Design of a Solution for a Biometric Face Recognition Task

Kshitiz Bajagain [#1],Swati Sah [#2]
[#]*Patan College for Professional Studies.* in collaboration with University of Bedfodshire,U.K

[1]drswatisah@gmail.com, [2]drswatisah@gmail.com

***Abstract*—This Biometric Face Recognition is technology which uses modern machine algorithms and techniques to identify face of specific individuals under different circumstances. Face recognition is famous as well as leading problem in machine learning. The best solution to solve this problem was to develop a Biometric Face Recognition System which can give robust solution and maximum accuracy during recognition of specific face. Different machine algorithms are deployed during development of this project in order to achieve maximum accuracy**

Keywords—component, Face Recognition, Supervised Learning, Classification Problem, Artificial Neural Network (ANN), Principal Component Analysis (PCA), Validation Ratio

I. INTRODUCTION (*HEADING 1*)

In the last few years, the growth of Artificial Intelligence has increased rapidly. It had made the life of human beings to do the tasks easier in different ways. From checkers playing games to self-driving cars, AI has been progressing ahead rapidly. Beside these, the growth of biometric has been very significant in last few years. The most common biometrics modes used for recognition is face that is used in applications at a distance, security scenarios and video games. (Vera-Rodriguez, et al., 2000).

Face recognition is the process of identifying or verifying the identity of a people using their face from the photos, video, or in real-time (Electronic Frontier Foundation, 2019). The concerned on facial recognition has significantly increased in image processing and computer vision that are used in biometrics, information security, video surveillance, law enforcement, identity authentication, smart cards, and access control system. Generally, the biometrics software representing the facial recognition maps the face mathematically and stores the data as a face print. (Techtarget, 2018)..

II. Design and Implementation

A. Reading an Image set

The main aim of reading the images is to display the images that are available in the dataset. 50 images of 30 persons each were given in the data set. So, there were total 1500 images available.

So, when is code is executed, it gives an output of 1024 x 1500 single matrix which is the vectorized form of 1500 images stored in the variable "ans" in the workspace.

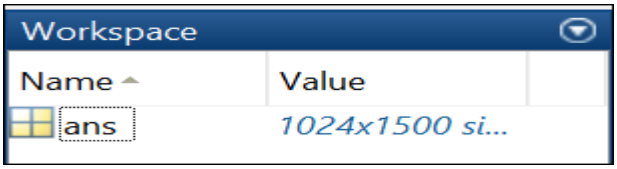


Figure11 Matrix value of images stored in variable "ans"

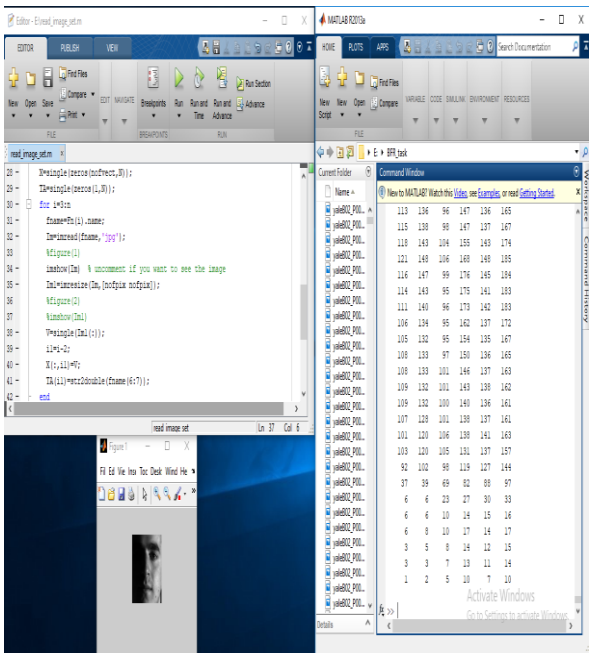


Fig:2 Reading Image Dataset and analysis of given image dataset

When you analyze the above value, it ranges from 0 to 255 which is the brightness of the pixels of the images.

