International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.9 (2015), pp.21-30
http://dx.doi.org/10.14257/ijsip.2015.8.9.03

A Novel Solution to Test Face Recognition Methods on the Training Data Set

Weiwei Wu

Zibo Vocational Institute in Shandong, China
zbvcwww@126.com

Abstract

In modern life, we need better techniques based on biometric features recognition such as face recognition, fingerprint recognition and iris recognition. We present a method which can be used for face recognition or verification applications. The method can solve the problem that when the number of data categories is large and each number of the category used for training is small. As the conventional four stages, face detection, face alignment, face representation and face classification, we propose a Siamese architecture especially for the representation stage and use a one-against-one support vector machine for the classification stage. LFW dataset is used for training and testing which gets a considerable result. And we also test our system on other face dataset, which has a high accuracy on the recognition.

Keyword: *face recognition, face verification, Siamese convolutional neural network, support vector machine*

1. Introduction

Face recognition had been proposed in the 1960s and raised a huge attention to study it. In 1993, the United States Department of Defense started a far-reaching program – FERET (Face Recognition Technology), which aimed at showing the newest progress of face recognition and the main problem it faced. And now, after 2000, the study of it has achieved a new level, we use it to help us with the information security. Take the laptop for example, we need to pass the face recognition before we enter the operating system. Since the electronic commerce is spread all across our daily life, we need a more safe method like the face recognition.

There are five useful methods for face recognition summed to the past study.

First, the method based on the geometrical characteristic, which is first applied to the face recognition problem. Its basic idea is the difference of everyone's face is because of different components of every face, like the eyes, noses, mouths and jaws are different. Thus we can use the set of architectures and shapes of these components to be the features for the face recognition problem. The common algorithms are active contour model [1] and deformable template model [2].

Second, the sub-space analysis method is often used in face recognition, which contains PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis) are the two common methods. The sub-space method is to find a linear or nonlinear space transformation, which can change the original image data to a subspace, so that it make the distribution of data in the sub-space is tenser.

The most classic method is PCA-based Eigenface which was put forward by Turk [3] in 1991. This method take the face images as random variables, which turns the N× N vector of a face image to a N2 × 1 vector, and after minuses the mean data vector, uses the K-L transformation to get a set of orthogonal basis, then after keeps

ISSN: 2005-4254 IJSIP

part of the principal components, the reduced dimension vector space of face images is got.

However, LDA [4] is aimed at the separability of the samples. It tries to find a projection direction, which can make the distance of within-class, is small and the distance of between-class is large based on the training samples' projection to that direction. Compared to the PCA method, only if the training sample is large, LDA can get a better result.

Summed on these mentioned linear sub-space methods, it is easily to find out that they in fact propose a linearly simplification on the complicated changes of expressions, postures and illumination of the face images. So they cannot get a sufficient representation of face images. The nonlinear sub-space method uses the kernel method to implement the face recognition. Its basic idea [5] is first using a nonlinear transformation when dealing with the dataset like classifying to the linear inseparable samples, which can change the original sample space to a high dimension space (kernel space) to be linearly separable or approximate linearly separable. Thus the data sample can be classified with a linear method to carry out the nonlinear problem. This method only needs a kernel function to compute the inner product of each pair vector in the kernel space without computing the exact nonlinear transformation. The nonlinear sub-space methods have been used, such as kernel principal component analysis [6, 7], kernel Fisher Discriminant Analysis [8] and kernel independent component analysis [9] and so on.

The third method of face recognition is based on the elastic graph matching (EGM).Its basic theory is graph match, which uses a graph to represent the face, the vertices represent the local characteristics of facial points and the topological edges stand for the relationship between face features. Matching measure considers distance between the vertices and edges and it is a local characteristic matching method which can identify the local feature points. Lades [10] proposed dynamic link structure (DLA) for face recognition: the face is composed of a group of link edge nodes which corresponding to the facial specific feature points and is known as the benchmark; the edges are represented by the distance of nodes and the nodes are represented by feature vectors contain local grey distribution information. And the similarity of face images is measured by the similarity of corresponded elastic graphs. Wiscott [11] improved this method that he used a set of image features to represent every node, enhancing the represent ability and adaptability. After these work, others continued this idea in feature analysis, reducing dimension and algorithm.

The face recognition method based on EGM takes into consideration the local details of faces and keeps the spatial information. Moreover, to a certain extent, it can ignore the deformation of changing from 3D to 2D faces. But because it essentially bundles several different frequencies of information into a single vector when extracting the face features, it is hard to extract the significant face features and the calculation costs a lot.

