Feature extraction and classification using deep convolutional neural networks, PCA and SVC for face recognition

Mohammed Kamel Benkaddour[1,2,*], Abdennacer Bounoua[2]

1. Laboratory Communication Network & Architecture Multimedia RCAM
   DJILLALILIABBES University Sidi Bel Abbes 22000, Algeria

2. University Kasdi Marbah, FNTIC Faculty, Ouargla 30000, Algeria

   benkaddour.kamel@univ-ouargla.dz

ABSTRACT. Recently, the facial recognition has aroused the interest of the scientific community, this technique of biometric that is effective, non-intrusive and contactless has taken an increasingly important part in the field of research. This paper proposes a face recognition and classification method based on deep learning, in particular Convolutional Neural Network (CNN), which are incredibly a powerful tools that have found great success in image classification and pattern recognition. In this work, the approach to this task is based on the Convolutional Neural Network (CNN) as a powerful feature extraction followed by Support Vector Machines (SVM) as a high classifier. To reduce the dimension of these features, a principal component analysis (PCA) technique is employed. We conduct an extensive evaluation of our methods on the FERET dataset. The results obtained showed that the proposed method CNN combine with PCA and Svc solution provide a significant improvement in performance and enhance the recognition accuracy.

RÉSUMÉ. Récemment, la reconnaissance faciale a suscité l'intérêt de la communauté scientifique. Cette technique de biométrie efficace, non intrusive et sans contact occupe une place de plus en plus importante dans le domaine de la recherche. Cet article propose une méthode de reconnaissance faciale et de classification basée sur un apprentissage en profondeur, en particulier le réseau neuronal convolutif (CNN en anglais), qui s'avère être un outil puissant et qui a connu un grand succès dans la classification des images et la reconnaissance des formes. Dans cet article, l'approche de cette tâche est basée sur le réseau neuronal convolutif en tant qu'extraction de caractéristique puissante, suivi par la machine à vecteurs de support (SVM en anglais) qui se sert à un classificateur élevé. Pour réduire la dimension de ces caractéristiques, une technique d'analyse en composantes principales est utilisée. Nous effectuons une évaluation approfondie de nos méthodes sur l'ensemble de données de la technologie de reconnaissance ficiale (FERET en anglais). Les résultats obtenus ont montré que la méthode du réseau neuronal convolutif associée aux solutions d'analyse en composantes principales et de SVC améliore considérablement les performances ainsi que la précision de la reconnaissance.

1. Introduction

Recently, face recognition has become one of the most interesting and important tasks in the field of computer vision and pattern recognition. The facial recognition as a basic biometric technology has taken attracted considerable interest due to their various applications (video surveillance, access control to sensitive sites, remote monitoring ...).

Face recognition systems are dependent on a robust feature extraction. That's why many approaches have been proposed to solve the problem of human face recognition, but despite the many methods it remains an extremely difficult problem, this is due to the fact that different people face generally have the same shape and vary due to the lighting conditions, the change of pose, facial expressions.

Before detailing the techniques used in this paper we will present first an overview of studies done by researchers in facial recognition task.

Various approaches have been proposed in the literature (Petrovska-Delacrétaz *et al.*, 2009), they can be categorized into tree main groups:

- Local approaches: they extract facial features by focusing on the critical points of the face such as the nose, mouth, eyes; which will generate more details.

- Global approaches: their principle is to use the entire surface of the face as a source of information regardless of local features such as eyes, mouth ... etc.

- Hybrid approaches: they combine both types of methods, potentially offering the best of both.

The first theoretical studies in automatic face recognition began in the early 1960s (Bledsoe, 1968), it's just the last thirty years that research on face recognition has grown extensively, when Kirby and Sirovich (Kirby & Sirovich, 1990) applied a linear algebra technique called principal component analysis (PCA) to model a human face. This technique popularized then under the name "Eigenfaces" after the improvement brought by Turk and Pentland (Turk & Pentland, 1991), it is one of the most successful methods widely used in face recognition.

