IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 January 2023, accepted 13 March 2023, date of publication 20 March 2023, date of current version 23 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3258972

RESEARCH ARTICLE

Global and Local Structure Network for Image Classification

JINPING WANG[1], RUISHENG RAN[1], AND BIN FANG[2], (Senior Member, IEEE)
[1]College of Computer and Information Science, Chongqing Normal University, Chongqing 401331, China
[2]College of Computer Science, Chongqing University, Chongqing 400044, China

Corresponding author: Ruisheng Ran (rshran@cqnu.edu.cn)

This work was supported in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K202100505, in part by the Chongqing Technology Innovation and Application Development Project under Grant cstc2020jscx-msxmX0190 and Grant cstc2019jscxmbdxX0061, and in part by the Project of Natural Science Foundation Project of Chongqing (CQ) Chongqing Science and Technology Commission (CSTC) of China under Grant cstc2016jcyjA0419.

ABSTRACT Principal component analysis network (PCANet) is a feature learning algorithm that is widely used in face recognition and object classification. However, original PCANet still has some shortages. One is that the principal component analysis (PCA) algorithm only extracts features by considering the global structure. The other lies in that the original PCANet only employs one particular single layer convolutional results, which loses the information of other convolutional layers. In this paper, we propose a new simple and efficient convolutional neural network called global and local structure network (GLSNet) to address the problems. The network extracts the features both from the global structure and the local structure of the original data space. Specifically, a principal component analysis (PCA) convolutional layer which learns the filters by PCA algorithm is used to remove the noises and redundant information at the first stage. Then at the second stage, another PCA convolution is added to extract features by considering the global structure. As for the local structure, we use the neighborhood preserving embedding (NPE) algorithm to learn the convolutional filters. At the output stage, the global structure feature extracted by PCA convolution and the local structure feature extracted by NPE convolution is concatenated as a united feature. Furthermore, the first layer convolutional feature is also taken into consideration to obtain shallow-level information. Finally, these features are concatenated as a united feature, and a spatial pyramid pooling layer is followed to pool above the united features. To test the effectiveness of the proposed algorithm, the experiments on some image datasets, including three types: human face dataset, object dataset, and handprinted dataset, proceeded. And it performs better than the original PCANet and some improvement algorithms of PCANet, such as PLDANet, and MMPCANet.

INDEX TERMS Global structure, local structure, convolution neural network, principal component analysis, neighborhood preserving embedding, image classification.

I. INTRODUCTION

Feature extraction [1], [2] is always a fundamental work in all of the machine learning fields. For example, face recognition [3] and medical image classification [4], have become vital techniques in life as the pandemic of COVID-19. Features extraction often indicates the process of transforming the input data into a set of features [5]. And the main goal of feature extraction is to obtain the most representative information of the original data. Feature extraction is an important step in the applications since whether the features extracted contain principal information greatly influences the next operation such as classification.

As the important role of feature extraction, many algorithms have been proposed to extract better features [6], [7]. Scale-invariant feature transform (SIFT) finds the key points in different scale spaces and calculates the directions of the key points [8], [9]. SIFT can extract the highlighted

The associate editor coordinating the review of this manuscript and approving it for publication was Wenming Cao.

features but it is inefficient in blurry images. The local binary pattern (LBP) feature describes the local texture of the images, which is easy to calculate and insensitive to illumination conditions [10], [11]. histogram of oriented gradient (HOG) obtains the feature by calculating and counting the histogram of the gradient direction in the local area of the images [12], [13]. Gabor filter is generally applied in texture recognition as it extracts the relevant feature at different scales and directions [14], [15]. But with the rapid development of deep neural networks (DNN), which has become a better alternative to the above traditional feature extraction techniques.

The deep neural network has almost become the hottest algorithm in computer vision as its extremely superior performance [16], [17]. The merit of DNN lies in that the network can learn the features of the images related to the corresponding labels autonomically instead of the specific feature calculated by the traditional feature extraction algorithms. Furthermore, the convolutional neural network (CNN) is a type of DNN having excellent feature learning ability and owes to the characteristic of the hidden layer containing the convolutional layers and pooling layers [18], [19]. There are many classical CNNs, such as LeNet-5 [20], AlexNet [21], GoogleNet [22], ResNet [23], etc. And the convolutional filters learned by the stochastic gradient descent (SGD) and backpropagation (BP) methods are the key to CNN. However, more layers of convolutional operations are needed to extract deeper features, which causes many problems, such as the increase of complexity of the algorithm and the computation time.

Thus, the feedforward learning network which learns the parameters only relies on the training data unsupervisedly is proposed to address the above problems. And there are mainly three types of feedforward learning networks. The methods of the first type are the simplest whose main idea is to use predefined filters instead of learning filters from the data. ScatNet [24] is a representative method of this type. The wavelet filters are employed in ScatNet to execute the process. However, this kind of learn-free method is difficult to handle complex tasks. The second kind of method is to learn the parameters layer by layer. And DBN [16] whose main idea is stacking multiple layers of the same units is one representative of this type. The main idea of the third type of method is the convolution operation. Compared with the traditional CNN, the filters of these methods are learned only from the training data layer by layer. PCANet [25] is the typical method of this type. And the filters of PCANet are learned the by principal component analysis (PCA) [26], [27] algorithm from the patches of the training data.

The main idea of PCANet is to employ the PCA algorithm to replace the SGD and BP methods to calculate the convolutional filters. The original two-layer PCANet is very simple that consists of two convolutional stages, a nonlinear layer, and a feature pooling layer. PCANet runs faster and needs less memory compared with traditional CNN due to the simplicity of PCA itself and the avoidance of vast iterations.

Many improved algorithms based on PCANet have been proposed. LDANet [25] is proposed in the paper of PCANet, which uses the LDA algorithm [28] to replace the original PCA algorithm. It is supposed to achieve better performance as the LDA algorithm takes the label information into consideration, but it fails to promote a lot on some datasets. DALNet [29] employing the discriminative locality alignment (DLA) algorithm [30] to learn convolutional filters attains better performance as DLA can handle the nonlinearity of the distribution of samples and exert the discriminative information better. Furthermore, Yuan et al. proposed RPCANet [31] whose main idea is to use a robust PCA algorithm to learn more representative filters. And RNPCANet [32] uses an explicit kernel PCA to learn convolutional filter banks, which also inherit the ability to handle nonlinear data of kernel PCA. PLDANet [33] is trying to combine the PCA filters and LDA filters as they thought that the noises may interfere with the LDA learning process. MMPCANet [34] is a multi-scale multi-feature fusion PCANet which is used for occluded face recognition.

In fact, PCANet still has a lot of room for improvement. Our motivation mainly comes from two perspectives, one is that the PCA algorithm is a kind of global dimensional reduction from the perspective of manifold learning [35], and the other is the feature extracted from different layers in the CNN structure is different. In the manifold learning field, the PCA algorithm is aimed at preserving the global structure by minimizing the global reconstruction error. i.e., without considering the local relationship between different data points, PCA cannot preserve the intrinsic geometrical structure of the dataset [36], [37]. As is known, the feature extracted from shallow-level convolution layers is mainly low-level characteristics, such as the gradient orientation, edges, color, and so forth [38]. However, this kind of low-level characteristic is meaningful in the recognition of simple content images.

To address the above problems, we propose a new deep learning network named global and local structure network (GLSNet). In GLSNet, we first use a PCA convolution layer to remove the noises of the original images while retaining the principal components. Then, the second layer consists of two convolutions. One is a neighborhood preserving embedding (NPE) [39] convolution layer to preserve the local information of the dataset, and the other is a PCA convolution layer to extract more abstract features. Notably, we take the first PCA convolutional feature, the second NPE convolutional feature, and the second PCA convolutional feature all into consideration. These three features are concatenated as a united feature. Same as PCANet, we use binary quantization as the nonlinear layer. As for the pooling layer, a spatial pyramid pooling layer [40], [41], [42] is used for better feature extraction.