Fourth, researchers use the hidden markov model (HMM) to solve the problem that the different appearance of organs and the connection of each other. Based on this model, the feature observed treated as a sequence of unobserved states. Different people use different HMM parameters, and for the same person, we use the model with same parameters to represent the observed sequence of gestures and facial expressions. Samaria [12] first proposed the face model, who used a rectangular window sampling face images from top to bottom. It arranged pixels in the window into vector and used grey value as the observation vector. Nefian [13] took two-dimensional discrete cosine transform to extract the features as observation vector, reducing the storage of the parameters. And Othman [14] put forward a low complexity two-dimensional HMM, which can better describe the relationship

between every organs and has a higher recognition rate. The face recognition method based on HMM allows the big change of facial expression and head rotation and gets a high recognition rate, but it costs a large computation in feature extracting and model training.

The last usually used method for face recognition is neural network (NN) to use its ability of learning and classifying for extract and recognize face features. Lin, etc. [15] use the positive and negative samples for reinforcing learning to get an ideal probability result. And they increase the learning speed by applying a modular network. This method gets a good application on face detection, face position and face recognition. Meng [16] used PCA to reduce the dimension of face samples before taking the LDA to extract discriminate features, and after these steps they proposed a RBF neural network classification. The result showed that this method had high learning efficiency and recognition performance.

And for our work, since the four stages of face recognition task, face detection, face alignment, face representation and face classification, we use the neural network method for the representation and support vector machine for the classification stage. From the summation mentioned above, we get that the self-learning ability of neural network is so strong that it can get the implicit expression of information after repeating the process of learning. It has a big advantage of extracting face features though it needs a lot of input nodes and parameters, and hard for training. The SVM model has been used for a huge amount of classification problems and it is easy to carry out compared to other methods.

After the section 2 of related work, we describe our system structure and detail parameters setting in section 3. Section 4 is the experiments and our result. We conclude our work in section 5.

2. Related Work

In recent years, since we put all our related information like photos and videos on the Internet, there are a large number of faces, objects and scenes we can crawl from the search engines. And to our face recognition task, we get a harder situation that we need to progress a bigger dataset, which not only increases the calculation but also cannot be implemented with the original classifiers because of the large dimension of the images.

From the introduction section, we can get the basic idea of face recognition is to reduce the dimension of training face data and then use a classifier to get the right class label. Besides the image preprocessing stage, we put more attention on the feature extracting stage.

The common idea of feature extracting is mapping face images to low dimensional target spaces, beginning with the PCA-based Eigenface method [3], which uses a linear projection to train non-discriminatively to maximize the variance. And then comes the LDA-based Fisherface method [17], also linear, but different emphasis. Nonlinear extension has been discussed based on Kernel-PCA and Kernel-LDA[18]. One shortcoming we cannot overlook of those methods is that they are very sensitive to geometric transformations of the training face images, such as shifting, scaling and rotation, and some variabilities, such as the changes in facial expression and glassed. So it is a problem to be solved first, some authors have proposed the similarity metric that is invariant to some known transformations, for instance the Tangent Distance method [19].

Summed to all the methods from the introduction, all those models are hand-designed. Recently, there has been a considerable interest in deep neural network [20, 21].And a deep neural network like CNN (Convolutional Neural Network) can learn the variations from the data without any prior knowledge. The

advantages of using it are: 1) it can be applied to deal with a large amount of training data, 2) from 1) it can learn a wide range of invariances exists in the data and 3) since thousands of CPU cores and GPU's [20] have been used, we can get the result with a less time. Krizhevsky *et al*. [20] gave us an example that large deep CNN [21] trained by standard back-propagation can achieve high recognition accuracy when trained on a large dataset.

Some dimensionality reduction techniques compute a target vector from each input data based on known pair-wise dissimilarities, without constructing a mapping, such as Multi-Dimensional Scaling (MDS) and Local Linear Embedding (LLE). Inside we use CNN one for all. Researchers use a siamese architecture for signature verification. And [22] also uses the network with such siamese architecture, but with different loss function minimized by the training process. We use the loss function just like [22] derived from the discriminative learning framework for energy-based models (EBM).