Since the main aim of read image set is to display the images that are available in the dataset.

```
%figure(1);
%imshow(Im);  % uncomment if you want to see the image
Im1=imresize(Im,[nofpix nofpix]);
%%figure(2);
%imshow(Im1);
```

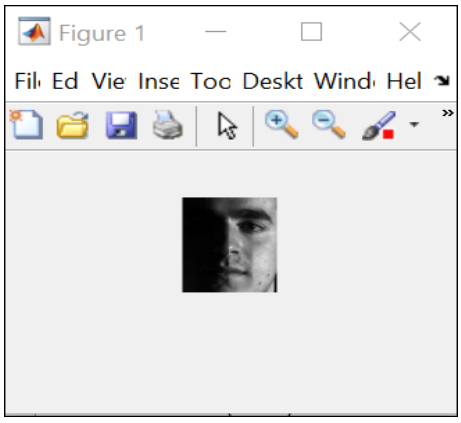


Figure 5 Image at its default pixels

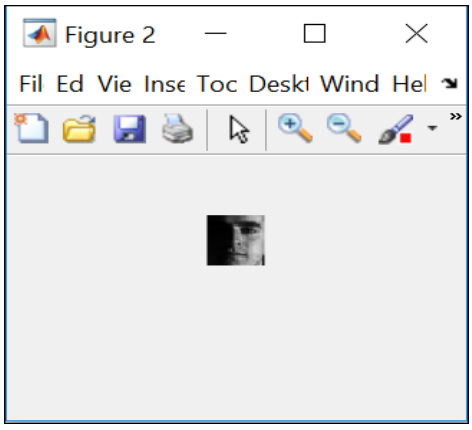


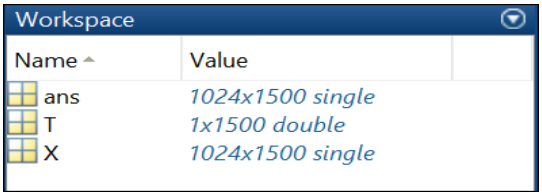
Figure 6 Image after resizing pixels

There are 3 main commands that functions to read the images. They are:

1. Imread: It reads a greyscale or color image from the file of the image type 'jpg'.
2. Imshow: It display the image in the picture box.
3. Imresize: It resizes the images into given pixels.

   In the scripts, the images are resized into 32*32 pixels which create smaller vector and use less processing power for feature extraction.

   After executing the script of read image set we get the output of X and T in the workspace as follows:



X is the matrix of vectorized images of the persons and T is the matrix with the picture labels for identification purposes.

close this file and download the Microsoft Word, Letter file.

B. Split Image Set

The Splitting Image dataset is important process for BFR task. Splitting Image into individual elements is necessary for comparing each unknown elements with already known labelled elements (output of read_image_set ( )) which will ensure the similarities between two images finally resulting into high accuracy. Eigen Face method was used in this assignment to split image dataset .Eigen Face is a set of eigenvectors which is commercially used in face recognition technology. The test image is transformed into its Eigen face components. First we compare line of our input image with our mean image and multiply their difference with each eigenvectors. (Shemi P M, Ali MA, 2012) During execution of split image algorithms, Training set, .validation set and testing set was calculated and stored in variable as:

X1 = L by n1 matrix of training set

X2 = L by n2 matrix of validation set

Similarly, element of target vector of input images in validation and training set was also calculated and stored in variable as:

T1 = n1 element target vector of input images in validation set

T2 = n2 element target vector of input images in validation set

X1, X2, T1 and T2 were the final outputs of split_image_set ( ) function which are needed further in applying PCA and training portion.