The rest of the paper is organized as follows: in Section II, review NPE and spatial pyramid pooling (SPP) algorithms. In section III, we introduce the proposed GLSNet. In Section IV, we evaluate GLSNet by making experiments on the image datasets compared with PCANet and some state-of-the-art methods. The conclusion and future work are given in Section V.

II. RELATED WORKS

A. NEIGHBORHOOD PRESERVING EMBEDDING

Neighborhood Preserving Embedding (NPE) is a manifold learning method that aims at preserving the local manifold structure. Supposing that there are l data points $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l$ in $\mathbb{R}^n$ space, which denotes as $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_l]$. NPE is to find a transformation matrix $\boldsymbol{A}$ that maps $\boldsymbol{X}$ to $\boldsymbol{Y} = \left[\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_l\right]$ in $\mathbb{R}^d$ ($d \ll n$) where $\boldsymbol{y}_i = \boldsymbol{A}^T\boldsymbol{x}_i$. And it works better especially when $\boldsymbol{X}$ belongs to a nonlinear manifold embedding in $\mathbb{R}^n$.

The process of the NPE algorithm proceeds as follows:

1) CONSTRUCT AN ADJACENCY GRAPH

The adjacency graph is denoted as $\boldsymbol{G}$ that contains m nodes and the element $\boldsymbol{G}_{ij}$ represents whether the node i is adjacent to the node j. There are two algorithms to construct $\boldsymbol{G}$.

a) k nearest neighbors (KNN): node i and node j are marked as adjacent if the node i is within the k nearest neighbors of the node j.
b) ε-neighborhood: node i and node j are marked as adjacent if $\| \boldsymbol{x}_i - \boldsymbol{x}_j \| \leq \varepsilon$.

As the ε-neighborhood is sensitive to the choice of the ε and is hard to choose a good value, thus KNN is generally chosen.

2) COMPUTE THE WEIGHTS

The weight matrix $\boldsymbol{W}$ whose element $\boldsymbol{w}_{ij}$ denotes the weight of the edge from node i to node j, and 0 if they are not adjacent. The weight matrix $\boldsymbol{W}$ is computed by minimizing the following object function:

$$\sum_i ||\boldsymbol{X}_\mathrm{i} - \sum_j \boldsymbol{W}_{ij}\boldsymbol{X}_j||_2^2 \tag{1}$$

With constraint

$$\sum_j \boldsymbol{W}_{ij} = 1 \tag{2}$$

This problem can be solved by the Lagrange multiplier algorithm, see reference [43].

This step is trying to put each data point and its nearest neighbors into a locally linear patch of the manifold. And the weight matrix $\boldsymbol{W}$ is used to represent the geometry relationship of the patches and reconstruct each data point.

3) COMPUTE THE PROJECTION

Supposing that $\boldsymbol{y_i}$ denotes the optimal projection of the data point $\boldsymbol{x_i}$ from high-dimensional space $\mathbb{R}^n$ to low-dimensional space $\mathbb{R}^d$ and it is computed by a linear transformation.

$$\boldsymbol{y}_i = \boldsymbol{A}^T\boldsymbol{x}_i \tag{3}$$

where $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_l]$. The following reconstruction function from high dimension to low dimension is being minimized to compute the linear transformation matrix $\boldsymbol{A}$:

$$\sum_i ||\boldsymbol{Y}_i - \sum_j \boldsymbol{W}_{ij}\boldsymbol{Y}_j||_2^2 \tag{4}$$

where $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_l]$ and constraint $\boldsymbol{Y}^T\boldsymbol{Y} = 1$ is usually imposed (4) to remove an arbitrary scaling factor in the projection.

Above problem can be expressed as follows:

$$\underset{\boldsymbol{a}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{a}\,=\,1}{\arg\min} \ \boldsymbol{a}^T\boldsymbol{X}\boldsymbol{M}\boldsymbol{X}^T\boldsymbol{a} \tag{5}$$

where $\boldsymbol{a}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{a} = 1$ is a constraint, $\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{W})^T(\boldsymbol{I} - \boldsymbol{W})$ and $\boldsymbol{I} = diag(1, \ldots, 1)$. This minimization problem can be easily converted to a generalized eigenvalue problem (GEVP):

$$\boldsymbol{X}\boldsymbol{M}\boldsymbol{X}^T\boldsymbol{a} = \lambda\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{a} \tag{6}$$

Once the solution of (6) is obtained, the low-dimensional projection $\boldsymbol{Y}$ can be computed by (3).

B. SPP

SPP is a popular pooling strategy that performs especially better on the object data set. And spatial pyramid machines (SPM) referring to SPP and the bag-of-words (BOW) model [44], [45], [46] are also effective and popular algorithms in the field of computer vision.

The aim the of pooling layer is to reduce the spatial size of the features to decrease the complexity of the model and training time. The main idea of SPP is to pool the response of each filter of the convolutional layer to different spatial bins. The outputs of SPP are a series of fixed-dimensional vectors to address the problem that fully-connected layers only accept fixed-dimensional vectors.

The advantage of SPP is apparent. First, SPP makes it possible to handle different sizes of images without editing the images. Typically, images exhibit non-uniform size, necessitating warping or cropping of the images to conform to desired dimensions. But warping operations may lose the information of the raw image, and cropping may cause changes of the image geometry. Second, SPP extracts features at different scales, capturing different spatial information. Third, SPP employs multi-level spatial bins instead of a single-window size to pool features.

III. GLOBAL AND LOCAL STRUCTURE NETWORK

A. THE STRUCTURE OF GLSNET

The structure of the proposed GLSNet can be divided into three stages: the first stage (a PCA convolutional layer), the second stage consists of a PCA convolutional layer and a NPE convolutional layer, and an output stage consists of a hashing and histogram operation. Especially, at the pooling stage, the first PCA convolutional features, the second PCA convolutional features, and the second NPE convolutional features are concatenated as a united feature for the next stage. Furthermore, SPP is added for better feature extraction. The diagram of the proposed GLSNet is shown in Fig. 1.

B. THE FIRST STAGE (PCA)

For the training process of the first stage, there are four main steps:

