

Face Recognition across Age Using Auto Encoder Neural Network

Neha Rahman¹, Nitin Naiyar²

¹M Tech. Scholar, Department of Electronics and Telecommunication, Rungta college of Engineering and Technology Bhilai (C.G.)

²Reader, Dept.ofElectronics and Telecommunication, Rungta college of Engineering and Technology Bhilai (C.G.)

Abstract: *In this modern era of digitization and advanced technology, human face has become a demanding icon to authenticate ones identity. One peculiar feature which distinguishes it from other biometric techniques is that it does not need the test subject to perform its working. Other conditions where face recognition does not work well include poor lighting, sunglasses, hats, scarves, beards, long hair, makeup or other objects partially covering the subject's face, and low resolution images. Facial Aging is such a process that affects both the shape as well as wrinkles on the face. These shapes and wrinkles changes degrade the performance of the automatic face recognition. In this paper, neural network is used for the performance evaluation of research work. Stack Auto-encoderis used for training purpose and serves as one of the input of deep network and confusion matrix is used for calculating the accuracy of the model. The evaluation is performed in the MATLAB environment.*

Keywords: Face aging, Stacked Auto-encoder, Deep Networks, Age Invariants, softmax layer, confusion matrix

1. Introduction

Human face plays indispensable role in social interaction. Whenever human eyes combined with brain processing for face recognition, it follows complex process with higher accuracy. There are various standard algorithms available which offer large calculation during checking of faces with the help of machine such as pattern recognition, PC vision and Neural networks. However humans have used various characteristics of body like face, eyes, voice to recognize each other's from thousands of years [1]. But problem arise in that case where we are unable to identify each other due to various changes such that facial expressions, age factors etc. Thus to overcome this problem numerous technologies came into existence. These technologies are Biometrics like fingerprints, face, signature, Iris, Palm printetc. Basically all these technologies are used for Image retrieval, Passport Photo Verification and Surveillance [2].

Accuracy of face recognition is limited by number of factors expression, pose, lighting and age. All these factors are responsible to degrade performance of the face recognitionsystem. Facial aging is not getting satisfactory attention ascompared to other factors like pose, expression and lighting. Facial aging is a complex scheme that affects shape and texture like skin tone, wrinkles of a face [3]. This aging process varies person to person according to their age group. Although facial aging is generally represented by facial growth of younger age groups, it is denoted by large texture changes and minor shape changes therefore, an age correction scheme needs to be able to compensate for both types of aging processes. There are few applications of face recognition where aging factor is widely used are: (1) to identify missing children (2) enrollment detection problems (3) passport verification etc. Common characteristics of above mentioned cases are:

- Age difference between probe and gallery images.
- Inability to get user's image to update the gallery.

Due to aging, faces undergo gradual variations periodically. Therefore there is a need of updating face databases with more recent images for the success of face recognition systems [4]. Moreover to update large and complex database is also a tough task, so an alternate of recognition system must be required to confirm the identity of individuals from a pair of age separated face images. Thus in this paper we will try to achieve an Age Invariant Face Recognition system using Neural Networks. Advantage of using this method is its more accuracy as compared to other methods. Neural networks have a capability to store more features which further adds its advantages to state-of-art method.

2. Related Work

In the earlier years, age variant of face recognition systems were not used widely due to lack of suitable databases. Nowadays with the advent of MORPH and FGNET databases, this field is widely used for research purposes.

Bor-Chun Chen et al. [5] in their paper present a technique for face recognition throughout age & a dataset holding disparities of age in isolation. They utilize a data-driven technique to state cross-age FR issue, known as CARC. By leveraging a big-scale picture dataset easily obtainable, cross-age reference coding can encrypt a face image low-degree feature with a reference space of age-invariant. In recovery segment, their technique needs a direct projection to encrypt features and for this reason it's extremely scalable.

Yandong Wen et al. [6] gave a paper dealing with the use of Neural Networks for AIFR. While significant developments were made on FR, age-invariant FR nevertheless stays a main task in FR real world applications. Most important issue of AIFR ascends from the reality that facial look is immaterial to extensive intra-private adjustments because of aging process through the years. In order to cope with this issue, they advise a new deep FR framework to study age invariant deep face features by a sensibly developed CNN

model. This is a first try to expose efficiency of deep CNNs in proceeding state-of-the-art of AIFR. Detailed investigations are carried out on numerous public domain datasets of face aging to determine effectiveness of projected model over state-of-the-art. They additionally affirm outstanding generalization of their novel version of well-known LFW dataset.

