LETTER

Visual Recognition Method Based on Hybrid KPCA Network

Feng YANG[†,††a)], ***Member***, **Zheng MA**[†], ***and*** **Mei XIE**[†], ***Nonmembers***

SUMMARY In this paper, we propose a deep model of visual recognition based on hybrid KPCA Network(H-KPCANet), which is based on the combination of one-stage KPCANet and two-stage KPCANet. The proposed model consists of four types of basic components: the input layer, one-stage KPCANet, two-stage KPCANet and the fusion layer. The role of one-stage KPCANet is to calculate the KPCA filters for convolution layer, and two-stage KPCANet is to learn PCA filters in the first stage and KPCA filters in the second stage. After binary quantization mapping and block-wise histogram, the features from two different types of KPCANets are fused in the fusion layer. The final feature of the input image can be achieved by weighted serial combination of the two types of features. The performance of our proposed algorithm is tested on digit recognition and object classification, and the experimental results on visual recognition benchmarks of MNIST and CIFAR-10 validated the performance of the proposed H-KPCANet.

key words: *visual recognition, KPCA, feature fusion, H-KPCANet*

1. Introduction

Visual recognition is an important research field in Computer Vision and Machine Learning. Over the last several decades, a variety of studies have been done for image classification. At the early of this century, the researches are mainly focus on how to select or design an effective manual feature to impove the classification accuracy. Many feature extraction methods, such as SIFT and HOG, were brought out and widely used in some visual recognition tasks, such as face recognition, object detection and image classifiction.

In the past decade, automatic visual recognition techniques has achieved great improvement due to the breakthrough in deep learning models. The classical model is Convolution Neural Networks(CNNs), followed by many deep learning networks, such as deep CNNs, "Network in Network", Inception, and so on. Althrough the CNNs have achieved great success, they need too much computational costs than manual feature selection methods. Therefore, several simple deep learning models are proposed in recent years. PCANet [1] was introduced by Chan T H by making use of PCA filters. Deep Sparse-Coding Network [2] was discussed by Zhang S by combining the algorithms of sparse-coding(SC) and convolutional neural network(CNN).

Aiming to be a simple baseline method of deep learning networks, PCANet has been extended to some different models, such as L1-(2D)2PCANet [3] and CSGF(2D)2PCANet [4], and has achieved great improvement in various computer vision tasks, such as object recognition and image classification. But the simple CNN model of PCANet is based on the method of PCA, which is a kind of linear projection with the shortcoming of ignoring the non-linear correlation of the input data. In order to solve this problem, we introduce a new visual recognition model based on hybrid KPCA network.

The main contributions of our paper can be summarized as following. 1) The model of Hybrid KPCA Network (H-KPCANet) is proposed in this paper. In order to overcome the drawbacks of PCANet, we extend the PCA model to hybrid KPCA by using kernel function. The nonlinear correlations of the input data are retained and the features from high dimensions are extracted by KPCA algorithm. 2) The feature of H-KPCANet are constructed by strategy of weighted serial fusion. Following the structure of H-KPCANet, an input image will gain two types of features: features from one-stage KPCANet and two-stage KPCANet, respectively. A weighted strategy of feature fusion is applied to achieve the final feature. 3) The effectiveness of our H-KPCANet is tested on applications of digit recognition and object classification. The experimental results show that the proposed method is promising.

The rest of our paper is organized as following: Sect. 2 introduces the proposed H-KPCANet model in detail, Sect. 3 reports experimental results of our method on the datasets of MNIST and CIFAR-10, and finally Sect. 4 concludes the paper with references.

2. Hybrid KPCA Network

In the following paper, an algorithm of visual recognition based on hybrid KPCA Network(H-KPCANet) is discussed in detail. The diagram of the architecture and network is shown in Fig. 1. As demonstrated in this figure, the proposed model consists of four parts: an input layer, an one-stage KPCANet, a two-stage KPCANet and a fusion layer. The main idea of H-KPACNet can be described as following.

Manuscript received March 18, 2020.
Manuscript publicized May 28, 2020.

