LETTER

PCANet-II: When PCANet Meets the Second Order Pooling

Chunxiao FAN[†], **Xiaopeng HONG**[††], ***Nonmembers***, **Lei TIAN**[†,††], ***Member***, **Yue MING**[†], **Matti PIETIKÄINEN**[††], ***and*** **Guoying ZHAO**[††a)], ***Nonmembers***

SUMMARY PCANet, as one noticeable shallow network, employs the histogram representation for feature pooling. However, there are three main problems about this kind of pooling method. First, the histogram-based pooling method binarizes the feature maps and leads to inevitable discriminative information loss. Second, it is difficult to effectively combine other visual cues into a compact representation, because the simple concatenation of various visual cues leads to feature representation inefficiency. Third, the dimensionality of histogram-based output grows exponentially with the number of feature maps used. In order to overcome these problems, we propose a novel shallow network model, named as PCANet-II. Compared with the histogram-based output, the second order pooling not only provides more discriminative information by preserving both the magnitude and sign of convolutional responses, but also dramatically reduces the size of output features. Thus we combine the second order statistical pooling method with the shallow network, i.e., PCANet. Moreover, it is easy to combine other discriminative and robust cues by using the second order pooling. So we introduce the binary feature difference encoding scheme into our PCANet-II to further improve robustness. Experiments demonstrate the effectiveness and robustness of our proposed PCANet-II method.

key words: *second order pooling, binary feature difference, face recognition, pain estimation*

1. Introduction

Face is one of the most interesting subjects in various computer vision tasks, and many face analysis tasks have achieved significant progress in recent years. However, there are still many unsolved problems in real applications, such as extreme intra-class variations and large number of subject classes in face recognition (FR) and affective computing (AC) tasks. In order to solve these problems, large number of learning-based methods [1], [2], especially deep learning (DL) methods [3], [4], have been proposed in recent years. The convolutional neural network (CNN) [5] is a typical deep learning model. Its architecture contains three main components: convolutional layers, activation functions and pooling layers. The FaceNet [3], which follows the idea of end-to-end learning, directly learns the mapping from raw image to Euclidean space. The work [4] integrates the center loss with original softmax loss functions to further enhance the deep features' discriminative power. However, these methods require a large number of samples to train the DL model. Comparatively, a number of "shallow" learning models are proposed, such as PCA-Network (PCANet) [6] and Stacked Image Descriptors (SID) [7]. Lei *et al.* [7] introduced several existing shallow descriptors into the deep model by stacking image descriptor layers and max-pooling layers alternatively. These shallow descriptors include Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Discriminant Face Descriptor (DFD) [1]. Another recent noticeable work, i.e., PCANet [6], achieved good performances for image classification. Compared with CNN, the PCANet does not need the backward propagation process to update the model's parameters, therefore, it performs well on small-scale dataset. The PCANet is considered as a simplified model of CNN [6]. The PCANet follows the basic architecture of CNN and consists of a few convolutional layers, non-linear processing layers and block histogram extraction layers. For convolutional layers, PCANet computes the local patch based filter kernels by the PCA. For non-linear processing layers, the whole feature map is binarized by a unit step hashing function. For the pooling layer, different feature maps are assigned to difference weights. Just like LBP [8], around each pixel, a set of binary values are summed with weights and a decimal-valued image is obtained. The block-wise histograms are computed in each local block from the decimal-valued image. At last, the block-wise histograms are concatenated into a long vector. We abbreviate the **block-wise histogram** as **histogram** hereinafter unless it is specifically indicated.

The PCANet method uses histogram technique as the pooling approach. However, there are three main problems for this kind of pooling methods. **Problem 1:** The binary hashing process sets pixel value of a feature map to be 1 when the original pixel values are larger than 0, otherwise, to be 0. This binarization process makes a lot of discriminative information loss during the pooling process. **Problem 2:** It is difficult for the histogram-based pooling method to combine other effective visual cues during the pooling process. **Problem 3:** The dimensionality 2^L of histogram grows exponentially with the number of feature maps L. Therefore, it limits the number of feature maps in the convolutional layer.

In order to solve the above problems, we propose a novel "shallow" network model, namely PCANet-II. We use the second-order statistics to pool the feature map set. Thus, we obtain more discriminative information from the

Manuscript received November 24, 2017.
Manuscript revised March 29, 2018.
Manuscript publicized May 14, 2018.