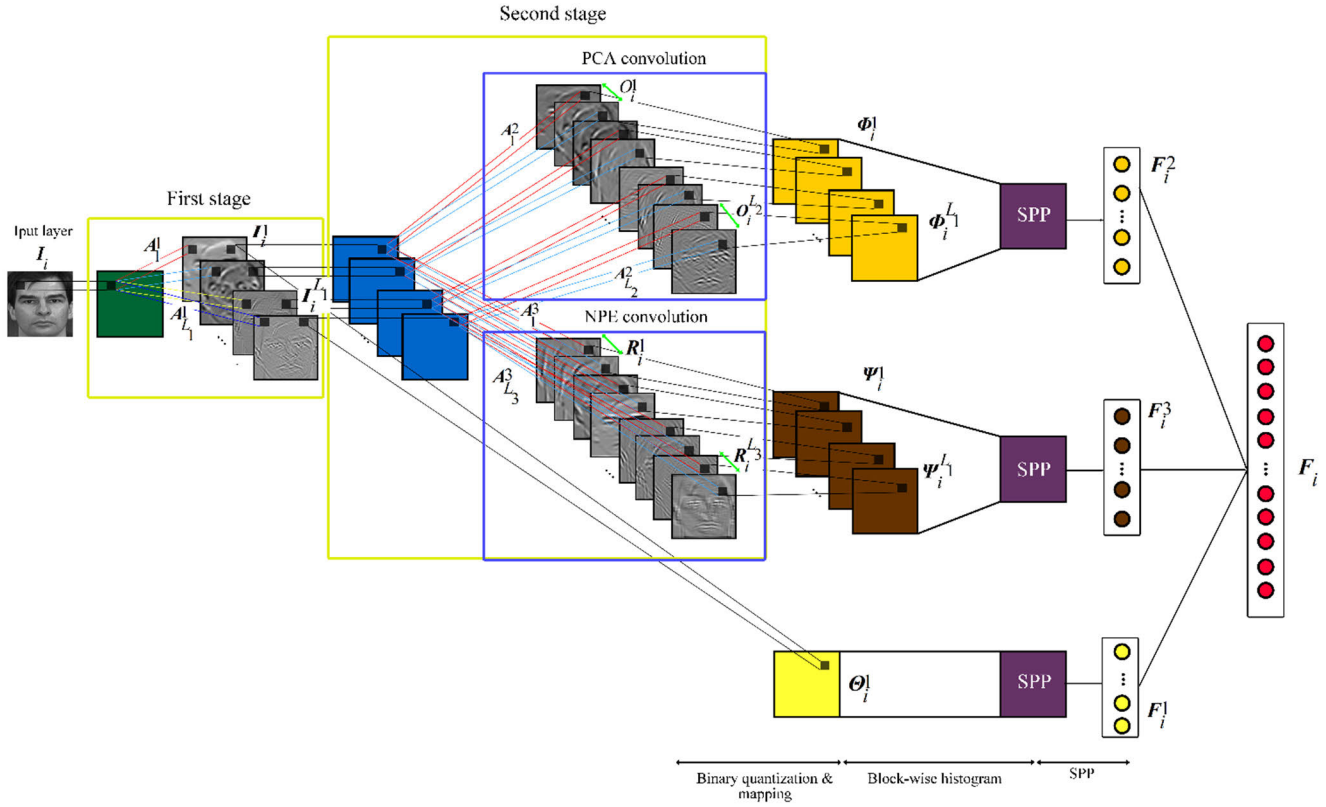

FIGURE 1. The diagram of the proposed GLSNet.

1) TAKE PATCHES FROM THE RAW IMAGES

Supposing that there are N raw images sizing $m \times n$ denotes $\{I_i\}_{i=1}^{N}$. For each pixel of the raw images, a $k_1 \times k_2$ patch is taken, then the patches are vectorized and expressed as $\{\boldsymbol{x}_{i,j}\}_{j=1}^{mn}$, where $m = m - \lceil k_1/2 \rceil$, $n = n - \lceil k_2/2 \rceil$ and $\lceil z \rceil$ means the round down operation.

2) PATCH MEAN REMOVE

To remove the common parts and highlight differences, the patch mean is subtracted from every patch. And the mean-removed patches denote $\{\boldsymbol{x}_{i,j}\}_{j=1}^{mn}$. The mean-remove operation can be easily proceeded by the formula $x_{i,j} = x_{i,j} - \frac{\mathbf{1}^T x_{i,j}}{k_1 k_2}\mathbf{1}$, where $\mathbf{1}$ is an all-one vector of proper dimension.

3) COMBINE THE INPUT IMAGES

At this step, the input images are obtained. The i-th image is combined by its all mean-removed patches, which denotes $\overline{\boldsymbol{X}}_i = \left[\overline{\boldsymbol{x}}_{i,1}, \overline{\boldsymbol{x}}_{i,1}, \ldots, \overline{\boldsymbol{x}}_{i,mn}\right]$. Then, the whole input matrix is constructed with all the input images:

$$\boldsymbol{X} = \left[\overline{\boldsymbol{X}}_1, \overline{\boldsymbol{X}}_2, \ldots, \overline{\boldsymbol{X}}_N\right] \in \mathbb{R}^{k_1 k_2 \times Nmn} \tag{7}$$

4) OBTAIN CONVOLUTIONAL FILTERS BY PCA

The main idea of PCA is to make the transformed data obtain maximal variance by basis transformation. And it also can be considered as minimization the reconstruction error within a family of orthonormal vectors. It can be expressed as the follows:

$$\min_{\boldsymbol{V} \in \mathbb{R}^{k_1 k_2 \times L_1}} \left\| \boldsymbol{X} - \boldsymbol{A}\boldsymbol{A}^T\boldsymbol{X} \right\|_F^2 \quad \text{s.t.} \quad \boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}_{L_1} \tag{8}$$

Equation (8) can be easily converted into the following expression:

$$\arg\min_{\boldsymbol{A}} tr(\boldsymbol{A}^T\boldsymbol{X}\boldsymbol{X}\boldsymbol{A}) \quad \text{s.t.} \quad \boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}_{L_1} \tag{9}$$

Then, (9) can be converted to following equation the by Lagrange multiplier method:

$$\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{A} = \lambda\boldsymbol{A} \tag{10}$$

Thus, the PCA filters can be computed by the following formula:

$$\boldsymbol{A}_l^1 = mat_{k_1,k_2}(\boldsymbol{q}_l(\boldsymbol{X}\boldsymbol{X}^T)) \in \mathbb{R}^{k_1 \times k_2}, l = 1, 2, \ldots, L_1 \tag{11}$$

where $mat_{k1,k2}(\boldsymbol{v})$ is a function that maps $\boldsymbol{v} \in \mathbb{R}^{k_1 \times k_2}$ to a matrix $\boldsymbol{A} \in \mathbb{R}^{k_1 \times k_2}$, and $\boldsymbol{q}_l(\boldsymbol{X}\boldsymbol{X}^T)$ means the l-th principal component of $\boldsymbol{X}\boldsymbol{X}^T$.

5) CONVOLUTION OPERATION

With the PCA filters obtained by (10), the convolutional operation proceeded. And it can be expressed as follows:

$$\boldsymbol{I}_i^1 = \boldsymbol{I}_i * \boldsymbol{A}_l^1, i = 1, 2, \ldots, N \tag{12}$$

where $*$ denotes the 2D convolution operation, and $\boldsymbol{I}_i^1$ means the output of this convolution layer. Besides, to make the size of $\boldsymbol{I}_i^1$ is the same as $\boldsymbol{I}_i$, the boundary of $\boldsymbol{I}_i^1$ is zero-padded before.

C. THE SECOND STAGE (PCA AND NPE)

There are two convolutions, PCA and NPE convolution, at the second stage. The output images of the first PCA convolution layer $\boldsymbol{I}_i^1$ are used as the input images of these two convolutions.

1) PCA CONVOLUTIO

Almost the same as the first PCA stage, the patch-taken operation, and the mean-remove operation proceeded to the input images of the second PCA convolution. And the result of $\boldsymbol{I}_i^1$ after these two operations denotes $\overline{\boldsymbol{Y}}_i^l = \left[\bar{\boldsymbol{y}}_{i,l,1}, \bar{\boldsymbol{y}}_{i,l,j}, \ldots, \bar{\boldsymbol{y}}_{i,l,mn}\right] \in \mathbb{R}^{k_1 k_2 \times mn}$, where $\bar{y}_{i,l,j}$ means the j-th mean-removed patch in $\boldsymbol{I}_i^1$. Similarly, all mean-removed images of the l-th filter are combined as $\boldsymbol{Y}^l = \left[\overline{\boldsymbol{Y}}_1^l, \overline{\boldsymbol{Y}}_1^l, \ldots, \overline{\boldsymbol{Y}}_N^l\right] \in \mathbb{R}^{k_1 k_2 \times Nmn}$. Then, the input matrix concatenated by the result of all filters denotes as follows:

$$\boldsymbol{Y} = \left[\boldsymbol{Y}^1, \boldsymbol{Y}^2, \ldots, \boldsymbol{Y}^{L_1}\right] \in \mathbb{R}^{k_1 k_2 \times L_1 Nmn} \tag{13}$$

Next, the second PCA filters are computed by following equation:

$$\boldsymbol{A}_l^2 = mat_{k_1,k_2}(\boldsymbol{q}_l(\boldsymbol{Y}\boldsymbol{Y}^T)) \in \mathbb{R}^{k_1 \times k_2}, l = 1, 2, \ldots, L_2 \tag{14}$$

Finally, the output images of the input image $\boldsymbol{I}_i^l$ of this convolution are expressed as follows:

$$\boldsymbol{O}_i^l = \boldsymbol{I}_i^l * \boldsymbol{A}_l^2 \tag{15}$$

Additionally, there are L_1 filters at the first stage and L_2 filters, thus, $L_1 L_2$ images are output at this stage.