[†]The authors are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, P.R. China.
[††]The author is with the School of Information and Engineering, Wenzhou Medical University, P.R. China.
a) E-mail: yangfeng_34@163.com

DOI: 10.1587/transinf.2020EDL8041

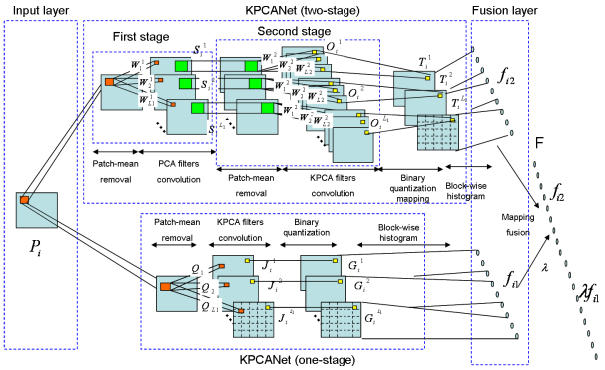


Fig. 1 The detailed outline of the proposed hybrid KPCANet algorithm.

2.1 The Input Layer

According to the settings in PCANet [1], we define the images of training set as $\{P_i\}_{i=1}^{N}$ with the image number of N. The size of the input image is $m \times n$, and the patch used in each layer is with size of $k_1 \times k_2$. Then, all the patches of the input image P_i can be expressed as Eq. (1) under overlapping condition, where $x_{i,j}$ is the j-*th* patch in image P_i, and $m' = m - \lceil k_1/2 \rceil$, $n' = n - \lceil k_2/2 \rceil$.

$$X_i = [x_{i,j}] \in R^{k_1 k_2}, j = 1, 2, \ldots, m'n' \tag{1}$$

2.2 One-Stage KPCANet

As illustrated in Fig. 1, the network of one-stage KPCANet consists of four steps: patch-mean removal, KPCA filters convolution, binary quantization and block-wise histogram.

1) Patch-mean removal. The patches of X_i, which are segmented in previous step should be preprocessed by the method of mean-value removing before KPCA filters convolution. Let's defined the patches after mean-removal in image P_i as $\bar{X}_i$ with the definition of Eq. (2), the training image set P after patch-mean remove can be expressed as the association of $\bar{X}_i$ by Eq. (3).

$$\bar{X}_i = X_i - \frac{1}{N}\sum_{i=1}^{N} X_i \tag{2}$$

$$X = [\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_N] \in R^{k_1 k_2 \times N m'n'} \tag{3}$$

2) KPCA filters convolution. The key problem in the step of KPCA filters convolution is calculating the fiters by a proper kernel function. In our proposed model, Gaussian function is brought out as our kernel function and is illustrated as Eq. (4), in which $v_i, v_j \in R^{m'n'}$ is the row vector of matrix $\bar{X}_i$, $i, j = 1, 2, \ldots, k_1 k_2$ and σ is real parameter.

$$K(v_i, v_j) = exp(-\frac{\|v_i - v_j\|^2}{2\sigma^2}) \in R^{k_1 k_2 \times k_1 k_2} \tag{4}$$

If the kernel transformation of the matrix $\bar{X}$ is marked as K_i, then the result of kernel transformation for the set X can be expressed as Eq. (5). Suppose the number of KPCA filters is L_1. By calculating the minimum value of the reconstruction error in Eq. (6), the KPCA filters that defined as Q_l can be obtained by Eq. (7), where I_{L_1} is the identity matrix with the size of L_1, $matx_{k_1,k_2}(T)$ denotes a mapping function that translates the vector $T \in R^{k_1 k_2}$ into a matrix $Q \in R^{k_1 \times k_2}$ and $q_l(KK^T)$ represents the *l-th* principal eigenvector of the matrix KK^T.

$$K = [K_1, K_2, \ldots, K_N] \in R^{k_1 k_2 \times N k_1 k_2} \tag{5}$$

$$\min_{V \in R^{k_1 k_2 \times L_1}} \|K - VV^T K\|_F^2, \quad s.t. \quad V^T V = I_{L_1} \tag{6}$$

$$Q_l = matx_{k_1,k_2}(q_l(KK^T)) \in R^{k_1 \times k_2}, \quad l = 1, 2, \ldots, L1 \tag{7}$$

After obtaining the KPCA filters Q_l, the outputs of the one-stage KPCANet can be achieved by the convolution of the input image P_i and the KPCA filters Q_l according to the following equation, in which J_i^l is the *l-th* output of image P_i, $l = 1, 2, \ldots, L1$.

$$J_i^l = P_i * Q_l, \quad i = 1, 2, \ldots, N \tag{8}$$

3) Binary quantization. The output matrix of the convolution J_i^l is divided into B blocks, and the pixels in each block are binarized according to the average value of the belonging block. We denote the matrix after binary quantization as G_i^l.