[†]The authors are with Beijing University of Posts and Telecommunications, P. R. China.
[††]The authors are with The Center for Machine Vision and Signal Analysis, University of Oulu, Finland.
a) E-mail: guoying.zhao@oulu.fi

DOI: 10.1587/transinf.2017EDL8258

floating based response value of feature maps (w.r.t **Problem 1**). Benefiting from the expandability property of the second-order pooling technology, we can integrate discriminative and robustness properties into our method simultaneously (w.r.t **Problem 2**). Moreover, the second-order pooling (usually is implemented by covariance matrix) is a kind of statistics instead of the distribution variable. So the dimensionality of covariance matrix's output is relatively lower (w.r.t **Problem 3**). We also employ the binary feature difference (BFD) scheme to translate the non-numerical encoding responses to numerical ones, which can be encapsulated into the feature map set, so that the robustness of PCANet-II can be improved.

There are several works about second order pooling for CNN model [9], [10]. However, to our best knowledge, there is no second order pooling scheme for PCANet model. The methods based on CNN model need backward propagation process and lots of samples to update the parameters of network. In contrast, our PCANet-II learns model's parameters by one forward propagation, so it works well on small-scale dataset. Moreover, the computational cost of our PCANet-II is also dramatically reduced.

2. The Proposed Approach

In this paper, we propose a novel "shallow" network model with the second order pooling and BFD, namely PCANet-II. Figure 1 illustrates the framework of our PCANet-II model. We first compute the filter kernels and convolutional responses of all stages. Next, for the convolutional responses of each stage, we compute the BFD images from a set of convolutional feature maps, and cumulate feature maps and BFD images together as a new feature map set. Then, the second order statistics is computed from the new feature map set of all stages. At last, we vectorize the covariance matrix and stack the output of all stages as the final output vector. In the following sections, we will describe each step.

2.1 Convolutional Layers

For convolutional layer, we first extract local patches from all training samples and form a local patch set $\mathbf{A} \in \mathbb{R}^{k_1 k_2 \times NP}$, where $[k_1, k_2]$ and P denotes the size and number of local patch. N is the number of training samples. Then, we compute the leading L_1 eigenvectors $\mathbf{V}_1 \in \mathbb{R}^{k_1 k_2 \times L_1}$ of $\mathbf{A}$ by PCA,

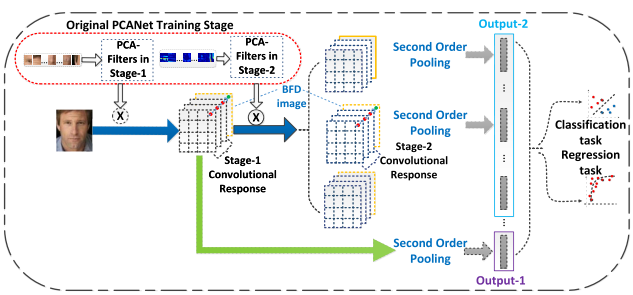

Fig. 1 The framework of our proposed PCANet-II method.

and resize each column vector in $\mathbf{V}_1$ as the matrix of size $[k_1, k_2]$, which is considered as the filter kernel of the first stage. The response of the first stage can be obtained by convolving images with learned filter kernel. We further obtain the filter kernels of the ith stage by learning the leading L_i eigenvectors of the local patch set of the $(i-1)$th stage's convolutional response. At last, we obtain the convolutional responses (i.e., feature maps) of all stages.

2.2 Computation of Binary Feature Difference

Original PCANet produces a kind of discrete patterns' index instead of numerical features through *encoding*. It is meaningless to directly use a discrete index for second-order pooling [11]. In order to further improve the robustness of PCANet-II, we extend the *LBP difference* [11] to a more general BFD encoding scheme, so that non-numerical encoding output can be translated to the numerical one. BFD encodes binary features in any form, which are not limited to the particular form of local binary patterns. Moreover, we successfully apply BFD to the face analysis task, which is out of the scope of [11].

There are L_i convolutional outputs $\left\{\mathbf{f}_l^i\right\}_{l=1}^{L_i}$ in the ith stage, where L_i denotes the number of filter kernels in the ith-stage. We binarize these L_i feature maps by the unit step function $\mathrm{S}(\cdot)$ and obtain the binary feature maps $\left\{\mathbf{B}_l^i\right\}_{l=1}^{L_i} = \left\{\mathrm{S}\left(\mathbf{f}_l^i\right)\right\}_{l=1}^{L_i}$. Therefore, the average pattern of the lth binarized feature map $\mathbf{B}_l^i(x,y)$ in the ith-stage can be defined as the mean as follow:

$$\mu_l^i = \frac{\sum\limits_{(x,y)\in \mathbf{f}_l^i} \mathbf{B}_l^i(x,y)}{N^i}, l = 1, 2, \cdots, L_i, \tag{1}$$

where N^i denotes the dimensionality of ith-stage's feature map. The binary constraint is added to form the mean pattern in integer type:

$$\mu_l^i = \left\lfloor \frac{\sum\limits_{(x,y)\in \mathbf{f}_1^i} \mathbf{B}_l^i(x,y)}{N^i} + 0.5 \right\rfloor, l = 1, 2, \cdots, L_i. \tag{2}$$