Edgar Osuna [23] used the SVM method to solve the face detection problem without any preprocessing of the face image. They represent the most important points to finish the detection work. James uses the SVM method to solve a classification and regression task, which shows a perfect result that SVM, can get a good recognition rate. Though the SVM method is not very suitable for the large dataset, [24] presents a novel SVM classification approach for large data sets by considering models of classes distribution (MCD). After getting a sketch of classes distribution, they then obtain the Support Vectors (SVs) between each class and construct a ball using minimum enclosing ball. For our method, after using the CNN method we can get low representation of the large training data, we can use the SVM method to classify based on those face features.

3. System Framework

We separate our system into three main stages, image preprocessing, face representation and face classification. So, we describe the architecture of each stage in the following three parts.

3.1 Image Preprocessing

When solving a face recognition problem or other face problem, we often do the face image preprocessing first. Because the original training data is not suitable for the training method. Usually, we choose to do face detection, face tracking, face cropping and face alignment. So, when comes to the face recognition problem, it is necessary to solve those problems before training.

Face detection is a computer technology that determines the locations and sizes of human faces in digital images. It detects face and ignores anything else, such as buildings, trees and bodies. Face detection can be regarded as a more general case of face localization. Researchers always believe that face detection is the first task to locate the human face among a lot of other objects in a single picture. And when faces could be located exactly in any scene, the recognition step afterwards would not be so complicated. After the face detection stage, it often comes to the face cropping progress to crop faces from the original training face images and then do the face alignment. The objective of face alignment [25] is to localize the feature points on face images such as the contour points of eye, mouth and outline. Face alignment is essential to many face processing applications including face recognition, modeling and synthesis. And after resizing, we get the input data of our model.

We use the LFW (Labeled Faces in the Wild) dataset, which will be described in detail in section 4, and there are some exiting aligned versions of it. Though the

problem of aligning faces is still considered difficult, there have been shown successful result by using sophisticated techniques. These methods are in the following descriptions: (1) employing an analytical 3D model of the face, (2) searching for similar fiducial-points configurations from an external dataset to infer from, and (3) unsupervised methods that find a similarity transformation for the pixels. Based the method on the, we directly download the deep funneling images from the LFW website. And use the detection program to get the 40*48 pixels face data for our training.

3.2 Face Representation

In computer vision literature, these years, many researchers focus on the descriptor engineering. They often use the same operator to all locations in the facial image when comes to face recognition. Recently, since the training dataset becomes larger, the learning methods have started to outperform engineered features due to they can discover and optimize features for the specific task at hand [20]. Here, we use a generic representation of facial images through a large deep convolutional network.

Each side of our convolutional neural network is trained on a muti-class face recognition task, which is to classify the identity of a face image. The net's architecture is shown in Fig. 1. The input of each CNN is a preprocessed funneled gray face image of size 60 by 48 pixels and is given to a convolutional layer (C1) with 32 filters of size 5*5 (we donate this by 32*5*5@60*48). Then we get 32 feature maps and feed them to a max-pooling layer (P2) with the max over 2*2 spatial neighborhoods and the stride is 1, separately for each channel. After these is another convolutional layer (C3) that has 64 filters of size 3*3, follows with a max pooling layer (P4) which takes the max over 2*2 spatial neighborhoods with a stride of 1. The purpose of these four layers is to extract low-level features, like simple edges and texture. Though the pooling layers may cause the network to lose information about the precise position of detailed facial structure and micro-textures, we still use two layers to make the network more robust to small registration errors. We set these layers for a front-end adaptive pre-processing stage. And they are responsible for most of the computation, which expand the input into a set of simple local features.