Another popular technique is called Fisherfaces based on the Linear Discriminant Analysis (LDA), which subdivides the faces into classes while maximizing the inter-class variance (between two different individuals) according to the Fisher criterion (Kriegman *et al.*, 1996).

Among the other global methods, we find Bayesian approaches, and the SVM method (Guo *et al.*, 2000), which uses Support Vector Machine (SVM), Active

Appearance Models (AAM) (Cootes *et al.*, 2001) or the "Local Binary Pattern "(LBP) method (Ahonen *et al.*, 2004) were used in order to easily model pose variations, lighting and expression in comparison to local methods.

These last decades have seen the introduction of artificial neural networks (ANNs) (Andrews *et al.*, 1995) which is a very powerful and robust classification technique used in various fields including pattern recognition.

The study of neural networks has gained as research interests and different ANNs structures has been proposed by many researchers for face recognition such as, Multilayer Perceptron (MLP) (Samal & Iyengar, 1992), Fast Neural Network (Moody & Darken, 1989), Back Propagation Network (BPN) and Radial Basis Function Network (RBF) (Lecun *et al.*, 1989). The challenge is to identify the most appropriate neural network model which can work reliably for solving our problem.

Over the last few years, a new model of neural network has been proposed which is based on Deep learning (Schmidhuber, 2015). This last have achieved state-of-the-art performances in wide areas according to the powerful ability for approximation and feature extraction to improve the accuracy of image classification.

When referring to the face recognition based on neural network, we may commonly think about the methods such as Convolutional Neural Network (CNN) (Lawrence *et al.*, 1997), Deep Belief Network (DBN) (Hinton *et al.*, 2006), and Stacked Denoising Autoencoder (SDAE) (Vincent *et al.*, 2010).

Recently, convolutional neural networks (CNNs) have been used as a powerful tool to solve different problems of machine learning and biometrics. This model of deep neural network inspired by the biology of human vision (Lecun *et al.*, 2015) consists of alternating convolutional and pooling layers, with fully-connected layers on top. Generally speaking, CNN is a good choice for face recognition. In 1998, Lecun *et al.*, (1998) have successfully processed the 2-D images with multi-layer CNN. With the development of computer hardware, in 2012, Hinton and Krizhevsky (2012) applied the deep CNN to process ImageNet database, and achieved a better result than ever. Several CNN architectures have been proposed in the literature (Khiyari & Wechsler, 2016) and some have been shown to produce better results than the most of deep neural network model.

To overcome the limitations of the techniques described above, we propose a new method based on a convolutional neural network (CNN) for feature extraction combined with Support Vector machine and PCA algorithm.

In this paper, first the CNN models are designed of how to extract the discriminate feature vectors is illustrated, then the practical adjustments necessary to combine the extracted image features (by CNN) from face database with feature dimension reduction by introducing principal component analysis and SVM is used for classification to enhance the recognition accuracy. After that, the experimental results obtained are analyzed, followed by a discussion with results interpretation are presented.

2. Proposed method

In our proposed method, first a convolutional neural network is adopted to learn and extract feature of input images. To obtain more compact feature, PCA is used to reduce the number of components while keeping the information characterizing the object to be analyzed. Finally, we conduct binary classification by SVM to realize face classification on images. The overall architecture of the recognition system being used is shown in Figure1.
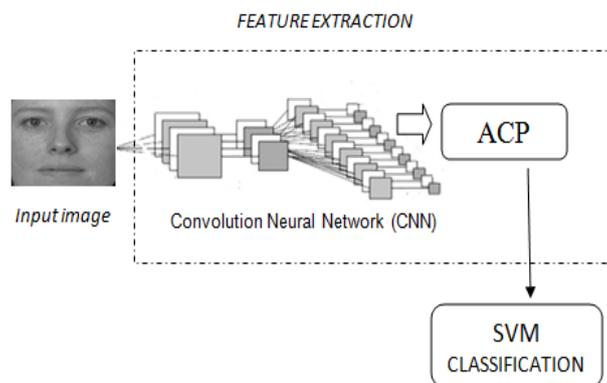

Figure 1. The proposed method

2.1. Convolutional neural networks

The idea of Convolutional Neural Networks was inspired by the structure of the visual system (Bengio, 2009), The first computational models have been introduced by Fukushima (1980), Later, Lecun *et al.*, (1999) generalized them to classify the image successfully and for recognizing handwritten control numbers by Lenet-5, recently Ciresan *et al.*, (2013) used convolutional networks and realized some performances in the literature of object recognition for different image databases.