Having obtained the average pattern of each feature map, the BFD vector can be computed as $\left(\mathbf{B}_l^i(x,y) - \mu_l^i\right)$. We denote the binary feature difference with float mean and integral mean as BFD-F and BFD-I. The sign information is also used to encode BFD and form a discriminative descriptor as follow:

$$\mathbf{f}_{BFD}^i = \mathrm{sgn}\left(\left\|\mathbf{B}^i\right\| - \left\|\boldsymbol{\mu}^i \cdot \mathbf{1}\right\|\right) \cdot \left\|\mathbf{B}^i - \boldsymbol{\mu}^i \cdot \mathbf{1}\right\|, \tag{3}$$

$\mathbf{B}^i = \left[\mathbf{B}_1^i, \mathbf{B}_2^i, \cdots, \mathbf{B}_{L_i}^i\right] \in \mathbb{R}^{k_1 k_2 \times L_i}$, $\boldsymbol{\mu}^i = \left[\mu_1^i, \mu_2^i, \cdots, \mu_{L_i}^i\right] \in \mathbb{R}^{1 \times L_i}$. $\mathrm{sgn}(v)$ is element-wise sign function, i.e., $\mathrm{sgn}(v) = 1$ when $v \geq 0$, and $\mathrm{sgn}(v) = 0$ otherwise.

We consider the BFD image $\mathbf{f}_{BFD}^i$ as a new kind of feature map. In each stage i, we cumulate this BFD image over the feature maps $\left\{\mathbf{f}_l^i\right\}_{l=1}^{L_i}$ to form the new feature map

set $\mathbf{g}^i = \left[\mathbf{f}_1^i, \cdots \mathbf{f}_{L_i}^i, \mathbf{f}_{BFD}^i\right]$, so $\mathbf{g}^i$ has $L_i + 1$ column vectors.

2.3 Second Order Statistical Pooling

Compared with the histogram-based pooling, the second-order statistics can not only provide more discriminative information of feature maps by preserving the floating-point format, but also enable to integrate other informative cues (such as BFD) into the feature map set.

Since different local face patches have different configuration of facial components, we compute the patch-wise covariance matrices from a set of feature maps. More specifically, there are $(L_i + 1)$ outputs $\left\{\mathbf{g}_l^i\right\}_{l=1}^{L_i+1}$ in the ith stage, and each convolutional output $\mathbf{g}_l^i$ is divided as M patches $\left\{\mathbf{g}_{l,m}^i\right\}_{m=1}^{M}$. The patch-based covariance matrix describes the $(L_i + 1) \times (L_i + 1)$ covariances between any pair of feature maps for the mth local patch. Its formulation is [12]:

$$CovM_m^i = c \times \sum_{l=1}^{L_i} \left(\mathbf{g}_{l,m}^i - \tilde{\mu}_m^i\right)\left(\mathbf{g}_{l,m}^i - \tilde{\mu}_m^i\right)^T, \tag{4}$$

where c is the normalization constant and $\tilde{\mu}_m^i$ is the mean of $\left\{\mathbf{g}_{l,m}^i\right\}_{l=1}^{L_i+1}$. Therefore, the correlation of the local blocks of these "feature maps" is summarized by using the covariance matrix. Moreover, the covariance matrix has positive semi-definite and symmetric properties. So we just need around half the size of covariance matrix. Compared with the first order statistics, our second order pooling approach reduces the dimensionality of each block's output feature from 2^{L_i} to $(L_i + 1)(L_i + 2)/2$. At last, the covariance matrices of all stages are stacked as the final output vector.