4) Block-wise histogram. Based on the divided B blocks in G_i^l, we calculate the histogram of all blocks of G_i^l and concatenate them into a vector $Bhist(G_i^l)$. As l differs from 1 to L_1, the feature of one-stage KPCANet f_{i1} is achieved by connecting the L_1 histograms as following:

$$f_{i1} = [Bhist(G_i^1), Bhist(G_i^2), \ldots, Bhist(G_i^{L_1})]^T \in R^{L_1 B} \tag{9}$$

2.3 Two-Stage KPCANet

The network of two-stage KPCANet can be divided into four parts: the first stage of PCANet, the second stage of KPCANet, binary quantization mapping and block-wise histogram. Both of the first stage and the second stage consists of opeartions of patch-mean removal and convolution.

1) The first stage

Similar as one-stage KPCANet, the first stage of two-stage KPCANet is an one-stage PCANet, which is composed of patch-mean removal and PCANet filters convolution.

Patch-mean removal. According to Eq. (2) and Eq. (3), the training image set P after patch-mean removal in the first stage is the same as in one-stage KPCANet, which is denoted as X.

PCANet filters convolution. The filters used in the first stage are PCA filters, which are acquired from the minimization of the following Equation:

$$\min_{V \in R^{k_1 k_2 \times L_1}} \|X - VV^T X\|_F^2, \quad s.t. \quad V^T V = I_{L_1} \tag{10}$$

According to Eq. (10), the PCANet filters of the first stage, which is named as W_l^1, can be calculated by the equation below:

$$W_l^1 = matx_{k_1,k_2}(q_l(XX^T)) \in R^{k_1 k_2}, \quad l = 1, 2, \ldots, L1 \tag{11}$$

Therefore, the output of the first stage can be easily gained by the convolution of the input image P_i and the PCA filters W_l^1 using Eq. (12), in which S_i^l is the *l-th* output of image P_i, $l = 1, 2, \ldots, L1$.

$$S_i^l = P_i * W_l^1, \quad i = 1, 2, \ldots, N \tag{12}$$

2) The second stage

The second stage of two-stage KPCANet is composed of patch-mean removal and KPCANet filters convolution.

Patch-mean removal. Similar as the computational method in Eq. (2) and Eq. (3), we denote the mean-removed patches of S_i^l as $\bar{Y}_i^l$. Therefore, we further define the patch-mean removal output of the *l-th* PCANet filter as Y^l and all of the PCANet filters as Y according to the equations below.

$$Y^l = [\bar{Y}_1^l, \bar{Y}_2^l, \ldots, \bar{Y}_N^l] \in R^{k_1 k_2 \times Nm'n'} \tag{13}$$

$$Y = [Y^1, Y^2, \ldots, Y^{L_1}] \in R^{k_1 k_2 \times L_1 Nm'n'} \tag{14}$$

KPCANet filters convolution. The filters used in the second stage are KPCA filters with the kernel Gaussian function defined as Eq. (4). Therefore, the KPCA filters of the second stage, which is named as W_l^2, can be acquired by the following equation, in which K' is the result of kernel transformation for the matrix of Y.

$$W_l^2 = matx_{k_1,k_2}(q_l(K'K'^T)) \in R^{k_1 k_2}, \quad l = 1, 2, \ldots, L2 \tag{15}$$

Then, the output of the second stage can be achieved by the convolution of S_i^l and the KPCA filters W_l^2 according to Eq. (16), in which O_i^l is the *l-th* output of the second stage, $l = 1, 2, \ldots, L2$.

$$O_i^l = \{S_i^l * W_l^2\}_{l=1}^{L2} \tag{16}$$

3) Binary quantization mapping. As calculated by convolution, the outputs of the second stage is up to $L_1 \times L_2$. Therefore, the number of outputs is reduced to L_1 according to binary quantization mapping by the formulation of Eq. (17), in which T_i^l is the hashing results of two-stage KPCANet and $H(\cdot)$ is a Heaviside step function denoted by Eq. (18).