LeCun's convolutional neural networks are organized in a sequence of alternating two types of layers S-layers and C-layers, called convolution and sub-sampling with one or more fully connected layers in the ends. Typical convolutional network architecture is shown in Figure 2.
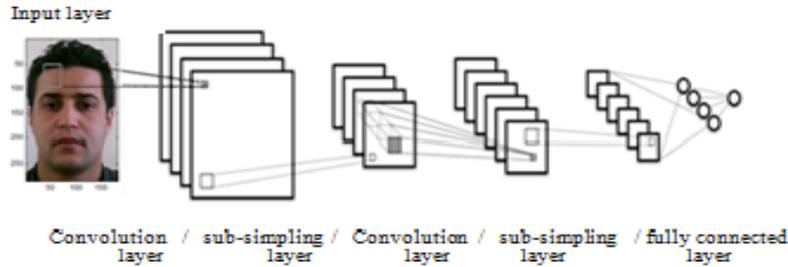
Figure 2. Typical convolutional network architecture

CNN is a multi-layered neural network. Figure 2 shows the general structure of a Convolutional neural network. The CNN has the ability to perform both feature extraction and classification. The input layer receives normalized images with the same sizes.

Then the image is convolved with multiple learned kernels using shared weights. CNN is composed of a stack of convolutional layers, a convolutional layer is parameterized by the number of maps. CNNs apply a number of filters to the raw pixel data of an image to extract and learn features, which the model can then be used for classification.

The feature map of each pixel is calculated as follows:

$$C_n = f(x * W + b) \tag{1}$$

where "$*$" indicates the convolution computation, n is the pixel in the feature map, x is the pixel-value, W and b are the convolution kernel and bias respectively, and f is the convolution kernel.

After this convolution layer, we apply a nonlinear layer (activation layer) immediately like tanh and sigmoid, but researchers found out that the application the Rectified Linear Unit (ReLU) ReLU layers (Vinod & Hinton, 1999) work far better because the network is able to train faster, The ReLU layer applies the function as follows:

$$f(x) = \max(0, x) \tag{2}$$

This function is applied to all of the values in the input volume to introduce nonlinearities into the model, the ReLU function has been employed in our work.
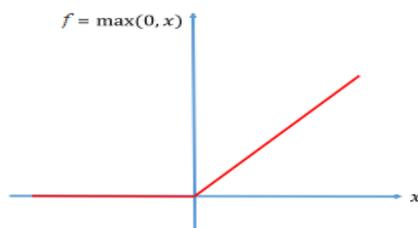
$$f = \max(0, x)$$

Figure 3. The ReLU function

After ReLU function, we apply a pooling layer to reduce the size or downsample the image data extracted by the convolutional layers in order to decrease processing time. There are also several options in this pooling layer corresponding to max-pooling, mean-pooling, or stochastic-pooling, over non-overlapping, but max-pooling being the most popular, which extracts sub-regions of the feature map to its maximum value.

Convolutional layer and pooling layer compose the feature extraction part. Afterwards, it comes the fully connected layers which perform classification on the extracted features by the convolutional layers and the pooling layers. These layers are similar to the layers in Multilayer Perceptron (MLP).

Finally, CNN contains one output neuron for each target class in the model, a softmax activation function was used to generate a value between 0–1 for each node, there exists one output neuron for each object category in the output layer. The output layer has one neuron per class in the classification task. A softmax activation function is used for each neuron's output to predict the classification result into each target class.