3. Experimental Results and Discussion

In order to further prove the generalization of PCANet and our proposed model, we choose three most representative datasets, which are not evaluated in the original PCANet paper, to investigate the performance of our proposed method for classification and regression tasks. For classification task, we use CAS-PEAL-R1 [13] and PaSC dataset [14] for constrained FR and unconstrained FR, respectively. For regression task, we use UNBC-McMaster pain dataset [15] for pain estimation (PE). The performances of PCANet and PCANet-II are improved with the increase of filter's number, while the running time and the output features' size are increased, too. Therefore, we set $[L_1, L_2] = [10, 10]$ for balancing the performance and computational cost. The filter size in the convolutional layer and local patch size in the pooling layer depend on the size of original face image. We set the number of local patch to be 11×11 (i.e., $M = 121$) in these layers. The number of stages in PCANet and our PCANet-II model are set to be 2. We not only directly evaluate the performance of all methods by using original output feature on CAS-PEAL-R1 dataset, but also use the Whitening PCA (WPCA) to reduce the size of all methods' output for a fair comparison with prior methods. The Whitening PCA (WPCA) reduces the dimensionality of all methods' output under comparisons into 1,039, 1,000 and 1,000 for CAS-PEAL-R1, PaSC and UNBC-McMaste datasets, respectively.

Table 1 The accuracy (%) comparison on PEAL-R1 dataset. The results in brackets are output feature with WPCA.

Methods	*Expression*	*Accessory*	*Lighting*
DFD [1]	98.3 (99.0)	93.7 (96.9)	59.0 (63.9)
E-LBP [16]	98.3 (98.7)	92.0 (94.4)	68.7 (72.9)
PCANet [6]	99.2 (99.4)	95.2 (96.2)	64.2 (69.0)
PCANet-II (BFD-F)	**99.4 (99.6)**	**95.5 (96.5)**	**83.5 (84.9)**
PCANet-II (BFD-I)	**99.4 (99.6)**	95.2 (96.2)	83.4 (84.3)

Evaluation on CAS-PEAL-R1 Dataset: For CAS-PEAL-R1 dataset, every image is aligned and cropped into the size of 150×130. Table 1 provides the result comparison between our method and other methods on CAS-PEAL-R1 dataset. We use the Nearest Neighbor (NN) classifier with cosine metric. As seen in Table 1, regardless of with WPCA or without WPCA projection, our method achieved good results on the *Expression* and *Accessory* subsets and significantly outperformed other methods on the *Lighting* subset.

Compared with other state-of-the-art methods, the results demonstrate that our method has excellent robustness and descriptive ability for various intraclass variabilities. Compared with original PCANet, our PCANet-II method not only preserves the magnitude information of feature maps by covariance matrix but also contains sign relationship of feature maps by BFD scheme. Because *Expression* and *Accessory* subsets are well controlled, our PCANet-II achieves slightly better accuracies than PCANet in these two saturated subsets. However, our PCANet-II significantly improves the accuracy of PCANet and other methods when faced with extreme lighting conditions.

Evaluation on PaSC Dataset: For PaSC dataset, we align all images and crop them into 128×128 pixels. Table 2 shows the verification rate at FAR $= 0.01$ of our method and other state-of-the-art methods on PaSC dataset for all images and near-frontal images scenarios, respectively.

Our PCANet-II obtains good robustness by reference to the non-linear process in the PCANet model. Therefore, both of them achieve better performance when facing complicated intraclass variations. Compared with histogram-based pooling scheme of PCANet model, our PCANet-II retains more discriminative power by computing the second order statistics of floating-based feature maps. So our method achieves the best performance on both scenarios. In other words, the second order pooling provides more discriminative power than PCANet, and the BFD scheme brings our method good robustness.

Evaluation on UNBC-McMaster Pain Dataset: The PE is a regression task, so we use the linear kernel Support Vector Regressors (SVR) (hyperparameter $C = 0.1$) to estimate the pain intensity. We align the all images and crop them into 128×128 pixels. The average Mean Squared Error (MSE) is used to describe the performance of evaluated methods. We compare PCANet-II with the state-of-the-art PE meth-

Table 2 Verification rate (%) at FAR = 0.01 on PaSC dataset

method	all	frontal	method	all	frontal
LBP [8]	17.6	29.6	IQBC [17]	21.2	38.8
LRPCA [14]*	10.0	19.0	PCANet [6]	28.6	53.0
CohortLDA [14]*	8.0	22.0	**PCANet-II (BFD-F)**	**30.4**	**54.0**
DFD [1]	21.5	36.1	**PCANet-II (BFD-I)**	30.2	53.7

* We directly cite the results from the original papers.

Table 3 MSE comparison on UNBC-McMaster pain dataset

Methods	MSE	Methods	MSE
PTS [18]	2.59	VGGface+CNN+SVR [19]	1.70
DC [18]	1.71	RCNN+Regression [19]	1.54
LBP [18]	1.81	**PCANet-II (BFD-F)**+SVR	**1.47**
Gradient Histograms [20]	4.76	**PCANet-II (BFD-I)**+SVR	1.48
Hess+Grad [20]	3.35		

ods as shown in Table 3. Compared with other methods, the proposed general method achieved the best MSE result. According to the experimental results, we obtain the similar conclusion with the above experiments. Though the PE is a regression task, the essence of PE task is still the description and representation for facial image. From this view, the results demonstrated the excellent facial description ability and robustness of our proposed model.