2) NPE CONVOLUTIO

At this convolution, NPE method is used to learn convolutional filters. The images of the first PCA convolution stage are processed as the second PCA convolution.

Thus, the NPE filters can be obtained by the following equation:

$$\boldsymbol{A}_l^3 = mat_{k_1,k_2}(\mathrm{NPE}_l(\boldsymbol{Y})) \in \mathbb{R}^{k_1 \times k_2}, l = 1, 2, \ldots, L_3 \tag{16}$$

where $\mathrm{NPE}_l(\boldsymbol{v})$ is a function that selects the first l mapping vectors ordered by the magnitude of the eigenvalue.

As the NPE algorithm is a little complex compared with the PCA algorithm, parallel techniques are employed at this stage to boost the computation. Specifically, we use parallel techniques integrated with Matlab software to accelerate the computation.

The convolutional result is also computed by the following equation:

$$\boldsymbol{R}_i^l = \boldsymbol{I}_i^{l*} \boldsymbol{A}_l^3 \tag{17}$$

As NPE and other NN-base algorithms are sensitive to the selection of the number of k. Thus, in order to give full play to the abilities of NPE, an adaptive neighborhood algorithm is taken. The algorithm can select the optimal neighbor parameter k only depending on the input data, which is based on estimates of intrinsic dimensionality and tangent orientation [43].

D. OUTPUT STAGE (HASHING, HISTOGRAMS, AND SPP)

At this stage, the convolutional results of the three layers are separately processed by following steps. Take the first layer for example.

1) HASING

Each image of the convolutional results is firstly binarized. Then, the binarized images are accumulated by each filter, which converts the $\boldsymbol{I}_i^l, l = 1, 2, \ldots, L_1$ into a single integer-valued "image". And the weights are also added by the priority of the filters.

The process of hashing can be computed by following equation:

$$\boldsymbol{\Theta}_i = \sum_{l=1}^{L_1} 2^{l-1} \mathrm{H}(\boldsymbol{I}_i^l) \tag{18}$$

where $\mathrm{H}(v)$ is a Heaviside step (like) function, whose output is one if the input equals or is bigger than zero, and zero otherwise. It can be defined as:

$$\mathrm{H}(v) = \begin{cases} 0, v < 0 \\ 1, v \geq 0 \end{cases} \tag{19}$$

After the function, every pixel of the images is in the range $\left[0, 2^{L_1} - 1\right]$. And the number of images is reduced to the product of the number of filters in the previous layers. In other words, the additional images produced by the last convolution are fused, and the number of images at this time is the same as that of the previous convolution. i.e., the size of the output images of the second PCA convolution equals NL_1.

As for the second PCA convolution and second NPE convolution, the results are also processed as above.

The second PCA convolution after the hasing operation is expressed as:

$$\boldsymbol{\Phi}_i = \sum_{l=1}^{L_1} 2^{l-1} \mathrm{H}(\boldsymbol{O}_i^l) \tag{20}$$

The second NPE convolution after the hasing operation is expressed as:

$$\boldsymbol{\Psi}_i = \sum_{l=1}^{L_1} 2^{l-1} \mathrm{H}(\boldsymbol{R}_i^l) \tag{21}$$

2) HISTOGRAMS

Each of the result images after the hasing operation is partitioned into B blocks. Then the histogram of the decimal values of each block is computed and it denotes as:

$$\boldsymbol{S}_i^b = \mathrm{LBhist}(\boldsymbol{\Theta}_{ib}), b = 1, 2, \ldots, B. \tag{22}$$

3) SPP

The local histograms of a single image are separately processed by SPP to extract the corresponding feature.

$$\boldsymbol{F}_{i,l}^{1} = \mathrm{SPP}(\boldsymbol{S}_i) \tag{23}$$

where $\boldsymbol{S}_i$ is the concatenation of all $\boldsymbol{S}_i^b$. Furthermore, the maximum function is used in the spatial pooling. Then, the features extracted from all images are concatenated as a single unified feature vector $\boldsymbol{F}_i^1$.

Similarly, the second PCA convolution and second NPE convolution feature are extracted and they are expressed as $\boldsymbol{F}_i^2$ and $\boldsymbol{F}_i^3$.

Finally, these three features are concatenated for recognition:

$$\boldsymbol{F}_i = [\boldsymbol{F}_i^1, \boldsymbol{F}_i^2, \boldsymbol{F}_i^3] \tag{24}$$

IV. EXPERIMENTS

In this Section, we make experiments on six different datasets consisting of four human face datasets, an object dataset, and a handprinted dataset to evaluate the performance of the proposed method on different types of datasets. In order to illustrate the effectiveness of our proposed algorithm, some typical methods such as LBP, Gabor, and the latest modified methods of PCANet such as PLDANet, and MMPCANet.

Additionally, a linear SVM [47], [48], [49] classifier is employed for all datasets.

A. EXPERIMENTAL DATASETS

GeorgiaTech (GT) face dataset [50] contains 50 person face images taken between 06/01/99 and 11/15/99. And there are 15 images of each people. The images contain frontal and tilted faces with different facial expressions, lighting conditions, and scales. In the experiment, the images are resized to 128 × 128 pixels.

CMU-PIE [51] dataset contains the facial images of 68 people. With the CMU 3d Room, there are 13 different poses, under 43 different illumination conditions, and with 4 different expressions of each person.

AR [52] dataset was created by Aleix Martinez and Robert Benavente in the Computer Vision Center (CVC). And the dataset consists of over 4000 images corresponding to 100 people's faces (50 men and 50 women). These images have different facial expressions, illumination conditions, and occlusions.

FERET [53] is widely used in face recognition, which was collected in 15 sessions. And there are up to 14126 images consisting of 365 individuals and 365 duplicate sets of images. Similarly, the images vary in different light conditions and expressions.

COIL100 [54] is an object dataset containing 7200 images of 100 objects. The objects are different in geometric characteristics, texture, and reflection light characteristics. These images are by a fixed camera, and the camera takes a photo when the object rotates 5 degrees. Thus, there are 72 images of each object.

NIST [55] handprinted forms and characters dataset (handprinted) is collected from 3600 writers. And the character contains digits, upper and lower case, and free text fields. As the characters are not complex, the images are resized into 28 × 28.

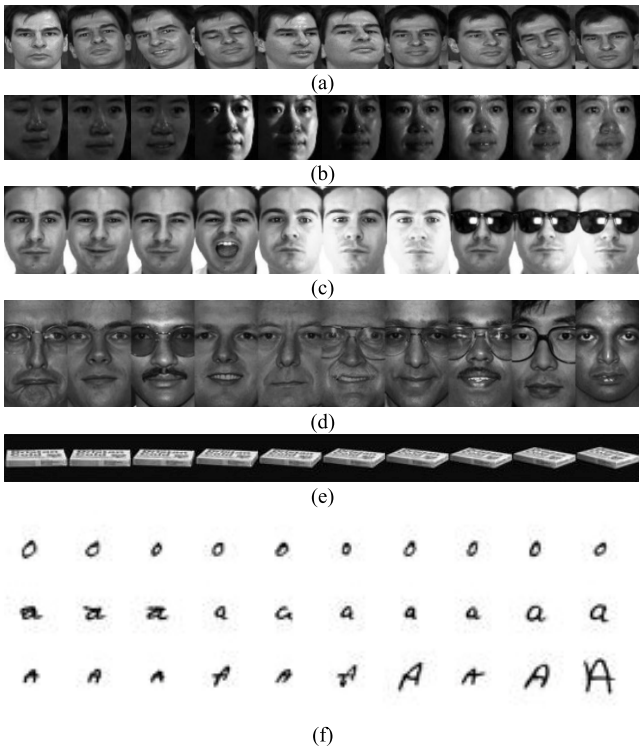

FIGURE 2. Some sample images in the experiments. (a) GT face dataset (b) CMU-PIE face dataset (c) AR face dataset (d) FERET face dataset (e) COIL100 object dataset (f) NIST handprinted forms and characters dataset.