2.2. The network structure

Many variants of the CNN architectures, including number of layers, feature maps and the connection scheme can be adapted to the given classification task. In our convolutional neural network all samples are scaled to 32×32 pixels, then we use a Pre-trained convolutional network model for face recognition which is trained on a database for face recognition other than the one used for face recognition task. In this work, we use part of the CASIA Webface dataset (Dong *et al.*, 2014) to train our CNN, and get weights that represent facial features. Recent studies have shown (Chowdhury *et al.*, 2016; Yosinski *et al.*, 2014) that trained model can be applied to similar classification problems and will perform well.

Then we use these weights to initialize the layers of CNN except the last fully connected layer of the network. This approach is well suited for face Classification problems, where instead of training a CNN from scratch. Once we have established

the weights of all the layers, we construct our CNN with a set of target training data to extract the facial features and use them to train SVM. For test dataset, we use the trained CNN to extract functionality and use these features to recognize all images by the SVM classifier formed.

The structure of our CNN as table1 shows is trained on a database to face recognition task, which is used to classify the face image. The network architecture includes number of layers, The input of CNN is gray face image of size 32 by 32 pixels and is given to three convolution layers that compose the network with Conv 1 consist of 30 feature map, C2 consist of 60 feature map and C3 consist of 80 feature map , then three pooling layers are used , Layer S1, S2 and S3 are subsampling layers, whose number of feature maps is equal to the maps number of their previous convolution , After the preprocessing layers, comes to the full connected layers Fc with 512 neurons, each one connects to all the maps of last pooling layer followed by the output layer is also a full connected layer.

Table 1. Detailed architecture of our CNN used for features extraction

name	type	Description of output size
input layer	Input data	$1\times32\times32$
Conv1	Convolution + ReLU	30 5×5 filters
S1	Max Pooling	2×2, stride 2
Conv2	Convolution + ReLU	60 5×5 filters
S2	Max Pooling	2×2, stride 2
Conv3	Convolution + ReLU	80 5×5 filters
S3	Max Pooling	2×2, stride 2
Fc	Fully Connected + dropout	512 units1×1
	Fully Connected	
output layer		

Some adjustments are used to optimize our CNN, each convolutional layer is followed by a Rectified Linear Unit's function (ReLU), since ReLU can speed up learning in deep neural networks and lead to higher performance. We apply also weight penalty based on L2 regularization in our model because the smaller network weights could effectively prevent the over-fitting. This is done by removing the top output layer and using the activations from the last fully connected layer as features. We use dropout in order to combat the problem of overfitting to the training data which is a regularization technique and it work well with fully-connected layers. Dropout improves performance of the smaller architecture and the fully-connected layers have many units and need regularization.

2.3. The pre-training of CNN

In this work, we have selected deep neural network architecture for face recognition as a pre-training task and examine the network's ability to perform feature extraction and classification while the databases have limited training samples.

Since big data are important for learning a convolutional neural network and It's often trained on large amounts of data to be able to extract features generalizing well, On the other hand several studies have investigated training CNNs using small data (Schroff *et al.*, 2015; Sun *et al.*, 2014) We explore this idea by introducing the CASIA-Webface dataset (Zhao *et al.*, 2003) for pre-training our CNN, which is the largest public dataset for the face recognition. The dataset contains 494,414 images from 10,575 identities.

The images display a wide range of variability in pose, expression, and illumination. We use scaled 32×32 input images to train a CNN with architecture detailed in table 1. We select part of the samples as our pre-training dataset about 120,000 face images of Casia-Webfaces. The pre-training dataset is inputted to the convolution neural network that we describe in the subsection 2. 2), and network weights will be updated after a certain number of iterations. The idea of Transfer Learning is that the knowledge learned from one problem can be applied to similar classification problems.

Then we use our target data FERET dataset to train CNN, we may be initialing the weights of CNN except its last layer with the weights obtained by Casia-Webfaces database. The final convolution neural network would have the stronger generalization ability and faster convergence rate.