$$T_i^l = \sum_{l=1}^{L_2} 2^{l-1} H(O_i^l) \tag{17}$$

$$H(t) = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases} \tag{18}$$

4) Block-wise histogram. Similar as the histogram operation in one-stage KPCANet, block-wise histogram of T_i^l can be expressed as $Bhist(T_i^l)$ and the feature of two-stage KPCANet f_{i2} can be vectorized as:

$$f_{i2} = [Bhist(T_i^1), Bhist(T_i^2), \ldots, Bhist(T_i^{L_1})]^T \in R^{(2^{L_2})L_1 B} \tag{19}$$

2.4 The Fusion Layer

Based on the features extracted by one-stage KPCANet and two-stage KPCANet, the final feature of the input image P_i can be achieved by weighted serial combination of f_{i1} and f_{i2} in the following function, where λ is the weighted value of f_{i1} and F denotes the final feature of P_i.

$$F = \begin{Bmatrix} f_{i2} \\ \lambda f_{i1} \end{Bmatrix} \tag{20}$$

Based on the model of H-KPCANet described in this section, the input image is transformed into a feature vector.

3. Experiments

We have performed experiments on hand-written digit recognition and object classification to demonstrate the performance of the proposed H-KPCANet. The parameters are set as k_1=k_2=5, L_1=L_2=8. And a linear SVM classifier is applied throughout the experiments.

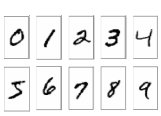
In the following part of this section, the databases of MNIST and CIFAR-10 are tested, and the parameters of λ and σ are discussed detailly on the database of MNIST.

3.1 MNIST Database

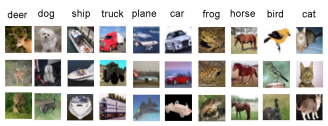
MNIST is a handwritten database of gray images with digit numbers ranging from 0 to 9. The size of each image is 28 × 28, and the total image number is 70,000. The samples of each digit number are shown in Fig. 2(a).

As the first dataset, we test the proposed method in detail in three aspects: the impacts of hyperparameter σ and λ, and the final classification accuracy. In the former two experiments, the hyperparameters of σ and λ are evaluated by 10-folder cross validation and the performances are compared by the average classification accuracy. In the last experiment, the total 70,000 images are split into 60,000 training images and 10,000 testing images.

In testing 1, we investigate the influence of Gaussian kernel σ. We calculate the average classification accuracy of MNIST database with σ varies from 0.5 to 10. The testing results are shown in Fig. 3(a), from which we can see that the classification accuracy achieves the best performance when σ is equal to 6.

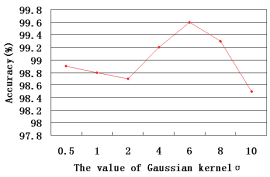


(a) The samples of each digit number in MNIST dataset.

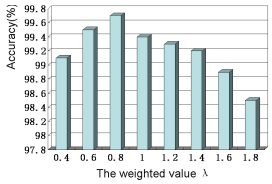


(b) Some samples of CIFAR-10 dataset.

Fig. 2 Some samples of two tested datasets



(a) The value of Gaussian kernel σ and accuracy.



(b) The weighted value λ and classification accuracy.

Fig. 3 The impact of parameters on MNIST database

Table 1 Comparisons with different methods on MNIST Dataset.

Authors	Architecture	Accuracy(%)
Chazal et al.[5]	SLFN	99.1
Tissera et al.[6]	MLP	99.2
Tang et al.[7]	MLP	99.1
Wang et al.[8]	K-Means	99.6
Ciregan et al.[9]	DCNN	99.7
Chan et al.[1]	CNN	99.3
Park et al.[10]	CNN	99.3
Piotr et al.[11]	Photonic RC	99.2
Mairal et al.[12]	CKN	99.6
Ours	CNN	99.5

In testing 2, we discuss the impact of the weighted value λ in the fusion layer. We perform some experiments with different values of λ and the testing results are illustrated in Fig. 3(b). It's easy to see that the best performance is achieved when λ equal to 0.8.