Some samples of the images are shown in Fig. 2. And the details and parameter settings of the images are tabulated in Table. 1.

B. PARAMETER SETTINGS

In the following experiments, the parameters are set as follows unless otherwise specified. For GLSNet, PCANet, and other PCANet-base algorithms, the filter patch size is all set to be 7 × 7, and the filter numbers are all set to be 8. Furthermore, the histogram's block size is 7 × 7 and the block overlap ratio is 0.5.

As for GLSNet and MMPCANet, the spatial pyramid has three levels, and the dimensions of each level are 1, 2, and 4, respectively.

C. EXPERIMENTS ON FACE DATASETS

Face recognition is always a vital task in the image recognition field. Thus, four face datasets are selected for different purposes.

1) EXPERIMENTS ON THE GT FACE DATASET

GT face dataset contains 750 images of 50 people, and the images are originally 640 × 480 pixels with the clustered backgrounds. The average size of the faces in these images is 150×150 pixels. In our experiments, the images are converted to gray and then cropped and resized to 128 × 128.

TABLE 1. Details and experimental settings of the datasets.

Datasets	Number of class used	Number of images used	Cropped and resized image sizes	purposes
GT	50	750	128×128	Test filter size and number
CMU-PIE	68	11560	64×64	Train the generic face filters
AR	100	2600	165×120	Test the generic face filters
FERET	1196	3541	150×90	Test the generic face filters
COIL100	100	7200	40×50	Test the object recognition
handprinted	62	248000	28×28	Test the handprinted recognition

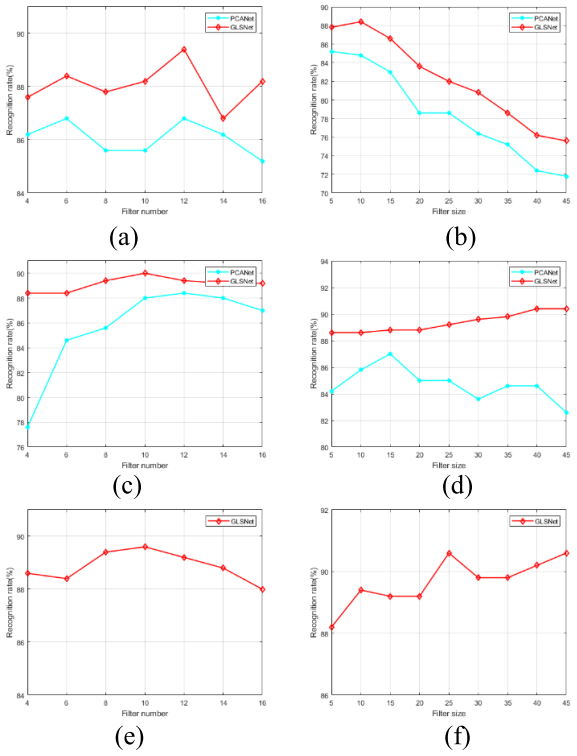

FIGURE 3. The recognition rates versus the filter size and filter number. (a) The recognition rates versus the first PCA filter number. (b) The recognition rates versus the first PCA filter size. (c) The recognition rates versus the second PCA filter number. (d) The recognition rates versus the second PCA filter size. (e) The recognition rates versus the second NPE filter number. (f) The recognition rates versus the second NPE filter size.

Since the size of the GT dataset is not very sufficient, this dataset is mainly used for testing filter size and number. And the first 4, 6, 8, and 10 images of each person were used for training and the rest for testing. We compared PCANet and GLSNet, the filter number increased from 4 to 18 by the step of 2. And the filter size is taken from 5 to 45 every 5 intervals. As the GLSNet has three convolutions when one convolution varies and others keep fixed the original parameters. The results are shown in Fig. 3.

Furthermore, we select different numbers (4, 6, 8, 10) of samples for training and the rest for testing with default parameter settings, and the results are listed in Table. 2.

2) EXPERIMENTS ON THE CMU-PIE FACE DATASET

CMU-PIE dataset has many samples with different poses, illumination, and expressions. In our experiment, the images

TABLE 2. Recognition accuracy (in percent) with varying numbers of training samples on the GT face dataset.

Method	4	6	8	10
CNN-2	2.73	1.33	2.00	2.40
LBP	33.45	42.00	51.14	56.00
Gabor	68.91	77.56	84.86	87.60
LDANet-2	82.91	90.22	90.86	95.60
PCANet-2	82.36	89.36	91.43	96.40
PLDANet	72.36	50.67	82.86	83.60
MMPCANet	82.73	90.22	92.86	95.60
GLSNet	**86.91**	**93.20**	**94.00**	**97.60**

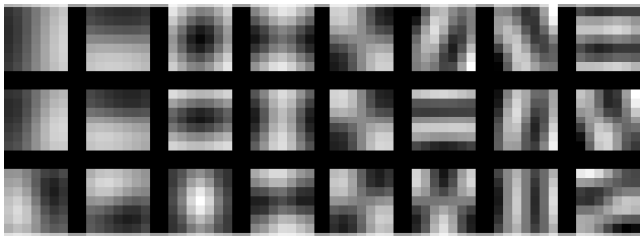

FIGURE 4. Filters learned by GLSNet from CMU-PIE face dataset. The first row is the first PCA filter, the second row is the second PCA filter, and the third row is the second NPE filter.

are cropped and resized into 64 × 64. Due to variations in poses among individuals, a subset of 170 images featuring 68 distinct persons was selected for analysis.

The dataset is used as a generic face training set that learns a generic filter bank. As the dataset contains enough faces under different poses, illumination conditions, and expressions, the CMU-PIE filter is applied to other face datasets in order to test the generalization ability of feature extraction of the model. And the filters learned are shown in Fig. 4.

Furthermore, the dataset is divided into three subsets by conditions, expression, illuminations, and posture. Expression, illumination, and posture subset separately contains 10, 50, and 40 samples of each person. These subsets are separately trained and tested with the training size of 3, 8, and 6. And Table. 3 shows the test accuracy.

According to the experiments, the proposed GLSNet performs best. Compared with the original PCANet, GLSNet promotes 0.37%, 0.24%, and 2.47%. Furthermore, compared with other modified methods of PCANet and LDANet, GLSNet also obtains the highest recognition rate.

3) EXPERIMENTS ON THE AR FACE DATASET

Similarly, to evaluate the generalization ability, the filters learned from the CMU-PIE dataset are used on the AR dataset

TABLE 3. Recognition accuracy (in percent) on the CMU-PIe face dataset.

Method	exps	illum	pose	average
CNN-2	47.79	63.74	55.56	55.70
LBP	88.97	84.47	76.21	83.22
Gabor	95.59	95.85	93.38	94.94
LDANet-2	90.44	92.08	45.64	76.05
PCANet-2	99.26	98.49	94.64	97.46
PLDANet	97.06	93.64	84.19	91.63
MMPCANet	99.63	95.33	97.01	97.32
GLSNet	**99.63**	**98.73**	**97.11**	**98.32**

TABLE 4. Recognition accuracy (in percent) on the AR face dataset.