2.4. Feature and dimensionality reduction by PCA

After feature extraction by CNN-based method, the features can contain higher dimensional that lead to higher computation and information redundancy, Therefore, we adopt a principal component analysis (PCA) (Moon & Phillips, 2001) to reduce the noise and the feature dimension before recognizing faces. PCA (Yambor *et al.*, 2002) is a statistical method for finding correlations between features and reducing the dimensions of data. When used on images of faces the resulting images are referred to as Eigenfaces.

The idea behind PCA is to project the CNN features to a lower dimensional subspace in order to improve the performance SVM classifiers, by mapping the data points to a lower dimensional space we can categorize data into a set number of classes.

Experiments have shown that in selecting only a few dimensions of the PCA feature space produces comparable face recognition rates to those of the original space. It is a very interesting property of CNN-learned features, because low dimensionality can significantly reduce memory and computation.

2.5. Classification using SVM

In this paper, we have developed face recognition based on CNN and SVM. After extract the image features, the input image is first transmitted through the CNN network, and the extracted features are obtained from the fully connected layers, whose dimension is reduced by PCA, then we use SVM as our final classifier to recognize faces because of its classification effect on nonlinear data. Support Vector Machine was proposed firstly by Vapnik and Cortes (1997), SVM shows many advantages in solving nonlinear and high dimensional pattern recognition, is a powerful discriminative classifier. It has been widely exploited with good performance for many pattern classification and face recognition (Wright *et al.*, 2009).

The SVM was trained on the deep features extracted from our CNN. In this paper, we adopt the Support Vector Clustering (SVC) (Osuna *et al.*, 1997) function to find the support vector, the objective of clustering is to partition a data set into groups according to some criterion in an attempt to organize data into a more meaningful form. SVC (Ben-hur *et al.*, 2001) may map the data points to a high dimensional feature space by using the Gauss kernel, which could find a smallest sphere that can surround all the data points. The sphere is mapped back to the data space, which makes a set of contour lines of closed data points. These data points that are closed by the contour line belong to the same cluster.

In our system, the input of SVM is the facial features of the output layer in CNN after reduction by PCA. The input training dataset and the testing dataset of SVM are the output features of the training dataset and the testing dataset of CNN, respectively. The training label and the testing label of SVM are respectively same to the training label and the testing label of CNN. In SVC function, we adopt Radial Basis Function (RBF) as our kernel function.

For the final recognition, we use the one-versus-one method in SVM. The one-versus-one is to design an SVM between any two types of samples in training dataset and determine which class the sample is more likely belongs to in each SVM. SVC it works best for low-dimensional data, so a preprocessing step by using principal component analysis produces data that clustered well.

3. Datasets

The Face Recognition Technology (FERET) is used to evaluate our method which is a large database for facial recognition system evaluation (Phillips *et al.*, 2000). The Face Recognition Technology program is managed by the Defense Advanced Research Projects Agency (DARPA) and the National Institute of Standards and Technology (NIST). FERET was collected in 15 sessions between August 1993 and July 1996. The database contains five datasets: Fa (1196 images), Fb (1195 images), Fc (194 images), Dup1 (722 images), and Dup2 (234 images) each of which has 7 pictures under the different expression, light, and posture conditions. The gallery set is Fa, and the other datasets are used for testing.
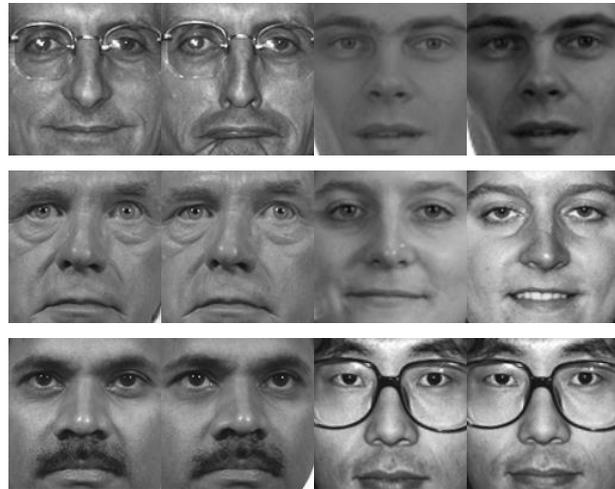
Figure 4. Some face images from FERET database