For one input,

MN* 1 vector = 1024 * 1

For 1500 input,

MN * 1500 vector = 1024*1500

As given image dataset contains 32*32 pixels 1500 image from 30 person (50 each) .Means,

M = N = 32

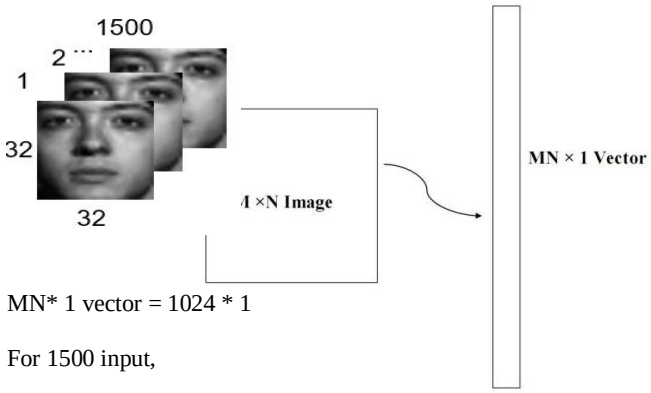
Np = 30

K = 50

N = Np * K = 1500

Total elements = 1500*32*32

= 1,536,000 your paper and style the



MN* 1 vector = 1024 * 1

For 1500 input,

MN * 1500 vector = 1024*1500

As given image dataset contains 32*32 pixels 1500 image from 30 person (50 each) .Means,

M = N = 32

Np = 30
K = 50
N = Np * K = 1500
Total elements = 1500*32*32
= 1,536,000

The main of split image is to divide the image dataset into two parts:

Part 1: Used to train the Artificial Neural Network(ANN)

Part 2: Used for validating the accuracy.

In the split image script, the value of val_ratio is 0.3 which is the default value for the experiments which means 30% of the images will be reserved for validation and 70% images will be used for training the neural network.

When the script is executed we get the output as:

Workspace

Name	Value
ans	1024x1500 single
T	1x1500 double
T1	1x1050 double
T2	1x450 double
X	1024x1500 single
X1	1024x1050 single
X2	1024x450 single

text.

All margins, column widths, line spaces, and text fonts are prescribed; please

C. Split Image Set

Principal component analysis (PCA) is a technique used to emphasize variation and bring out strong patterns in a dataset. It's often used to make data easy to explore and visualize (Lehe, 2015). It is an approach that is used for reducing the number of variables in face recognition (Paul & Sumam, 2012)

After completion of splitting image by Eigen Face Method , Principal Component Analysis(PCA) is executed to training image set (X1 and X2) obtained from split_image_set( ) function . Image data are processed efficiently when they are represented as low-dimensional vectors. Principal component analysis (PCA), allowing data to be represented in a low dimensional space of principal components,is a common technique for image representation in face recognition systems (Uglov, J. et al. 2007). Every image in the training image set is represented as a linear combination of weighed Eigen vectors which are already obtained from co-variance matrix of a training image set calculated from split_image_set( ) function. PCA was applied to the training image set X1 and X2 to find out the component of P1, P2, and matrix of coefficient C and M. After applying PCA to training image set , matrix of components for training set were calculated and stored in PS1(nofpc by n1) and PS2(nofpc by n2 do not alter them.

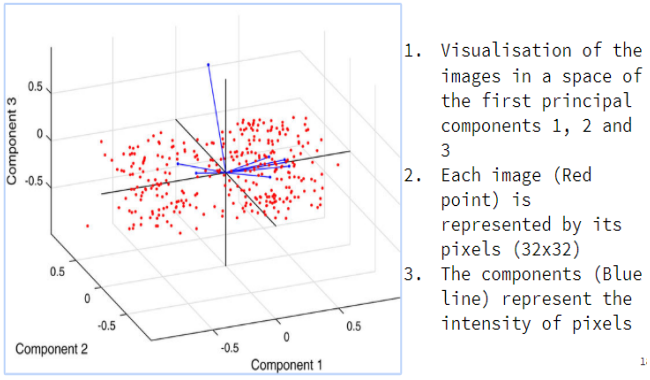


Fig 8: Low Dimensional Space of Principal Components

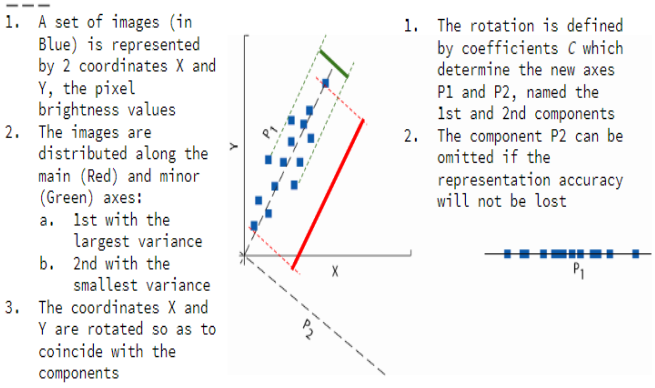


Fig: Above figure illustrates how PCA can reduce dimensionality of Image Data (2D Case) (Schetinin, 2019)