Agarwal et al. [7] projected that human face is a complicated multidimensional visual version and growing a computational version for FR is hard. Authors have utilized Feed Forward BPNN for face recognition. An unsupervised PR system is unbiased of immoderate geometry & computation. Additional ANN was utilized for organization. Concept of NN is utilized due to its capability to examine its features from discovered information. ANN-based technique is impartial of any bug features decision. Self-organizing maps appear to be computationally expensive & can be replaced with Principal Component Analysis without damaging the accuracy.

Intrator et al. [8] projected a hybrid technique. They blended unsupervised strategies for feature extraction and supervised techniques for searching features to minimize errors of classification. Authors utilized FFNN for classification. Conversely, it consumed large time as compared to simple technique.

3. Research Methodology

Age invariant face recognition system can be implemented using support vector machines but deep learning provides less test error and is suitable for larger dataset. Therefore this research work is focused using neural networks and auto encoders. Steps of algorithms are as follow:

Step 1: Read the images from folders and subfolders: This is the initial step and most important step. Images of different persons need to be read and stored. Important care must be taken that algorithm must support read different format of images.

Step 2: Calculate score and coefficients of each image: After the completion of the step 1, feature extraction of each image is done and score of each image is calculate and stored in a mat file.

Step 3: Load the target and PCA Features: After the completion of step 1 and step 2, a neural network needs to be trained. For this load the target vector and PCA Features.

Step 4: Training the neural network as below.

Training: `Trainscg` is a network training function that updates weight and bias values according to the scaled conjugate gradient method. `Trainscg` can train any network as long as its weight, net input, and transfer functions have derivative functions. Back-propagation is used to calculate derivatives of performance with respect to the weight and bias variables X .

Training stops when any of these conditions occur:

- The maximum number of epochs (repetitions) is reached.
- The maximum amount of time is exceeded.
- Performance is minimized to the goal.

- Performance gradient falls below `min_grad`.
- Validation performance has increased more than `max_fail` times since the last time it decreased (when using validation).

Training the neural network with the features is done using the stacked auto-encoder and further stacked auto-encoder is used to form deep network which actually checks the accuracy of the model.

Following Steps are followed to train model and calculating accuracy

A. Input to neural network is PCA features with 100 hidden layers of auto-encoder. Auto-encoder is explained in the step2 along with its functionality.

B. **Auto-Encoder** (Encoder and Decoder Transfer Function):

Stacked Auto Encoders (SAEs): Stacked Auto Encoders are used which are fundamentally multilayer feed-forward networks, but in this model weights are initialized in different manners. Here, through a generative learning algorithm, weights initialization is achieved as this offers good initial weight parameters for the network. An auto encoder with single layer is a feed-forward network skilled to repeat the matching inputs at the output and is a tool to learn deep network. If the input to an auto encoder is a vector $x \in R^{D_x}$, then the encoder maps the vector x into another vector $z \in R^{D^1}$ is as follows:

$z^1 = h^1(W^1x + b^1)$ Where superscript (1) represents first layer.

Where Decoder maps z back into x as follows:

$\hat{x} = h^2(W^{(2)}z + b^{(2)})$ Where superscript (2) represents second layer.

Auto-encoders can be set on one another to attain more dispersed and hierarchical sign of information removal from data. These types of designs are referred as **Stacked Auto Encoders (SAEs)**. Auto-encoders are trained with the following options:

Hidden size = 100 it is the number of neurons in the hidden layer and is specified as positive number.

L2WeightRegularization = Coefficient that controls the weighting of the L2 weight regularization

Sparsity Regularization: Coefficient that controls the impact of the sparsity regularization in the cost function, specified as the comma-separated pair consisting of Sparsity Regularization and a positive scalar value.

Sparsity Proportion = Desired proportion of training examples which a neuron in the hidden layer of the auto-encoder should activate in response to. It must be in between 0 and 1. It controls the sparsity of the output from the hidden layer. A low value for Sparsity Proportion usually leads to each neuron in the hidden layer specializing by only giving a high output for a small number of training examples.

Therefore, a low sparsity proportion encourages higher degree of sparsity.

Decoder Transfer Function: Three decoder transfer function used are **logsis**, **satlin**, **purelin** and in this algorithm logsis is used which is a linear transfer function defined by $f(z) = z$.