In order to extract facial features, the gallery Fa is used as training dataset, while testing dataset we use part of Fb gallery which contains pictures taken on the same day as the Fa images, using the same camera and under the same lighting conditions. In our experiments, FERET requires pre-processing step by region-of-interest (ROI) extraction, and image resizing (cropped) to 128 ×128 pixels.
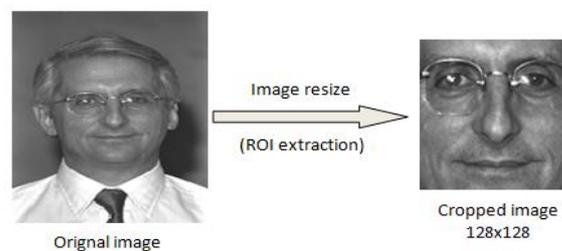

Figure 5. Pre-processing FERET database

4. Experimental work and results

After we have built up our pre-trained CNN, we compare the result of the FERET dataset before and after the use of pre-training in order to evaluate the neural network. The result of using pre-training has demonstrated promising performance and is better than not using pre-training in recognition rate and convergence speed.

The pre-trained CNN is a model created to solve a similar problem, it has demonstrated a strong ability to generalize to faces images. However, we should be very careful while choosing the CNN model used in your case.

In this case, since the pre-formed network was formed fairly well, then we do not modify the weights, then we use the pre-trained model as extractor features.

To validate the proposed algorithm, for each person, around half of the image FERET data set is used for training and the rest for testing. After feature extraction, First PCA applies to all features, and then a classifier is needed to find the corresponding label for each test image. Different classifiers can be used for this task, in this work, SVM has been used which is one of the most powerful technique for image classification. In this section, we investigate the performance of the CNN based-SVM, kind of SVM approach is evaluated and the results are summarized as shown in the Table 2 below.

Table 2. Comparison of results on FERET dataset for different algorithm. Mean accuracy

Model	Accuracy
CNN	83,2
CNN-Dropout	85.3
CNN- linear SVM L1 regularization	87.1
CNN- linear SVM L2 regularization	86.4
CNN-SVC	89.6
CNN-SVC - With Dropout	93.4
CNN-PCA-SVC	95.2

The best performing convolutional neural network, introduced in this work, achieve a recognition classification rate of 83.2%. In order to improve this architecture, we added a dropout technique to outperform the basic CNN classifier and recognition rate was ameliorated.

Then we have used different methods based on SVM to replace the last layer of our CNN. Base on Table 2, it is shown that CNN based-SVM model with Non-parametric SVM-based clustering methods may allow for much improved performance compared to SVM with L1 regularization orL1 regularization.

Our proposed system was compared to the other methods being recently discussed, Once the features was extracted by using CNN with dropout technique, this last are transformed to a better representation by introducing PCA and the most important components are then used as new features, describing data vectors better than original. After the resulting vectors are finally passed to SVM classifier (such as SVC). The results show that the recognition rate of CNN+PCA+SVC can reach 95.2% and performs higher than the other methods.

Table 3. Comparison accuracy rate performance with state-of-the-art methods on FERET database

Authors Reference/Year	Methods	Accuracy (%)
Syafeeza (2014)	CNN	81.25 %
Guo *et al.* (2016)	CNN +SVM	87.29%
Zhang *et al.* (2016)	ACNN	91.67%
Proposed approach	CNN +PCA+SVC	95.2%

A comparative study of the performance of our proposed approach to other previous face recognition work on FERET database was shown in (TABLE X), in the light of the above experimental results, our proposed approach achieve the highest accuracy recognition rates, which demonstrates the aptitude of the proposed architecture of CNN based PCA-SVC to outperforms the other current methods.