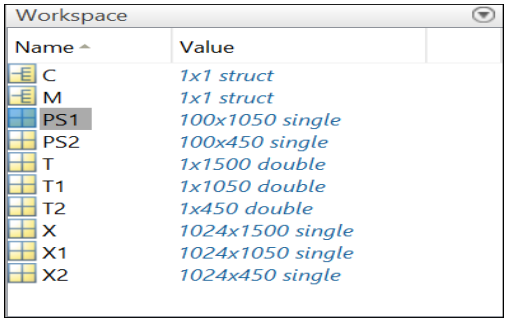


Figure 9 Output of script apply_pca

The new output variable after executing the scripts are C, M, PS1 and PS2.

Here PCA finds the components P1 and matrix of coefficient C.

PS1: 100x 1050 matrix of components for training set.

PS2: 100x 450 matrix of components for training set.

your paper as

one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

D. Reading an Image set

Training is most important for classification problem. As Artificial intelligence (A.I.) is the study of systems that act in a way that to any observer would appear to be intelligent. (Coppin, 2004). Training neural network with specific settings (altering weight, no. of hidden layers, validation ratio, and no. of PCA components) will dynamically evolve machine algorithms used in BFR to meet the criteria of self-intelligent A.I. principles. Scaled Training Conjugate (trainscg) process was used in training. Trainscg is a network training function that updates weight and bias values according to the scaled conjugate gradient method. To make our algorithms to behave as artificially intelligent, we need to analyse and train our machine algorithms as many times as possible. Training PCA executed outputs to analyse the result of assignment and comparing all those results (accuracy) to create confusion (matching) matrix which will be crucial for further improvement in algorithms used.

Designing of Biometric Face Recognition Task

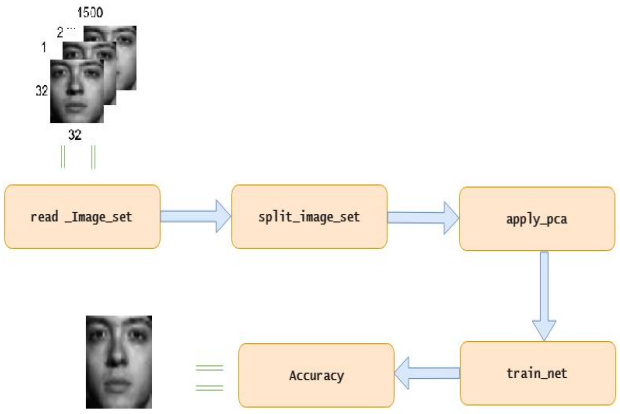
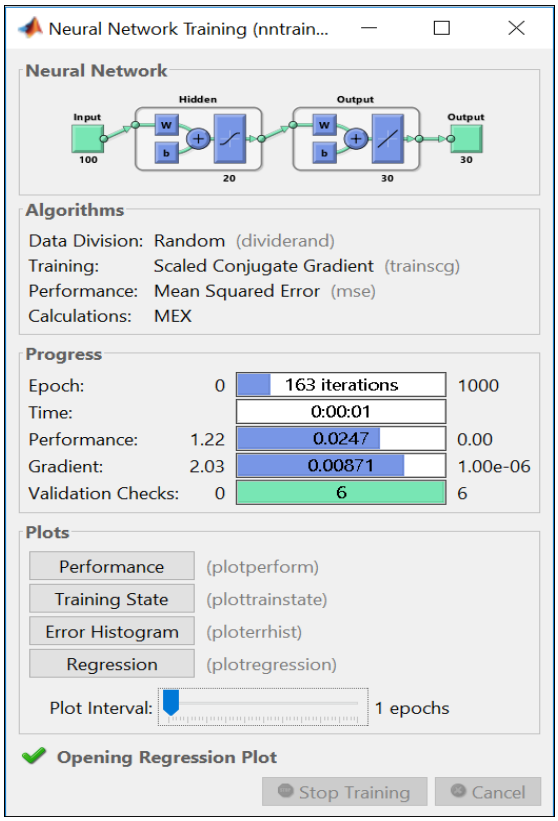