C. Features are extracted in the hidden layer and second auto-encoder is trained using features obtained from the first encoder with 50 hidden sizes.

D. The logistic output function can only be used for the classification between two target class $t=1$ and $t=0$. This logistic function can be generalized to output a multiclass categorical probability distribution by the Softmax function. This Softmax function takes input a D-dimensional vector x and outputs a D-dimensional vector O of real values between 0 and 1. This function is a normalized exponential and is defined as

$$O_d = \sigma(x)_d = \frac{e^{x_d}}{\sum_{c=1}^d e^{x_c}} \text{ for } d=1 \dots D$$

The denominator $\sum_{c=1}^d e^{x_c}$ acts as a regularizer to make sure that $\sum_{d=1}^d e^{x_d} = 1$. As the output layer of a neural network, the Softmax function can be represented graphically as a layer with d neurons.

Probabilities of the class is $t=d$ for $d=1 \dots d=1 \dots D$ given input x as:

$$\begin{bmatrix} P(t=1|x) \\ \vdots \\ P(t=d|x) \end{bmatrix} = \begin{bmatrix} \sigma(x)_1 \\ \vdots \\ \sigma(x)_d \end{bmatrix} = \frac{1}{\sum_{c=1}^d e^{x_c}} \begin{bmatrix} e^{x_1} \\ \vdots \\ e^{x_d} \end{bmatrix}$$

Where $P(t=d|x)$ is the probability that that the class is d given the input x . Softmax layer is trained for classification using the features obtained from **Step 3** using loss function and cross entropy where loss function is error between model output and measured response and cross entropy is used to calculate neural network performance given target and outputs.

E. Two auto-encoders and Softmax layer are stacked to form deep network. It captures a useful hierarchical grouping or part whole decomposition of the input. The first layer of a stacked auto encoder tends to learn first-order features in the raw input (such as edges in an image). The second layer of a stacked auto encoder tends to learn second-order features corresponding to patterns in the appearance of first-order features (e.g., in terms of what edges tend to occur together--for example, to form contour or corner detectors). Higher layers of the stacked auto encoder tend to learn even higher-order features.

F. Deep network is trained on input and target data and accuracy is calculated using it. Fig 1.1 represents how the work progress step by step.

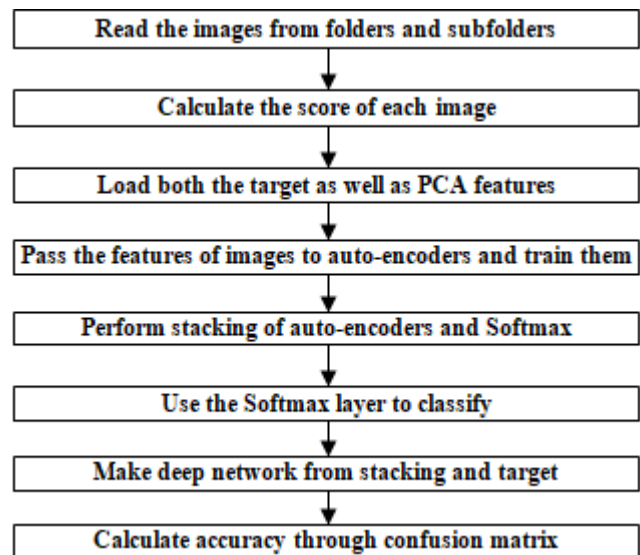


Figure 1.1: Flowchart of methodology

4. Results and Discussions

In this section, we describe our analysis of the learned representations in recognizing faces using neural network and present control experiments to understand invariance properties of the face detector neural model. All the images are true color images. The image size is **250*250** therefore the dimensionality of input vector is **62500**. Images are taken from Cross-Age Celebrity Dataset (CACD).

Standard back-propagation (BP) remains a popular supervised learning algorithm to train feed forward neural network, but if the exact global minimum for pattern classification is to be found, conjugate methods are more superior, therefore above discussions signifies importance of conjugate gradient algorithm.

During first phase of our training, input containing 100 hidden layers is passed to an auto encoder comprising encoder and decoder. Features obtained from auto encoder 1 are passed to second with 50 hidden layers.

Encoder takes the input $x \in R^\theta = X$ and maps it to $z \in R^\theta = F$ where $z = \sigma(wx + b)$ where z is the latent representation and σ is an activation function $f(x)$, w is the weight matrix and b is bias vector.