5. Conclusion

In this paper, we evaluated the application of deep neural network for face recognition. In our system, we treat the pre-trained CNN model as a feature extraction, then PCA is applied to reduce feature dimensionality and SVM is used as a classifier to perform the recognition. In order to better evaluate the CNN after the pre-training, we compare the result of the FERET dataset before and after the use of pre-training. The result of using pre-training is better than not using pre-training in recognition rate and convergence speed.

Based on the experiment results using data from FERET dataset, the proposed method achieves an accuracy rate better than only CNN method. The proposed method was also validated PCA and support vector clustering for classification, and it shows that the recognition rate for this proposed method provide higher face recognition rates than the methods discussed in the comparative analysis. In the future, we will try to explore different pre-trained CNNs with more optimization techniques for a larger dataset.

References

Ahonen T., Hadid A., Pietikainen M. (2004). Face recognition with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 28, No. 12, pp. 2037-2041. https://doi.org/10.1007/978-3-540-24670-1_36

Andrews R., Diederich J., Tickle A. B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based Systems*, pp. 373-389. https://doi.org/10.1016/0950-7051(96)81920-4

Bengio Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, No. 1, pp. 43–44. https://doi.org/10.1561/2200000006

Ben-hur A., Horn D., Siegelmann H. T., Vapnik V. (2001). Support vector clustering. *Journal of Machine Learning Research*, Vol. 2, pp. 125-137. https://doi.org/10.1162/15324430260185565

Bledsoe W. W. (1968). Semiautomatic facial recognition. Technical Report Project 6693, Stanford Research Institute, Menlo Park, California.

Chowdhury A. R., Lin T. Y., Maji S., Learned-Miller E. (2016). One-to-many face recognition with bilinear CNNs. *Applications of Computer Vision (WACV),* pp. 1-9. https://doi.org/10.1109/WACV.2016.7477593

Ciresan D., Meier U., Masci J., Gambardella L. M., Schmidhuber J. (2013). High performance convolutional neural networks for image classification. *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 1237–1242. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210

Cootes T., Edwards G., Taylor C. (2001). Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/34.927467

Cortes C., Vapnik V. (1997). *Soft Margin Classifier*. U.S. Patent No. 5, 640,492.

Dong Y., Zhen L., Shengcai L., Stan Z. L. (2014). Learning face representation from scratch. *arXiv preprint arXiv*: 1411.7923.

Fukushima K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, Vol. 36, pp. 193–202. https://doi.org/10.1007/BF00344251

Guo G., Li S., Chan K. (2000). Face recognition by support vector machines. *The IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 196-201. https://doi.org/10.1109/AFGR.2000.840634

Guo S., Chen S., Li Y. (2016). Face recognition based on convolutional neural network and support vector machine. *Proceedings of the IEEE International Conference on Information and Automation Ningbo, China*. https://doi.org/ 10.1109/ICInfA.2016.7832107

Hinton G. E., Osindero S., Teh Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, Vol. 8, No. 7, pp. 1527-1554. https://doi.org/10.1162/neco.2006.18.7.1527

Khiyari H. E., Wechsler H. (2016). Face recognition across time lapse using convolutional neural networks. *Journal of Information Security*, No. 7, pp. 141-151.

Kirby M., Sirovich L. (1990). Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 1, pp. 103-108. https://doi.org/10.1109/34.41390

Kriegman D. J., Hespanha J. P., Belhumeur P. N. (1996). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *ECCV*. https://doi.org/10.1109/34.598228

Krizhevsky A., Sutskever I., Hinton G. E. (2012). ImageNet classification with deep convolutional neural networks. *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12); Lake Tahoe, Nevada*, USA. pp. 1097–1105. https://doi.org/10.1145/3065386

Lawrence S., Giles C. L., Chung T. A., Back A. D. (1997). Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks*, Vol. 8, No. 1, pp. 98-113. https://doi.org/10.1109/72.554195

Lecun Y., Bengio Y., Hinton G. E. (2015). Deep learning. *Nature*, Vol. 521, No. 7553, pp. 436-444. https://doi.org/10.1038/nature14539

