet obtains the best performance in all testing tasks. Although the training samples are only seventh of the whole data set, weighted-PCANet can get excellent performance because of the painstaking design of feature extraction with sample specific matrix. As analyzed before, when the amount of samples of per individual is small and the difference between every sample is global such as illumination, weighted-PCANet can extract more precise features than other feature depicters. Compare with CNNs-2, the efficiency of the weighted-PCANet is remarkable. Since the use of time without iteration in weighted-PCANet is hundredth of traditional

CNNs-2, the accuracy of weighted-PCANet super outperforms this 500-iteration CNNs-2. Random-weight TCNNs [12] is used to compare with weighted-PCANet because it is also without iteration. However, the performance is barely satisfactory. The performance of weighted-PCANet also outperforms PCANet, LRC and their variants. Especially, the error rate of weighted-PCANet cuts down more than 40 % of LRC and at the same time weighted-PCANet reduces 20 % of the error-rate of PCANet. These results proved that the design of weighted-PCANet applies the LRC in PCANet is suitable for face recognition. The outstanding performance of LBP fea-ture and weighted-PCANet also merits our attentions that these simple methods can also get excellent performance in benchmark dataset.

Table 1. The Error Rates of the Recognition on Extended YaleB Database(%)

Algorithms	Error rate
CNNs-2	96.60
random-weighted TCNNs-2	77.54
LBP [14]	6.91
SVM [13]	23.79
CRC [17]	28.51
RCR [16]	28.11
SRC [15]	41.53
SSRC [4]	40.09
LRC [10]	46.14
PCANets-2 [2]	7.09
weighted-PCANet-1	10.89
weighted-PCANet-2	**5.88**

To prove the analysis mentioned before, the corrupted test images are used for further test tasks, including two levels of contiguous occlusion 10 percent and 20 percent, by replacing a randomly located square block of each image with an unrelated image and multiplicative-Gaussian-noised test image. See Fig. 2 for example. Because the accuracy of CNNs-2 and random-weight TCNNs-2 are too low in the raw test task, they are ignored in these corrupted image tasks. The results are shown in Table 2.

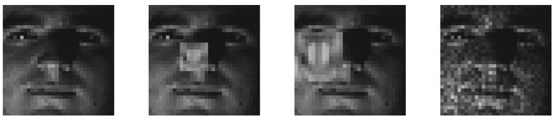

Fig. 2. Corrupted testing images (Left to Right) : raw, 10 % occlusion, 20 % occlusion and noisy image.

Table 2. The Error Rates on Corrupted Test Set(%)

Algorithms	Noised	10 %	20 %
LBP [14]	12.44	8.58	12.73
SVM [13]	24.08	23.91	25.17
CRC [17]	29.72	32.60	53.92
RCR [16]	30.41	49.77	34.45
SRC [15]	41.70	46.03	51.96
SSRC [4]	35.25	40.04	46.49
LRC [10]	46.66	49.77	53.69
PCANets-2 [2]	12.21	7.50	9.62
weighted-PCANet-1	16.24	11.00	13.77
weighted-PCANet-2	**10.77**	**5.88**	**7.14**

It is apparent that weighted-PCANet outperforms other methods even though the images are corrupted seriously. Furthermore, the performance of weighted-PCANet even better than PCANet while in the original testing task. It is a powerful certificate that the new weighted-PCANet model is more robust than PCANet while the input data is contaminate cause the design of combination with LRC for detail conducting in training stage.

4 Conclusion

In this paper, a novel feature learning method by exploiting PCA-Net is proposed to face recognition. The sample specific hat matrix is used to handle different image in training set, which is capable of learning complementary, hierarchical representations. The idea of LRC is applied in PCA-Net after appropriate adaption. Experimental results show the effectiveness of weighted-PCANet for face recognition on Extended YaleB dataset. Particularly, weighted-PCANet outperforms LRC when images are different in illumination and more meticulous than PCANet in information conduct to training samples.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
2. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: a simple deep learning baseline for image classification? arXiv preprint (2014). arXiv:1404.3606
3. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) **2**(3), 27 (2011)
4. Deng, W., Hu, J., Guo, J.: In defense of sparsity based face recognition. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 399–406. IEEE (2013)

5. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.J.: From few to many: illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 643–660 (2001)
6. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint (2012). arXiv:1207.0580
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
8. Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D.: Face recognition: a convolutional neural-network approach. IEEE Trans. Neural Netw. **8**(1), 98–113 (1997)
9. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
10. Naseem, I., Togneri, R., Bennamoun, M.: Linear regression for face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **32**(11), 2106–2112 (2010)
11. Ngiam, J., Chen, Z., Chia, D., Koh, P.W., Le, Q.V., Ng, A.Y.: Tiled convolutional neural networks. In: Advances in Neural Information Processing Systems (2010)
12. Saxe, A., Koh, P.W., Chen, Z., Bhand, M., Suresh, B., Ng, A.Y.: On random weights and unsupervised feature learning. In: Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 1089–1096 (2011)
13. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. **9**(3), 293–300 (1999)
14. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cogn. Neurosci. **3**(1), 71–86 (1991)
15. Shankar Sastry, S., Wright, J., Ganesh, A., Ma, Y., Yang, A.: Robust face recognition via sparse representation. IEEE Trans. Pattern Anal. Mach. Intell. **31**(2), 210–227 (2009)
16. Yang, M., Zhang, L., Zhang, D., Wang, S.: Relaxed collaborative representation for pattern classification. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2224–2231. IEEE (2012)
17. Zhang, L., Yang, M., Feng, X.: Sparse representation or collaborative representation: which helps face recognition? In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 471–478. IEEE (2011)
18. Zhu, Z., Luo, P., Wang, X., Tang, X.: Deep learning identity-preserving face space. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 113–120. IEEE (2013)