Method	Disguise	Exps	Illum	Disguise &Illum
CNN-2	26.00	60.00	77.33	42.25
LBP	61.00	76.60	94.33	60.75
Gabor	65.50	75.20	94.67	**68.75**
LDANet-2	53.50	72.60	90.33	50.50
PCANet-2	54.50	72.40	90.00	52.50
PLDANet	58.50	72.80	87.00	52.75
MMPCANet	62.00	79.40	91.33	55.00
GLSNet	**66.00**	**82.20**	**96.00**	57.00

to extract features. Besides, all modified methods based on PCANet and LDANet also use the filters learned from the CMU-PIE dataset.

AR dataset consists of 50 males and 50 females, and each person has 26 images. In our experiments, these images are cropped from the background, resized into 165 × 120, and converted to gray. Since the AR dataset contains a set of disguise images, i.e., wearing sunglasses and the scarf. This dataset is used for testing the ability to recognize occluded faces.

The dataset is also divided into four subsets, disguise set, expression set, illumination set, and disguise & illumination set. On each subset, there are 4, 8, 6, and 8 images of each person, and the first 2, 3, 3, and 4 images are used for training.

The recognition results are given in Table. 4.

As is shown in Table. 4, GLSNet is much ahead of most other methods. Compared with PCANet, GLSNet obtain much higher recognition rate, which proves that GLSNet has greater generalization ability. The promotion of the ability is mainly relaid on the multi-feature structure of GLSNet. PCANet extracts the features by considering minimizing the reconstruction of the global error, thus, some local information may lose during this operation. But the NPE convolution mainly pays attention to the local structure, which makes the features extracted mainly represent the local structure. Thus, the united features extracted from GLSNet can obtain higher recognition rate.

4) EXPERIMENTS ON THE FERET FACE DATASET

Finally, we make the experiment on the FERET face dataset, and the filters learned from CMU-PIE are also applied. The dataset contains 1564 sets of images including 365 individuals. The images are cropped to 150 × 90 and converted to gray. Furthermore, the dataset is divided into four categories: Fb, with different expressions; Fc, with different illumination conditions; Dup-I: with a longer taken a period of three to four months; Dup-II: with a longer taken a period of one and a half years.

TABLE 5. Recognition accuracy (in percent) on the FERET face dataset.

Method	Fb	Fc	Dup-I	Dup-II
CNN-2	58.82	8.24	37.39	28.20
LBP	76.32	72.68	68.42	63.25
Gabor	95.06	97.42	78.67	72.65
LDANet-2	92.64	95.79	68.89	72.17
PCANet-2	92.64	96.84	70.83	72.17
PLDANet	92.30	94.74	65.42	69.57
MMPCANet	99.41	90.72	84.76	83.76
GLSNet	**99.67**	**99.48**	**90.03**	**87.61**

FIGURE 5. Some occluded samples of the COIL100 dataset.

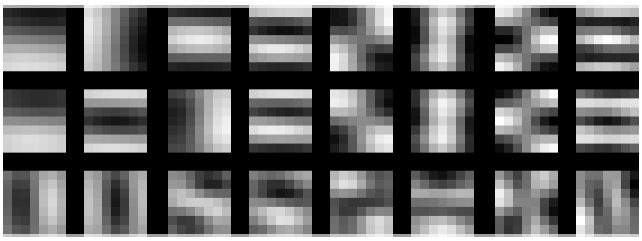

FIGURE 6. Filters learned by GLSNet from COIL100. The first row is the first PCA filter, the second row is the second PCA filter, and the third row is the second NPE filter.

Besides, the results of GLSNet and other algorithms are listed in Table. 5.

According to Table. 5, GLSNet also obtains the best performance. Especially on the Fb and Dup-I subsets, the recognition of GLSNet achieves 99.67% and 90.03%, which is 7.03% and 19.2% higher than PCANet. This proves the generalization ability of GLSNet again.

D. EXPERIMENTS ON OBJECT AND TEXT DATASETS

In this section, we test the effectiveness of the proposed GLSNet on object and text datasets.

1) EXPERIMENTS ON THE COIL100 DATASET

The COIL100 dataset is widely used for object recognition as the set contains a variety of object types and the taken angle is from 0 to 360 degrees. In our experiments, the gray images are cropped to 40 × 55 pixels. There are 100 objects and 72 images of each object. Besides, the first 30 images are used for training the filters, and the rest of the images for testing.

Furthermore, we also test the occluded condition on the COIL100 dataset. On the bottom of each image, a black occluded part filled is added. The occluded rate increased from 0 to 40% with the step of 10% as the object. Some occluded images are shown in Fig. 5. And the filters learned from the COIL100 dataset are displayed in Fig. 6.

The recognition results on COIL100 are given in Table. 6.

As the results list in Table 6, GLSNet also obtains the best performance in most conditions. The modified methods based on PCANet perform not so well and the recognition

TABLE 6. Recognition accuracy (in percent) with different occluded rate on the COIL100 dataset.

Method	0	0.1	0.2	0.3	0.4
CNN-2	70.50	2.71	1.73	2.26	2.45
LBP	63.88	55.55	42.31	31.86	20.14
Gabor	78.52	52.58	50.07	45.70	42.61
LDANet-2	88.12	85.90	79.12	**77.21**	17.88
PCANet-2	88.52	86.07	82.21	67.62	40.18
PLDANet	88.62	86.45	81.90	74.10	32.76
MMPCANet	86.24	84.45	78.83	65.24	48.88
GLSNet	**91.57**	**87.64**	**84.62**	72.93	**49.07**

TABLE 7. Details of the subset of the handprinted dataset.

Subset	Sample number	Train sample number	Test sample number
Digit	5000	500	4500
Lower case	5000	500	4500
Upper case	4000	500	3500
Digit & Lower case	5000	500	4500
Digit & Upper case	4000	500	3500

TABLE 8. Recognition accuracy (in percent) on the handprinted dataset.

Method	Digit	Lower case	Upper case	Digit & Lower case	Digit & Upper case
CNN-2	96.51	88.02	85.21	78.35	87.36
LBP	74.21	52.98	52.98	48.21	55.28
Gabor	96.04	47.83	84.46	76.35	85.47
LDANet-2	97.60	89.32	85.47	83.55	91.20
PCANet-2	95.95	88.43	85.26	81.63	88.42
PLDANet	**97.91**	**91.91**	86.92	**85.41**	90.82
MMPCANet	93.75	85.37	81.79	75.23	83.10
GLSNet	96.67	90.09	**87.02**	84.86	**91.91**

rate decreases a lot when the main part of the object body is occluded. But GLSNet alleviates the rate of decline. For example, GLSNet decreased by 3.93%, 3.02%, 11.69%, and 23.86%, while PCANet decreased by 2.45%, 3.86%, 14.59%, and 27.44% with the increase of occlusion rate.

2) EXPERIMENTS ON THE HANDPRINTED DATASET

We select a more complex and diverse handprinted dataset, NIST Handprinted Forms and Characters Database. The dataset originally contains 3600 writers, 810000 character images isolated from their forms, and ground truth classifications for those images. In our experiment, the images are resized into 28×28 and the dataset is divided into 6 subsets, digit, lower case, upper case, digit & lower case, and digit & upper case. And the details of the subsets are listed in Table. 7.

The experiment results on the above subsets are tabulated in Table 8.

On the handprinted dataset, compared with the original PCANet, GLSNet also promotes 0.72%, 1.66%, 1.76%, 3.23%, and 3.49% on 5 subsets. But on such a dataset with simple content, label information played a crucial role. Thus, LDANet and PLDANet algorithms taking advantage of the label information obtain higher recognition rates on some of the subsets.

TABLE 9. Recognition accuracy (in percent) with varying numbers of training samples on the MNIST dataset.