Fig: Overall Process of Designing a solution

save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.



This tool is used for monitoring the training process and it also helps to visualize the effects for every change in variables.

The algorithms that is used for training is Scaled Conjugate Gradient (trainscg) that updates weight and bias values (MathWorks, 2019). During training, the progress is constantly updated in the training window. The gradient value and the number of validation checks terminates the training. As the training reaches a minimum of the performance, the gradient value will become small. If the

magnitude of the gradient is less than 1e-5, the training will stop. The number of validation checks represents the number of successive iterations. The training will stop if the number reaches to 6(default value). Stop Training button can be used to stop the training (MathWorks, 2019).

From the training window, you can access four plots: performance, training state, error histogram, and regression.

Performance plot: It shows the value of the performance function versus the iteration number. It plots training, validation, and test performances.

Training state plot: It shows the progress of other training variables, such as the gradient magnitude, the number of validation checks, etc.

Error histogram plot: It shows the distribution of the network errors.

Regression plot: It shows a regression between network outputs and network targets. You can use the histogram and regression plots to validate network performance (MathWorks, 2019)

III. EXPERIMENTS

Keep your text and graphic files separate until after the

Test	nofpc(default=100)	val_ratio	nofhn	Accuracy
1	100	0.3	20	0.800
2	100	0.3	20	0.844
3	100	0.3	20	0.802
4	100	0.3	20	0.791
5	100	0.3	20	0.831
6	100	0.3	20	0.809
7	100	0.3	20	0.878
8	100	0.3	20	0.829
9	100	0.3	20	0.809
10	100	0.3	20	0.820
11	100	0.3	20	0.813
12	100	0.3	20	0.800
13	100	0.3	20	0.871
14	100	0.3	20	0.824
15	100	0.3	20	0.776
16	100	0.3	20	0.787
17	100	0.3	20	0.791
18	100	0.3	20	0.816
19	100	0.3	20	0.820
20	100	0.3	20	0.813
21	100	0.3	20	0.760
22	100	0.3	20	0.833
23	100	0.3	20	0.827
24	100	0.3	20	0.840
25	100	0.3	20	0.864
26	100	0.3	20	0.796
27	100	0.3	20	0.847
28	100	0.3	20	0.782
29	100	0.3	20	0.807
30	100	0.3	20	0.853

Maximum of 0.878(87.8%) accuracy was achieved while executing given ANN with default settings.

3.1.1. Changing default nofhn keeping nofpc constant(Series-2)

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract.

Nofpc	nofhn(default=20)	Accuracy
100	20	0.844
100	50	0.891
100	100	0.900
100	150	0.913
100	180	0.923
100	200	0.933

Changing default nofpc keeping nofhn constant (Series-3):

nofpc(default=100)	nofhn	Accuracy
80	20	0.800
120	20	0.862
150	20	0.824
180	20	0.822
200	20	0.816
220	20	0.842

Changing both default nofpc and default nofhn keeping val_ratio constant(Series-4) :

nofpc(default=100)	val_ratio	nofhn(default=20)	Accuracy
80	0.3	30	0.909
120	0.3	50	0.936
150	0.3	100	0.931
150	0.3	150	0.942
200	0.3	150	0.929
200	0.3	200	0.942

Every series has 6 set of different parameters except default setting (30 test taken).