4. Conclusion

In this paper, we propose a general "shallow" network for face analysis, named as PCANet-II. Compared with the histogram-based pooling method of PCANet, our model not only gains supplementary discriminative information by preserving both the magnitude and sign of convolutional responses, but also provides robustness by BFD scheme. Our PCANet-II also eases the high dimensionality problem of histogram-based feature. Our method achieves promising performances on both FR and PE tasks.

Acknowledgments

This work was supported by the BUPT Excellent Ph.D. Students Foundation CX2016304, the National Natural Science Foundation of China (No. NSFC-61402046, 61572205), the Academy of Finland, Infotech Oulu, and Tekes Fidipro Program.

References

[1] Z. Lei, M. Pietikäinen, and S.Z. Li, "Learning discriminant face descriptor," IEEE Trans. Pattern Anal. Mach. Intell., vol.36, no.2, pp.289–302, 2014.

[2] J. Lu, V.E. Liong, G. Wang, and P. Moulin, "Joint feature learning for face recognition," Information Forensics and Security, IEEE Transactions on, vol.10, no.7, pp.1371–1383, July 2015.

[3] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.815–823, 2015.

[4] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," European Conference on Computer Vision, vol.9911, pp.499–515, Springer, 2016.

[5] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," Commun. ACM, vol.60, no.6, pp.84–90, 2017.

[6] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: A simple deep learning baseline for image classification?," IEEE Trans. Image Process., vol.24, no.12, pp.5017–5032, 2015.

[7] Z. Lei, D. Yi, and S.Z. Li, "Learning stacked image descriptor for face recognition," IEEE Trans. Circuits Syst. Video Technol., vol.26, no.9, pp.1685–1696, 2016.

[8] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol.28, no.12, pp.2037–2041, Dec. 2006.

[9] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear cnn models for fine-grained visual recognition," Proceedings of the IEEE International Conference on Computer Vision, pp.1449–1457, 2015.

[10] P. Li, J. Xie, Q. Wang, and W. Zuo, "Is second-order information helpful for large-scale visual recognition?," arXiv preprint arXiv:1703.08050, 2017.

[11] X. Hong, G. Zhao, M. Pietikäinen, and X. Chen, "Combining lbp difference and feature correlation for texture description," IEEE Trans. Image Process., vol.23, no.6, pp.2557–2568, 2014.

[12] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on riemannian manifolds," IEEE Trans. Pattern Anal. Mach. Intell., vol.30, no.10, pp.1713–1727, 2008.

[13] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The cas-peal large-scale chinese face database and baseline evaluations," IEEE Trans. Syst., Man, Cybern.-Part A: Systems and Humans, vol.38, no.1, pp.149–161, 2008.

[14] J.R. Beveridge, P.J. Phillips, D.S. Bolme, B.A. Draper, G.H. Given, Y.M. Lui, M.N. Teli, H. Zhang, W.T. Scruggs, K.W. Bowyer, P.J. Flynn, and S. Cheng, "The challenge of face recognition from digital point-and-shoot cameras," Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on, pp.1–8, IEEE, 2013.

[15] P. Lucey, J.F. Cohn, K.M. Prkachin, P.E. Solomon, and I. Matthews, "Painful data: The unbc-mcmaster shoulder pain expression archive database," Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE Conference on, pp.57–64, IEEE, 2011.

[16] L. Liu, P. Fieguth, G. Zhao, M. Pietikäinen, and D. Hu, "Extended local binary patterns for face recognition," Information Sciences, vol.358-359, pp.56–72, 2016.

[17] L. Tian, C. Fan, and Y. Ming, "Learning iterative quantization binary codes for face recognition," Neurocomputing, vol.214, pp.629–642, 2016.

[18] S. Kaltwang, O. Rudovic, and M. Pantic, "Continuous pain intensity estimation from facial expressions," International Symposium on Visual Computing, vol.7432, pp.368–377, Springer, 2012.

[19] J. Zhou, X. Hong, F. Su, and G. Zhao, "Recurrent convolutional neural network regression for continuous pain intensity estimation in video," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp.84–92, 2016.

[20] C. Florea, L. Florea, and C. Vertan, "Learning pain from emotion: transferred hot data representation for pain intensity estimation," European Conference on Computer Vision, vol.8927, pp.778–790, Springer, 2014.