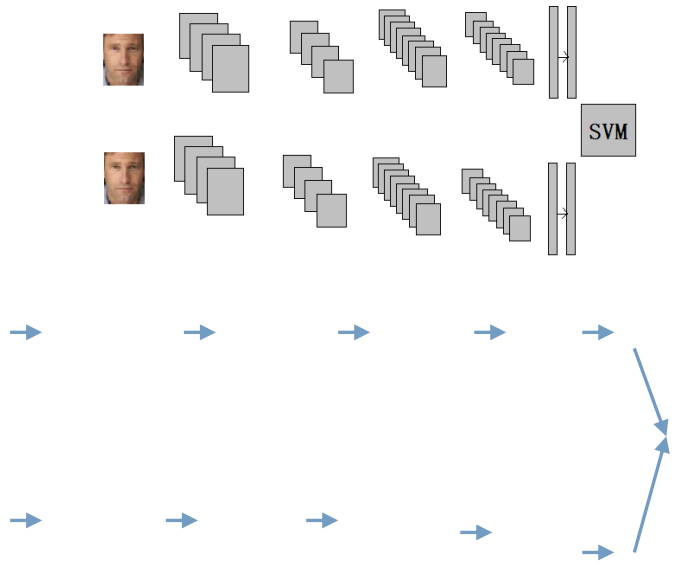

Figure 1. The Overall Architecture of our System

After the preprocessing layers, it comes two full connected layers (F5, F6) with each output unit is connected to all inputs. These layers are able to capture correlations between features captures in distant parts if the face images, such as position and shape of eyes and position and shape of mouth. And the first fully connected layer (F5) in the network will be used as our raw face representation feature vector, which is the input of the SVM classifier. The features this method gets are different with the existing LBP-based representation, which normally pool very local descriptors.

Since we use a siamese convolutional neural network, the two sides of the network are same from the figure shows. And after the last fully connected layer, we compute the distance of the two sides and put the output of it as the input of our final layer – a two label softmax layer, which produces a distribution over the class labels. The softmax function is:

The goal of training is $p_k = \exp(o_k) / \sum_h \exp(o_h)$ to maximize the probability of the correct class (the same people with 1, different with 0). And we use the backward propagation method to train our siamese network to get the low value of our loss function. After we get the well-trained model, we use the output of first fully connected layer for the input of our next stage – face classification.

3.3 Face Classification

SVM performs pattern recognition between two classes by finding a decision surface that has maximum distance to the closet points in the training set which are termed support vectors. Take the Figure 2 a) for an example, there are many possible linear classifiers that can separate the data, while there is only one classifier which maximizes the margin showed in b). This one is called the optimal separating hyperplane (OSH).

Fig. 1. Arbitrary hyperplanes Fig. 2. Optimal hyperplane

Figure 2. a) Arbitrary Hyperplanes b) Optimal Hyperplane

Then we consider our face classification problem to a verification problem with two classes, same or not. A verification algorithm is presented with two face images which may from the same person or not, which is the input training data for our SVM classifier, after the CNN feature extracting.

A SVM algorithm generates a linear decision surface, and the result of the face in image p, the other uses a, is the same if

$$w \cdot p + b \leq 0,$$

Otherwise the face ids are different.

This SVM classifier is designed to minimize the structural risk, which is an overall measure of classifier performance. While, verification performance is usually measured by two statics, the probability of correct verification, P_V, and the probability of false acceptance, P_F. And when the claims are rejected,

$P_V = P_F = 0$; while all claims are accepted, $P_V = P_F = 1$. The operating values for P_V and P_F are dictated by the application.

For SVM, the decision surface produces a single performance point for P_V and P_F. To allow for adjusting P_V and P_F, we parameterize a SVM decision surface by Δ. The parameterized decision surface is

$$w \cdot z + b = \Delta,$$

and the identity of the face image p is the same person if

$$w \cdot p + b \leq \Delta.$$

If $\Delta = -\infty$, then all claims are rejected and $P_V = P_F = 0$; if $\Delta = +\infty$, all claims are accepted and $P_V = P_F = 1$. By varying Δ between negative and positive infinity, all possible combinations of P_V and P_F are found.

The distance of a point x from the hyperplane is,

$$d(w,b;x) = \frac{|w \cdot x + b|}{\|w\|}$$

The margin is according to its definition. Hence the hyperplane optimally separates the data is the one that minimizes

$$\frac{2}{\|w\|} \qquad \phi(w) = \frac{1}{2}\|w\|^2$$

And we use the Lagrange functional to solve the optimization problem above. The solution to the dual problem is given by,

$$\overline{\alpha} = \arg\min_{\alpha} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1|}^{l} \alpha_i \alpha_j y_i y_j x_i \cdot x_j$$

with constraints, $\alpha_i \geq 0, \ i = 1,...l$, $\sum_{i=1}^{l} \alpha_i y_i = 0$

After the training of our SVM, we get the overall architecture of our face recognition system. The next section will introduce the detailed steps and the result of the performance.

4. Experiments and Results

Under the architecture the section 3 gave, we develop our experiment based on the training data from the Internet. First, we do the images preprocessing, based on the LFW dataset. After getting the preprocessed face image vectors, we train our model and extract the face features. And use these reduced dimension features to model our SVM classifier. At last, we value our system, using the testing pairs, and show the ROC result.