Changing all default nofpc and default nofhn and default val_ratio(0.3) (Series-5)

nofpc	val_ratio	nofhn	Accuracy
150	0.22	150	0.968
150	0.15	150	0.952
100	0.112	180	0.946
150	0.10	150	0.960
190	0.298	110	0.926
180	0.169	125	0.957
150	0.30	150	0.942
170	0.289	125	0.926
150	0.345	150	0.817
125	0.369	145	0.762
150	0.40	150	0.802
165	0.412	132	0.825
150	0.16	150	0.948
185	0.145	122	0.940
150	0.15	200	0.933
200	0.16	220	0.953

130	0.28	190	0.950
160	0.123	200	0.946
190	0.21	210	0.952

Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

A. Units

```
>> main
Accuracy of recognition on validation subset 0.968
```

Use Again, one set of parameters from every series except default setting series-1 was taken and executed for 10 times to see the fluctuation on accuracy and performance.

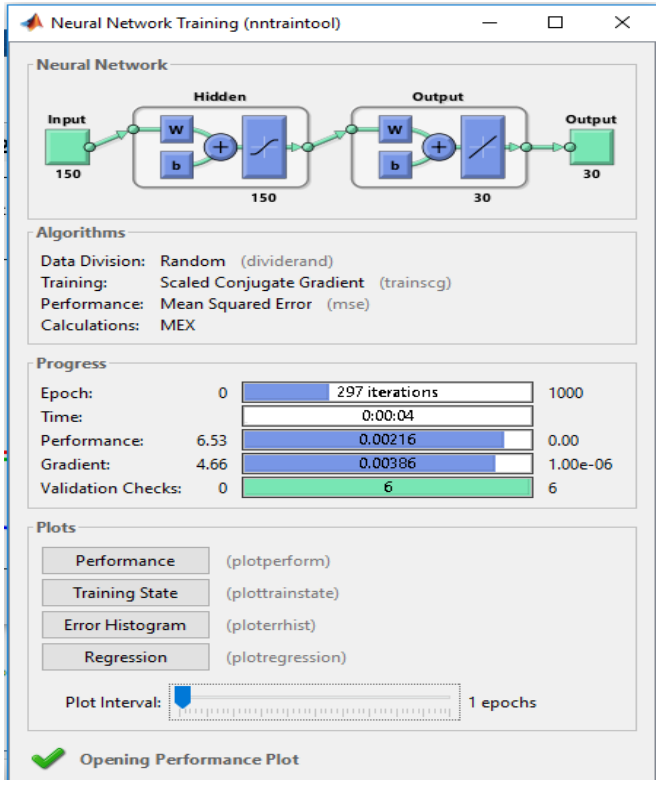
B. Equations

The

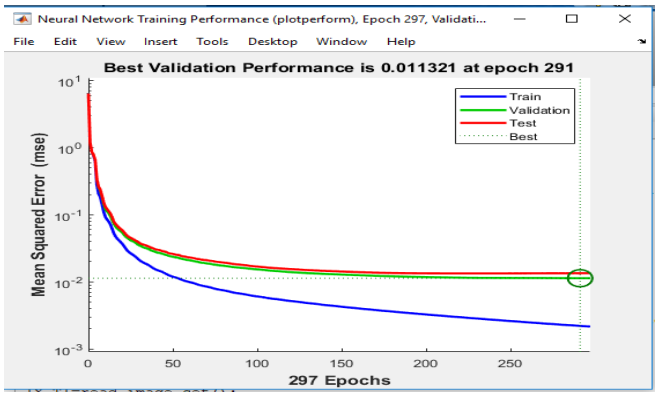
When nofpc=150, nofhn=150 and val_ratio = 0.22, following results were achieved.

```
>> main
Accuracy of recognition on validation subset 0.968
```