Finally, we compare our proposed algorithm with several methods based on different architectures. The testing results on MNIST database are listed in Table 1, in which the parameters of our proposed networks are set to k_1=k_2=5, L_1=L_2=8, σ=6 and λ=0.8, respectively. Our proposed method achieves slightly improvement with some models, such as ELM [5] and PCANet [1], while the classification rate is a bitter lower than MCDNN [9] and CKN [12].

3.2 CIFAR-10 Dataset

Dataset of CIFAR-10 is widely used in object recognition, containing 60,000 color images of 10 classes. There are 5000 training images and 1000 testing images in each class with the size of 32×32 for each image. The objects in this dataset are with large various in colors, positions, textures and scales. The 10 classes of this dataset, as well as 3 random samples from each class are illustrated in Fig. 2(b)

The parameters of our proposed networks in dataset of CIFAR-10 are set to k_1=k_2=5, L_1=L_2=8, σ=8 and λ=0.8, respectively. The testing results of our proposed method and several algorithms with different architectures on CIFAR-10 database are listed in Table 2. It is easy to see from the table that our discussed network outperforms Chan [1] by more than 4%. It is also clear that the models based on CNN architecture, including our proposed model, work better than Tissera [6] and Chazal [5] that are base on MLP and SLFN architecture on CIFAR-10 dataset.

Table 2 Comparisons with different methods on CIFAR10 dataset.

Authors	Architecture	Accuracy(%)
Chazal et al.[5]	SLFN	45.5
Tissera et al.[6]	MLP	56.0
Chan et al.[1]	CNN	78.6
Park et al.[10]	CNN	82.5
Mairal et al.[12]	CKN-GM	74.84
Mairal et al.[12]	CKN-PM	78.30
Mairal et al.[12]	CKN-CO	82.18
Ours	CNN	82.7

4. Conclusion

In this paper, we propose a model of visual recognition based on hybrid KPCA Network. The testing image from the input layer is processed parallel in an one-stage KPCANet and a two-stage KPCANet. The features extracted from the two types of KPCANets are binary quantificated and fused by the method of weighted serial fusion. Several experiments are performed on the tasks of object classification and digit recognition, and the experimental results validated the proposed H-KPCANet is promising.

References

[1] 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, Dec. 2015.

[2] S. Zhang, J. Wang, X. Tao, Y. Gong, and N. Zheng, "Constructing Deep Sparse Coding Network for image classification," Pattern Recognit.vol.64, pp.130–140, April 2017.

[3] Y.-K. Li, X.-J. Wu, J. Kittler, "L1-(2D)2PCANet: A deep learning network for face recognition," Multimedia Tools and Applications, vol.77, no.10, pp.12919–12934, 2018.

[4] J. Kong, M. Chen, M. Jiang, J. Sun, and J. Hou, "Face recognition based on CSGF(2D)2PCANet," IEEE Access, vol.6, pp.45153–45165, 2018.

[5] P.D. Chazal and M.D. Mcdonnell, "Regularized training of the extreme learning machine using the conjugate gradient method," In 2017 International Joint Conference on Neural Networks, IEEE, 2017.

[6] M.D. Tissera and M.D. McDonnell, "Deep extreme learning machines: supervised autoencoding architecture for classification," Neurocomputing, 2015.

[7] J. Tang, C. Deng, and G.-B. Huang, "Extreme Learning Machine for Multilayer Perceptron," IEEE Trans. Neural Networks and Learning Systems, vol.27, no.4, pp.809–821, April 2016.

[8] D. Wang and X. Tan, "Unsupervised feature learning with C-SVDDNet," Pattern Recognit., 2016:S0031320316301182.

[9] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," In CVPR, 2012:3642-3649.

[10] Y. Park and H.S. Yang, "Convolutional Neural Network Based on an Extreme Learning Machine for Image Classification," Neurocomputing, 2019.

[11] A. Piotr, N. Marsal, and D. Rontani, "Large-Scale Spatiotemporal Photonic Reservoir Computer for Image Classification. IEEE J. Selected Topics in Quantum Electronics, 2019, 99:1-1.

[12] J. Mairal, P. Koniusz, and Z. Harchaoui, "Convolutional Kernel Networks," Advances in neural information processing systems(NIPS), 2014:2627-2635.