4.1 Face Datasets

We extract our face features from a large collection of photos from a well-known labeled dataset network, Labeled Faces in the Wild (LFW). This dataset is designed, by the compute vision lab of University of Massachusetts, Amherst for studying the problem like face recognition, face detection or face alignment. It contains more than 13,000 images of faces (right now is 13233 images) collected from the web, which have been labeled with the name of the person photographed. The whole dataset has about 5749 people in it. But there are not every person has more than one picture. There are only 1680 of them have two or more distinct photos in the dataset. The only constraint on these faces is that they were detected by the Viola-Jones face detector. Figure 3 is the examples of the LFW dataset.

Figure 3. The Examples of LFW Dataset

4.2 Training on the LFW

We randomly select 5500 face ids for being the training set and 247 people for validation every training time. We develop our 1 label image pairs from the same face ids and 0 label image pairs from different ones. The original picture is 150*150 pixels, and we use the deep funneled images with 250*250 pixels. Since the images we download have already aligned, we then cut the images into 60*48 pixels and turn them to gray images. Before the training, we do the histogram equalization for all. However, we also compute the mean data of all the face pictures and let every picture to subtract this vector. After the image preprocessing, we get the training and testing dataset, which can be directly used to our model. We train our Siamese convolutional neural network on the true and false list of 5500 picked face ids. We select all the same pairs under the same id for the true samples and use the two-size number of them for seldom picking the false samples from different ids. This Siamese CNN model is trained on a GPU-based engine, implementing the standard back-propagation on feed-forward nets by stochastic gradient descent (SGD) with momentum (set to 0.9). We choose the batch size as 100 and have set an equal learning rate for all learning layers to 0.01, which was manually decreased, each time by an order of magnitude once the validation error stopped decreasing, to a final rate of 0.0001. The weights are initialized in each layer from a zero-mean Gaussian distribution with $\sigma = 0.01$, and biases are set to 0.5. We train our network for 100 epochs over the whole training data which takes almost 4 days. As section 3.2 described, we use the first fully connected layer's output as our face representation for the next step SVM training. We value our system from the SVM result, which will be showed at next part.

4.2 Results on the LFW

In face recognition task, recently, the computer vision community has made significant progress in unconstrained environments. The mean recognition accuracy on LFW [26] marches steadily towards the human performance of over 97.5% [27]. Given some very hard cases due to large lighting, aging effects and face pose variations in LFW, any improvement over the state-of-art is very remarkable and the system has to be composed by highly optimized modules. For our model, we have achieved the accuracy among 97% and the ROC curve Figure shows the results.

Figure 3 Model Results

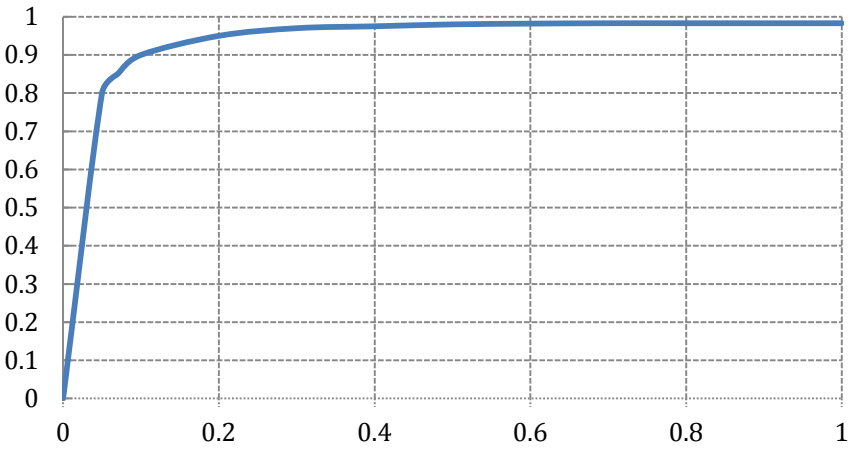

Figure 3. Model Results

5. Conclusion

We build a discriminative model for solving the face recognition task. This model can also solve verification problem. The task always focuses on large number of classes, or all the classes are not available when training. This method we propose is a siamese convolutional neural network to extract the face features and use them as input to the support vector machine classifier. This work takes shorter time and almost same accuracy compared with other methods. Our future work aims at enriching our training image dataset and tries to use other loss function and find better network architectures.