Method	20	100	1000	1000
PCANet2	53.78	78.23	95.89	98.46
PCANet12	57.30	80.61	96.26	98.67
NPE-PCANet	60.02	82.20	96.48	98.69
GLSNet	**61.78**	**82.73**	**96.67**	**98.78**

E. ABLATION STUDY

To evaluate our proposed network and the effectiveness of each component in the network. The ablation study proceeds in this section. In detail, we compared 4 different models: the second PCA convolutional feature (PCANet2), the first and the second PCA convolutional feature (PCANet12), the second PCA and the NPE convolutional features (NPE-PCANet), global and local structure network (GLSNet).

In the ablation study, the MNIST dataset [20] with different training sample sizes is used to evaluate the above models. The parameters are also kept the same as before. And the result is shown in Table. 9.

As is shown in Table. 9, compared with the original PCANet, other models obtain different degrees of improvement. For example, PCANet12 promotes 3.52%, NPE-PCANet promotes 6.24%, and GLSNet promotes 8% when the training size is 20. The experiment results also confirm our thought. PCANet12 promotes mainly because the feature not only contains abstract information extracted from high layers but also contains low-level information such as orientation, edges, color, and so forth. And the recognition rate of NPE-PCANet is much higher than the original PCANet and PCANet12, which is mainly because the local information of the space structure does play an important role in the recognition. And with both low-level information of the PCANet12 and the local information of the space structure, GLSNet performs better than other models.

F. ANALYSIS

From the above experiment, it can be concluded that: 1) On all datasets, our proposed GLSNet obtain higher recognition rates than the original PCANet and most modified algorithm of PCANet and LDANet. In detail, compared with the original PCANet, GLSNet promotes 1.2% to 4.55% on the GT face dataset, 0.24% to 2.47% on the CMU-PIE face dataset, 4.5% to 11.5% on the AR face dataset, 2.64% to 19.2% on FERET face dataset, 1.57% to 8.89% on occluded COIL100 object dataset, and 0.72% to 3.49% on handprinted datasets. Furthermore, the performance of the GLSNet is also better than the state-of-art based on PCANet. 2) Based on the results of the AR face dataset and FERET face dataset, we can conclude that GLSNet promotes the generalization ability a lot. And this promotion is mostly due to the extra features extracted from the NPE convolution learning features based on the local structure of the dataset and the first PCA convolution learning the sallow features. 3) As is known, CNNs like PCANet perform not so well on occluded images. But GLSNet mitigates the declining

trend to a certain extent. i.e., on the occluded COIL100 dataset, GLSNet decreased by 3.93%, 3.02%, 11.69%, and 23.86%, while PCANet decreased by 2.45%, 3.86%, 14.59%, and 27.44% with the increase of occlusion rate. 4) On the handprinted dataset whose texture is simple and with little content, PCANet performs not so well compared with LDANet due to the lack of label information. However, our proposed algorithm GLSNet increases the recognition rate a lot, and it even exceeds the modified method of LDANet, PLDANet, on some subsets of handprinted datasets. 5) In the ablation experiment, the results confirmed our idea that local filter and shallow filter information enhanced network performance to varying degrees. And local filters perform much better than shallow filters, which is why we named the network GLSNet.

V. CONCLUSION

In this paper, a novel deep learning network called global and local structure network (GLSNet) is proposed to address the drawbacks of the original PCANet. The structure of GLSNet is mainly divided into 3 stages: input stage, first convolutional stage, second convolutional stage, and output stage. Especially, the first convolutional filters are learned by PCA to remove the noises of the original images. The second convolutional layer consists of two convolutions: PCA convolution and NPE convolution. The second PCA convolution learns the filters by considering the global construction error and the NPE convolution learns the filters by considering the local structure. Besides, the convolutional result of the first PCA convolution, the second PCA convolution, and the second NPE convolution are sent into SPP to extract more representative features.

There are still some works that can be further studied. Firstly, the filter learning algorithms (PCA, and NPE) used in our model are the 1-D version, which means that the input data of the algorithms is combined with the vectorized data points. Thus, the algorithms can be replaced by the 2-D version, i.e., 2D-PCA [56], and 2D-NPE [57]. Studies have shown that the 2-D version provides a significant performance boost over 1-D. Secondly, the label information is not used since LDA performs not well in convolution learning, but other algorithms that work with label information such as DLA and supervised neighborhood preserving embedding (SNPE) [58] can be added to the structure of the network to obtain better performance.

REFERENCES

[1] M. Nixon and A. S. Aguado, *Feature Extraction and Image Processing*. Amsterdam, The Netherlands: Elsevier, 2020.

[2] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, Nov. 2018, doi: 10.1145/3136625.

[3] A. L. Ramadhani, P. Musa, and E. P. Wibowo, "Human face recognition application using PCA and eigenface approach," in *Proc. 2nd Int. Conf. Informat. Comput. (ICIC)*, Nov. 2017, pp. 1–5.

[4] Z. Cao, L. Duan, G. Yang, T. Yue, and Q. Chen, "An experimental study on breast lesion detection and classification from ultrasound images using deep learning architectures," *BMC Med. Imag.*, vol. 19, no. 1, p. 51, Dec. 2019, doi: 10.1186/s12880-019-0349-x.

[5] G. Kumar and P. K. Bhatia, "A detailed review of feature extraction in image processing systems," in *Proc. 4th Int. Conf. Adv. Comput. Commun. Technol.*, Rohtak, India, Feb. 2014, pp. 5–12.

[6] Y. Qi, M. Qiu, H. Jiang, and F. Wang, "Extracting fingerprint features using autoencoder networks for gender classification," *Appl. Sci.*, vol. 12, no. 19, p. 10152, Oct. 2022, doi: 10.3390/app121910152.

[7] F. Yuan, K. Li, C. Wang, J. Shi, and Y. Zhu, "Fully extracting feature correlation between and within stages for semantic segmentation," *Digit. Signal Process.*, vol. 127, Jul. 2022, Art. no. 103578, doi: 10.1016/j.dsp.2022.103578.

[8] J. Wu, Z. Cui, V. S. Sheng, P. Zhao, D. Su, and S. Gong, "A comparative study of SIFT and its variants," *Meas. Sci. Rev.*, vol. 13, no. 3, pp. 122–131, 2013, doi: 10.2478/msr-2013-0021.

[9] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato, "SIFT features tracking for video stabilization," in *Proc. 14th Int. Conf. Image Anal. Process.*, Modena, Italy, Sep. 2007, pp. 825–830.

[10] G. Zhang, X. Huang, S. Z. Li, Y. Wang, and X. Wu, "Boosting local binary pattern (LBP)-based face recognition," in *Sinobiometrics*, in Lecture Notes in Computer Science, D. Hutchison, Eds., Berlin, Germany: Springer, 2005, pp. 179–186.

[11] A. Satpathy, X. Jiang, and H.-L. Eng, "LBP-based edge-texture features for object recognition," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 1953–1964, May 2014, doi: 10.1109/TIP.2014.2310123.

[12] O. Déniz, G. Bueno, J. Salido, and F. De la Torre, "Face recognition using histograms of oriented gradients," *Pattern Recognit. Lett.*, vol. 32, no. 12, pp. 1598–1603, 2011, doi: 10.1016/j.patrec.2011.01.004.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, Jun. 2005, pp. 886–893.

[14] L. Shen and L. Bai, "A review on Gabor wavelets for face recognition," *Pattern Anal. Appl.*, vol. 9, nos. 2–3, pp. 273–292, 2006, doi: 10.1007/s10044-006-0033-y.

[15] I. Fogel and D. Sagi, "Gabor filters as texture discriminator," *Biological*, vol. 61, no. 2, pp. 103–113, 1989, doi: 10.1007/BF00204594.

[16] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, May 2006, doi: 10.1162/neco.2006.18.7.1527.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[18] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, and T. Liu, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.

[19] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015, *arXiv:1511.08458*.