Phase I

Performance is calculated using Mean Square error with L2 and Sparsity Regularizer. The purpose of passing the input through auto-encoder is to reduce the dimensions. *Table 1* shows the parameters used during phase 1.

- **Epochs:** It is a measure of the number of times all of the training vectors are used once to update its weight
- **Validation Check:** The number of validation checks represents the number of successive iterations that the validation performance fails to decrease and is used to terminate the training.
- **Gradient:** This method learns a parameter's value by understanding how a small change in the parameters value will affect the network output.
- **Performance:** This checks the efficiency of the neural network during the training period.

Table 1: Parameters Used During Auto-Encoder 1

Sr. No	Parameters	Obtained Value
1	Epochs	400
2	Time	10
3	Performance	9.18 e+0.5
4	Gradient	28.6
5	Validation Check	0

Table 2: Parameters Used During Auto-Encoder 2

Sr. No	Parameters	Obtained Value
1	Epochs	100
2	Time	2
3	Performance	0.0531
4	Gradient	0.00366
5	Validation Check	0

Phase II

After passing the input from the encoder and decoder the resultant neurons are passed through Softmax layer and got reduced to 35 in number. Softmax layer is used to highlight the largest value and suppress values which are significantly below the maximum value. Table 2 represents the parameters used in the phase 2. Softmax layer is used in the output layer and not in hidden layer because of the following reasons:

- a) **Variable independence:** A lot of regulations and efforts are needed to keep variables independent, uncorrupted and sparse. Using Softmax in the hidden layer makes all nodes linearly dependent which may results in poor generalization.
- b) **Training issue:** Mathematical issue, accuracy and speed are other parameters which signify the use of softmax layer in the output layer

During phase II, Performance is calculated using cross-entropy whereas training is continued using scaled conjugate gradient.

Table 3 shows algorithm used whereas Table 4 represents progress of work for training the machine for character recognition.

Table 3: .Algorithm Used

Sr. No	Name of Algorithm
1	Training (scaled conjugate gradient)
2	Performance (cross entropy)
3	Calculation (MEX)

Table 4: Progress

Sr. No	Parameters	Obtained Value
1	Epochs	400
2	Time	3
3	Performance	0.105
4	Gradient	8.15e-0.5
5	Validation Check	0

Some important discussion

- 1) Scaled conjugate gradient (SCG) algorithm performance does not degrade when the error is reduced.
- 2) The validation error normally decreases during the initial phase of training as does the training set error. However, when the network begins to over fit the data, the error on the validation set typically begin to rise.

- 3) If the error in the test set reaches minimum at a significantly different iteration number than the validation set error, this indicates a poor division of data set.

Phase III

Confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one (in unsupervised learning it is usually called a matching matrix). Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class (or vice versa). It will give the accuracy of the neural network which is depicted in the following Fig. 1.2.

5. Conclusion

In this paper, we have implemented face recognition age invariant using Deep Belief Networks on CACD (Cross Age Celebrity Dataset) dataset. In this work, deep neural network auto encoder is used. The advantages of auto encoder are as below:

- As compared to Convolutional Neural Network, the number of hidden unit in auto encoder is much less which enables it to generate a low dimensional feature representation.
- It finds the most efficient compacted representation for the input data.

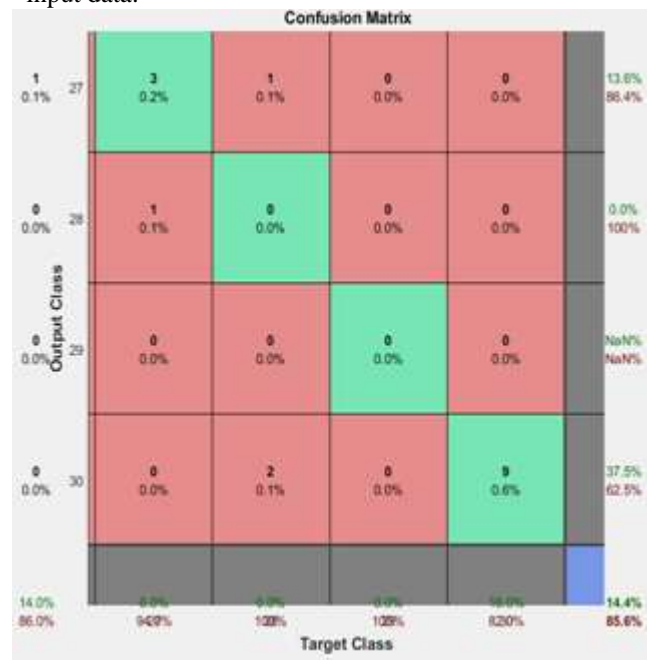


Figure 1.2: Confusion matrix

Auto encoder is a three layer vanilla where output unit is directly connected to the input unit. One major application of auto encoder is it provides the weights to deep networks which are a better choice as compared to randomly initialized weights. After training the model it is tested on the target classes. Even the calculation time reduces because of its dimensional reduction technique. From the simulation results it is found that this model provided 87% accuracy on the CACD dataset.