equations are an exception to the prescribed

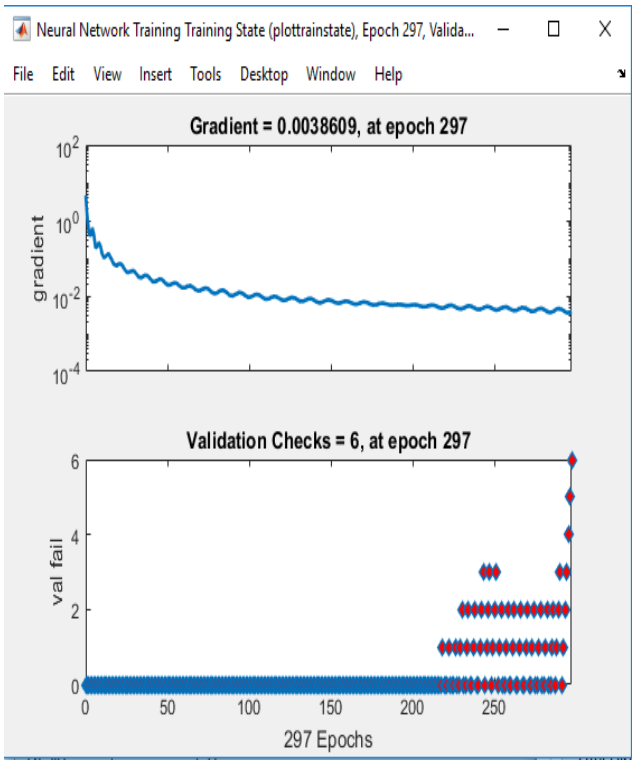


Best Validation Performance



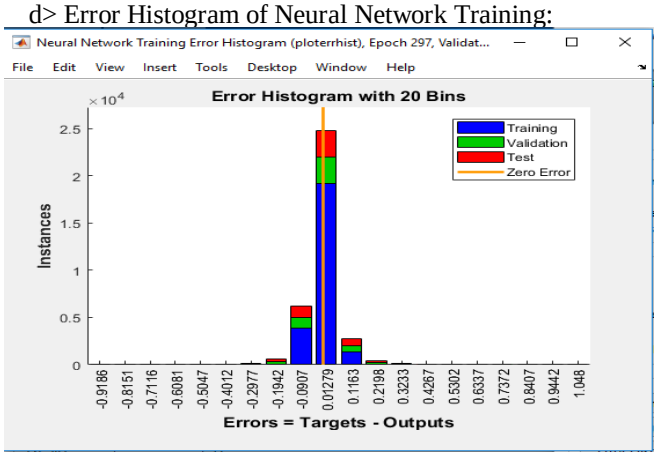
specifications of this template. You will need to

> Neural Network Training State (plottrainstate)



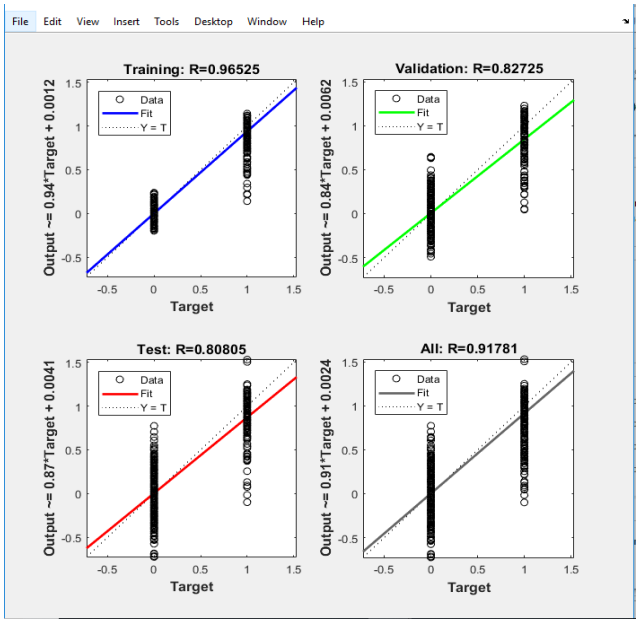
determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font).

d> Error Histogram of Neural Network Training:



e>Neural Network Training Regression:

To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.