Lecun Y., Boser B., Denker J. S., Henderson D., Howard R., Hubbard W., Jackel L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, Vol. 1, No. 4, pp. 541-551. https://doi.org/10.1162/neco.1989.1.4.541

Lecun Y., Bottou L., Bengio Y., Haffner P. (1998). Gradient-based learning applied to document recognition. *In Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278-2324. https://doi.org/10.1109/5.726791

Lecun Y., Bottou L., Bengio Y., Haffner P. (1999). Object recognition with gradient-based learning. *Shape, Contour and Grouping in Computer Vision*, pp. 823-823. https://doi.org/10.1007/3-540-46805-6_19

Moody J., Darken C. J. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, Vol. 1, No. 2, pp. 281-294. https://doi.org/10.1162/neco.1989.1.2.281

Moon H., Phillips P. J. (2001). Computational and performance aspects of PCA-based face-recognition algorithms. *Perception*, Vol. 30, No. 3, pp. 303-321. https://doi.org/10.1068/p2896

Osuna E., Freund R., Girosit F. (1997). Training support vector machines: An application to face detection. *In: Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.1997.609310

Petrovska-Delacrétaz D., Chollet G., Dorizzi B. (2009). Guide to biometric reference systems and performance evaluation. *Springer*, pp. 111. https://doi.org/10.1007/978-1-84800-292-0

Phillips J., Moon H., Rizvi S., Rauss P. (2000). The FERET evaluation methodology for face-recognition algorithms IEEE Trans. *Pattern Anal. Mach. Intell.*, Vol. 22, pp. 1090-1104. https://doi.org/10.1109/34.879790

Samal A., Iyengar P. A. (1992). Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognition*, Vol. 25, No. 1, pp. 65-77. https://doi.org/10.1016/0031-3203(92)90007-6

Schmidhuber J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, Vol. 61, pp. 85-117. https://doi.org/10.1016/j.neunet.2014.09.003

Schroff F., Kalenichenko D., Philbin J. (2015). Facenet: A unified embedding for face recognition and clustering. *In CVPR*. https://doi.org/10.1109/CVPR.2015.7298682

Sun Y., Wang X., Tang X. (2014). Deeply learned face representations are sparse, selective, and robust. *arXiv preprint arXiv*: 1412.1265. https://doi.org/10.1109/CVPR.2015.7298907

Syafeeza A. R. (2014) Convolutional neural network for face recognition with pose and illumination variation. *International Journal of Engineering and Technology (IJET)*. Vol. 6, No. 1.

Turk M., Pentland A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neurosicence*, Vol. 3, No. 1, pp. 71-86. https://doi.org/10.1162/jocn.1991.3.1.71

Vincent P., Larochelle H., Lajoie I. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, Vol. 11, No. 12, pp. 3371-3408. https://doi.org/10.1016/j.mechatronics.2010.09.004

Vinod N., Hinton G. E. (2010). Rectified linear units improve restricted boltzmann machines. *Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel*. https://doi.org/10.1.1.165.6419

Wright J., Yang A., Allen Y., Ganesh A., *et al.* (2009). Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No 2, pp. 210-227. https://doi.org/US5640492 A

Yambor W. S., Draper B. A., Beveridge J. R. (2002). Analyzing PCA-based face recognition algorithms: Eigenvector selection and distance measures. *In Empirical Evaluation Methods in Computer Vision*, pp. 39-60. https://doi.org/10.1142/9789812777423_0003

Yosinski J., Clune J., Bengio Y., Lipson H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, pp. 3320-3328.

Zhang Y., Zhao D., Sun J. (2016). Adaptive convolutional neural network and it's application in face recognition. *Neural Processing Letters*, Vol. 43, pp. 389-399. https://doi.org/10.1007/s11063-015-9420-y

Zhao W., Chellappa R., Phillips P. J., Rosenfeld A. (2003). Face recognition: A literature survey. *ACM Computing Surveys (CSUR)*, Vol. 35, No. 4, pp. 399-458. https://doi.org/10.1145/954339.954342