[20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 2, pp. 84–90, Jun. 2012, doi: 10.1145/3065386.

[22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.

[24] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1872–1886, Aug. 2013, doi: 10.1109/TPAMI.2012.230.

[25] T.-H. Chan, K. Jia, S. Gao, J. Lu, and 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, doi: 10.1109/TIP.2015.2475625.

[26] I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 374, Apr. 2016, Art. no. 20150202, doi: 10.1098/rsta.2015.0202.

[27] A. Mackiewicz and W. Ratajczak, "Principal components analysis (PCA)," *Comput. Geosci.*, vol. 19, pp. 303–342, Mar. 1993, doi: 10.1016/0098-3004(93)90090-R.

[28] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," *AI Commun.*, vol. 30, no. 2, pp. 169–190, 2017, doi: 10.3233/AIC-170729.

[29] Z. Feng, L. Jin, D. Tao, and S. Huang, "DLANet: A manifold-learning-based discriminative feature learning network for scene classification," *Neurocomputing*, vol. 157, pp. 11–21, Jun. 2015, doi: 10.1016/j.neucom.2015.01.043.

[30] T. Zhang, D. Tao, and J. Yang, "Discriminative locality alignment," in *Computer Vision—ECCV 2008* (Lecture Notes in Computer Science), D. Forsyth, P. Torr, and A. Zisserman, Eds. Berlin, Germany: Springer, 2008, pp. 725–738.

[31] Z. Yuan, Q. Wang, and X. Li, "ROBUST PCANet for hyperspectral image change detection," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2018, pp. 4931–4934.

[32] M. Qaraei, S. Abbaasi, and K. Ghiasi-Shirazi, "Randomized nonlinear PCA networks," *Inf. Sci.*, vol. 545, pp. 241–253, Feb. 2021, doi: 10.1016/j.ins.2020.08.005.

[33] C. Zhang, M. Mei, Z. Mei, J. Zhang, A. Deng, and C. Lu, "PLDANet: Reasonable combination of PCA and LDA convolutional networks," *Int. J. Comput. Commun. Control*, vol. 17, no. 2, pp. 1–13, Feb. 2022, doi: 10.15837/ijccc.2022.2.4541.

[34] Z. Wang, Y. Zhang, C. Pan, and Z. Cui, "MMPCANet: An improved PCANet for occluded face recognition," *Appl. Sci.*, vol. 12, no. 6, p. 3144, Mar. 2022, doi: 10.3390/app12063144.

[35] D. Lunga, S. Prasad, M. M. Crawford, and O. Ersoy, "Manifold-learning-based feature extraction for classification of hyperspectral data: A review of advances in manifold learning," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 55–66, Jan. 2014, doi: 10.1109/MSP.2013.2279894.

[36] M. Zhang, Z. Ge, Z. Song, and R. Fu, "Global–local structure analysis model and its application for fault detection and identification," *Ind. Eng. Chem. Res.*, vol. 50, no. 11, pp. 6837–6848, Jun. 2011, doi: 10.1021/ie102564d.

[37] J. Yu, "Local and global principal component analysis for process monitoring," *Process Control*, vol. 22, no. 7, pp. 1358–1373, 2012, doi: 10.1016/j.jprocont.2012.06.008.

[38] D. Bhatt, C. Patel, H. Talsania, J. Patel, R. Vaghela, S. Pandya, K. Modi, and H. Ghayvat, "CNN variants for computer vision: History, architecture, application, challenges and future scope," *Electronics*, vol. 10, no. 20, p. 2470, Oct. 2021, doi: 10.3390/electronics10202470.

[39] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proc. Tenth IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1. Beijing, China, Oct. 2005, pp. 1208–1213.

[40] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1. Beijing, China, 2005, pp. 1458–1465.

[41] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. New York, NY, USA, 2006, pp. 2169–2178.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2014, doi: 10.1109/TPAMI.2015.2389824.

[43] J. Wei, H. Peng, Y.-S. Lin, Z.-M. Huang, and J.-B. Wang, "Adaptive neighborhood selection for manifold learning," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Kunming, China, Jul. 2008, pp. 380–384.

[44] H. M. Wallach, "Topic modeling," in *Proc. 23rd Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, 2006, pp. 977–984.

[45] R. Zhao and K. Mao, "Fuzzy bag-of-words model for document representation," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 794–804, Apr. 2018, doi: 10.1109/TFUZZ.2017.2690222.

[46] W. A. Qader, M. M. Ameen, and B. I. Ahmed, "An overview of bag of words; importance, implementation, applications, and challenges," in *Proc. Int. Eng. Conf. (IEC)*, Erbil, Iraq, Jun. 2019, pp. 200–204.

[47] D. M. Abdullah and A. M. Abdulazeez, "Machine learning applications based on SVM classification a review," *Qubahan Academic J.*, vol. 1, no. 2, pp. 81–90, Apr. 2021, doi: 10.48161/qaj.v1n2a50.

[48] Z. Yin, J. Liu, M. Krueger, and H. Gao, "Introduction of SVM algorithms and recent applications about fault diagnosis and other aspects," in *Proc. IEEE 13th Int. Conf. Ind. Informat. (INDIN)*, Cambridge, U.K., Jul. 2015, pp. 550–555.

[49] V. K. Chauhan, K. Dahiya, and A. Sharma, "Problem formulations and solvers in linear SVM: A review," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 803–855, 2019, doi: 10.1007/s10462-018-9614-6.

[50] V. A. Nefian. (2013). *Georgia Tech Face Database*. [Online]. Available: http://www.anefian.com/research/face_reco.htm

[51] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. 5th IEEE Int. Conf. Autom. Face Gesture Recognit.*, Washington, DC, USA, May 2002, pp. 53–58.

[52] A. Martinez and R. Benavente, "The AR face database: CVC technical report," Universitat Autònoma de Barcelona, Barcelona, Spain, Tech. Rep. 24, 1998.

[53] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image Vis. Comput.*, vol. 16, no. 5, pp. 295–306, 1998, doi: 10.1016/S0262-8856(97)00070-X.

[54] S. Nayar. (1996). *Columbia Object Image Library (COIL100)*. [Online]. Available: http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php

[55] P. J. Grother and P. A. Flanagan, "NIST handprinted forms and characters," NIST, Gaithersburg, MD, USA, Tech. Rep., 19, 1995.

[56] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004, doi: 10.1109/TPAMI.2004.1261097.

[57] H. Du, S. Wang, J. Zhao, and N. Xu, "Two-dimensional neighborhood preserving embedding for face recognition," in *Proc. 2nd IEEE Int. Conf. Inf. Manage. Eng.*, Chengdu, China, Apr. 2010, pp. 500–504.

[58] X. Bao, L. Zhang, B. Wang, and J. Yang, "A supervised neighborhood preserving embedding for face recognition," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Beijing, China, Jul. 2014, pp. 278–284.

JINPING WANG is currently pursuing the master's degree with the College of Computer and Information Science, Chongqing Normal University, Chongqing, China. His research interests include machine learning and computer vision.

RUISHENG RAN received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, and the M.S. degree in computational mathematics and the Ph.D. degree in computer application technology from the University of Electronic Science and Technology of China, Chengdu, China. He is currently a Professor with the College of Computer and Information Science, Chongqing Normal University. His research interests include machine learning and computer vision.

BIN FANG (Senior Member, IEEE) received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, the M.S. degree in electrical engineering from Sichuan University, Chengdu, China, and the Ph.D. degree in electrical engineering from The University of Hong Kong, Hong Kong.

He is currently a Professor with the Department of Computer Science, Chongqing University, Chongqing, China. He has published more than 100 technical articles. His research interests include computer vision, pattern recognition, medical image processing, biometrics applications, and document analysis. He is an Associate Editor of the *International Journal of Pattern Recognition and Artificial Intelligence*.