IV. CONCLUSION

Biometric Face Recognition is Supervised Learning based classification problem where validation image dataset is used as valid (template) image for recognizing given input images. At first given image dataset was read by running read_image_set ( ) to obtain X and T. And further to obtain validation (training) image set, split_image_set ( ) function was run and X1, X2, T1 & T2 were the matrix obtained used further for applying PCA and training portion. After splitting PCA was applied which helped to reduce dimension because low dimension principal components PS1 and PS2 will make face recognition more convenient and robust. Finally, training ANN was carried out by using transcg training method. After training with different set of parameters, analysis of result (accuracy) was figured out. Analysis of result is shown below:

a. Executing the same ANN with nofpc (100), nofhn (20) and val_ratio (0.3) several times, average of 81.8~ 82% was found.
b. Executing the same ANN by changing only nofhn (No. of hidden networks) keeping all default, accuracy got increased on increasing nofhn.
c. Executing the same ANN by changing only nofpc (No. of PCA) keeping all default, accuracy got increased at first and decreased on increasing nofpc.
d. Executing the same ANN by changing all nofhn, nofpc and val_ratio keeping all default, maximum accuracy of 0.968(96.8%) was achieved when nofpc=150 ,nofhn=150 and val_ratio = 0.22.

REFERENCES

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first ..."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

- Techtargetcom. 2019. SearchEnterpriseAI. [Online]. [17 February 2019]. Available from: https://searchenterpriseai.techtarget.com/definition/facial-recognition
- Shemi P M, Ali M A, A Principal Component Analysis Method for Recognition of Human Faces: Eigenfaces Approach, International Journal of Electronics Communication and Computer Technology (IJECCT),Volume 2 Issue 3 (2012).
- Uglov J. et al. 2007. Comparing Robustness of Pairwise and Multiclass Neural-Network Systems for Face Recognition.EURASIP Journal on Advances in Signal Processing. 116(1), pp. 1-5.
- Coppin, B (2004). Artificial Intelligence Illuminated. (1st ed.). United States of America: Jones and Bartlett Publishers.
- Popova E. et al. 2018 Deep Neural Networks for Face Recognition: Pairwise Optimisation
- Schetinin, V. (2019) 'Principal Component Analysis for Face Recognition' [PowerPoint presentation]. Available at: https://drive.google.com/open?id=1wotwxmLj3xBvzdvuBQHV6tr7oC3B_27kHIa7cLWacb0 (Accessed: 17 February 2019).
- Machinelearningmasterycom. 2016. Machine Learning Mastery. [Online]. [2 March 2019]. Available from: https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/
- Epifano, C. M. & Quaglia, A., 2012. Face Recognition: Methods, Applications and Technology, s.l.: Nova Science Publishers.

- Caudill, M., 1989. Neural Network Primer: Part I. s.l.:s.n.
- Dans, E., 2018. Forbes. [Online] Available at: https://www.forbes.com/sites/enriquedans/2018/06/25/facial-recognition-and-future-scenarios/#72a393811ac9 [Accessed 17 February 2019].
- Electronic Frontier Foundation, n.d. Facial Recognition. [Online] Available at: https://www.eff.org/pages/face-recognition[Accessed 13 February 2019].
- Lehe, L., 2015. Principal Component Analysis. [Online] Available at: http://setosa.io/ev/principal-component-analysis/[Accessed 23 February 2019].
- MathWorks, 2019. Scaled Conjugate Gradient Backpropagation- MATLAB trainscg. [Online] Available at: https://www.mathworks.com/help/deeplearning/ref/trainscg.html [Accessed 24 February 2019].
- MathWorks, 2019. Train and Apply Multilayer Shallow Neural Networks. [Online] Available at: https://www.mathworks.com/help/deeplearning/ug/train-and-apply-multilayer-neural-networks.html [Accessed 24 February 2019].
- Paul, L. C. & Sumam, . A. A., 2012. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET). Face Recognition Using Principal Component Analysis Method, I(9), p. 135.
- Techtarget, 2018. Facial Recognition. [Online] Available at: https://searchenterpriseai.techtarget.com/definition/facial-recognitio [Accessed 13 February 2019].
- Vera-Rodriguez, R., Fierrez, J., Tome, P. & Ortega-Garcia, J., 2000. Advances in Intelligent and Soft Computing, s.l.: Springer.