References

- [1] Singh, WangjamNiranjan, DipankarMedhi, and Mrudula S. Nimbarte. "Study on Age Invariant Face Recognition System",IOSR Journal of Computer Science (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727 PP 68-73
- [2] Park, Unsang, Yiyong Tong, and Anil K. Jain. "Age-invariant face recognition." *IEEE transactions on pattern analysis and machine intelligence* 32.5 (2010): 947-954.
- [3] Nana RachmanaSyambas, UntungHariPurwanto. "Image Processing and ace Detection Analysis on Face Verification Based on the Age Stages",7th International Conference on Telecommunication Systems, Services and Applications, IEEE, 2012.
- [4] Felixluefei-Xu, Khoa LUU, MariosSavvides, Tien D. Bui and Ching Y. Suen. —Investigating Age Invariant Face Recognition Based on Periocular Biometricsl IEEE, 2011
- [5] Chen, Bor-Chun, Chu-Song Chen, and Winston H. Hsu. "Face recognition and retrieval using cross-age reference coding with cross-age celebrity dataset." *IEEE Transactions on Multimedia* 17.6 (2015): 804-815.
- [6] Wen, Yandong, Zhifeng Li, and Yu Qiao. "Latent factor guided convolutional neural networks for age-invariant face recognition." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016.
- [7] M. Agarwal, N. Jain, M. Kumar and H. Agrawal, Face recognition using eigen faces andartificial neural network, *Int. J. Comput. Theor.Eng.* 2(4) (2010) 624629.
- [8] N. Intrator, D. Reisfeld and Y. Yeshurun, Face recognition using a hybrid supervised/unsupervised neural network, *Pattern Recogn. Lett.* 17 (1995) 67- 76.
- [9] K.-C. Kwak and W. Pedrycz, Face recognition using fuzzy integral and wavelet decomposition method, *IEEE Trans. Syst., Man, Cybern. B, Cybern.* 34(4) (2004) 16661675.
- [10] X. Jing and D. Zhang, Face recognition based on linear classifiers combination, *Neuro-computing* 50 (2003) 485488.
- [11] A. Sinha and K. Singh, The design of a composite wavelet matched filter for face recognition using breeder genetic algorithm, *Opt. Lasers Eng.* 43 (2005) 12771291
- [12] A. Bouzalimat, N. Belghini, A. Zarghili, J. Kharroubi and A. Majda, Face recognition using neural network based fourier Gabor filters & random projection, *Int. J. Comput. Sci. Secur.* 5(3) (2011) 376.
- [13] A. Khatun and Md. Al-Amin Bhuiyan, Neural network based face recognition with Gabor filters, *Int. J. Comput. Sci. Netw. Secur.* 3(5) (2011) 376386.
- [14] C. A. Perez, L. A. Cament and L. E. Castillo, Methodological improvement on local Gabor face recognition based on feature selection and enhanced Borda count, *Pattern Recogn.* 44 (2011) 951963.
- [15] W. Yang, H. Yan, J. Wang and J. Yang, Face recognition using complete Fuzzy LDA, face recognition using complete fuzzy LDA, in *Proc. 19th Int. Conf. Pattern Recognition* December 2008, pp. 14.
- [16] A. Nefian and M. Hayes, Hidden markov models for face recognition, in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, ICASSP'98, Vol. 5, Washington, USA, May 1998, pp. 27212724.*
- [17] W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, Face recognition: A literature survey, *ACM Comput. Surv.* 35(4) (2003) 399458.
- [18] S. T. Gandhe, K. T. Talele and A. G. Keskar, Intelligent face recognition techniques: A comparative study, *GVIP J.* 7(2) (2007) 5360.
- [19] M. Kirby and L. Sirovich, Application of the Karhunen-Lo`eve procedure for the characterization of human faces, *IEEE Trans. Pattern Anal. Mach. Intell.* 12(1) (1990) 103108.