References

[1] B. Olstad and A H. Torp, "Encoding of a priori information in active contour models [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 18, no. 9, (**1996**), pp. 863-872.
[2] A K. Jain and Z. Yu, "Deformable template models: A review [J]", Signal Processing, vol. 71, no. 2, (**1998**), pp. 109-129.
[3] M A. Turk and A P. Pentland, "Eigenfaces for recognition [J]", Journal of Cognitive Neuroscience, vol. 3, no. 1, (**1991**), pp. 71-86.
[4] J. Lu, K N. Plataniotis and A N. Venetsanopoulos, "Face recognition using LDA-based algorithms [J]", IEEE Trans. on Neural Networks, vol. 14, no. 1, (**2003**), pp. 195-200.
[5] V N. Vapnik, "The nature of statistical learning theory", **(2004).**
[6] C. Liu, "Gabor-based kernel PCA with fractional power polynomial models for face recognition [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 26, no. 5, (**2004**), pp. 572-581.
[7] J. Yang, A F. Frangi and J Y Yang, "KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 27, no. 2, (**2005**), pp. 230-244.
[8] C. Liu, "Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 28, no. 5, (**2006**), pp. 725-737.
[9] J. Yang, X. Gao and D. Zhang, "Kernel ICA: An alternative formulation and its application to face recognition [J]", Pattern Recognition, vol. 38, (**2005**), pp. 1784-1787
[10] M. Lades, J C. Vorbruggen and J. Buhmann, "Distortion invariant object recognition in the dynamic link architecture [J]", IEEE Trans. on Computer, vol. 42, no. 3, (**1993**), pp. 300-311.
[11] L. Wiskott, J M. Fellous and N. Kruger, "Face recognition by elastic bunch graph matching [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, (**1997**), pp. 775-779.
[12] F. Samaria, "Face recognition using hidden Markov model [D]", Cambridge, University of Cambridge, (**1994**).
[13] A. Nefian, "A hidden Markov model-based approach for face detection and recognition [D]", Georgia, Georgia Institute of Technology, (**1999**).

[14] H. Othman and T. Aboulnasr, "A separable low complexity 2D HMM with application to face recognition [J]", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, no. 10, (**2003**), pp.1229-1238.
[15] S H. Lin, S Y. Kung and L J. Lin, "Face recognition/detection by probabilistic decision based neural network [J", IEEE Trans. on Neural Networks, vol. 8 , no. 1, (**1997**), pp. 114-132.
[16] M J. Er, S. Wu and J. Lu, "Face recognition with radial basis function (RBF) neural networks [J]", IEEE Trans. on Neural Networks, vol. 13, no. 3, (**2002**), pp. 697-710.
[17] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specifi c linear projection", IEEE Trans. PAMI, Special Issue on Face Recognition, vol. 19, no. 7, (**1997**) July.
[18] M. Hsuan Yang, N. Ahuja, and D. Kriegman, "Face recognition using kernel eigenfaces", In Proc. of the 2000 IEEE International Conference on Image Processing (ICIP), vol. 1, (**2000**), September, pp. 37–40.
[19] P. Y. Simard, Y. LeCun, J. S. Denker, and B. Victorri, "Transformation invariance in pattern recognition – tangent distance and tangent propagation", International Journal of Imaging Systems and Technology, vol. 11, no. 3, (**2000**).
[20] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments", In ECCV Workshop on Faces in Real-life Images, vol. 1, (**2008**), p. 6.
[21] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification. In ICCV, (**2009**), p. 6.
[22] S. Chopra, R. Hadsell and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification".
[23] E. Osuna, R. Freund and F. Girosi, " Training Support Vector Machines: an Application to Face Detection", (**1997).**
[24] J. Cervantes, X. Li and W. Yu, "SVM classification for large data Sets by considering models of classes distribution", (**2008).**
[25] L. Zhang, H. Ai, S. Xin, C. Huang, S. Tsukiji and S. Lao, "Robust face alignment based on local texture classifiers".
[26] G. B. Huang, M. A. Mattar, H. Lee, and E. G. Learned-Miller, "Learning to align from scratch", In NIPS, vol. 2, (**2012**), pp. 773–781.
[27] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks", In ANIPS, (**2012**), p. 1.

Author

Weiwei Wu, she received her Master Degree in Software Engineering from Electronic Science and Technology University, China in 2008. She is currently a lecture in Animation and Art Department, Zibo Vocational Institute in Shandong, China. Her research interests on art of animation and digital media.

