Bioinformatics, Proteomics and Immaging Analysis

OMMEGA Publishers
Answering Intelligence
www.ommegaonline.org

Research Article

OPEN ACCESS

# A MATLAB-Based Convolutional Neural Network Approach for Face Recognition System

**Syazana-Itqan K, Syafeeza A.R* and Saad N.M**

Faculty of Electronic and Computer Engineering, Universiti Teknikal Malaysia Malacca, Malaysia

***Corresponding author:** Syafeeza, A.R., Faculty of Electronic and Computer Engineering, Universiti Teknikal Malaysia Malacca, Malaysia, E-mail: syafeeza@utem.edu.my

## Abstract

The research on face recognition still continues after several decades since the study of this biometric trait exists. This paper discusses a method on developing a MATLAB-based Convolution Neural Network (CNN) face recognition system with Graphical User Interface (GUI) as the user input. The proposed CNN has the ability to accept new subjects by training the last two layers out of four layers to reduce the neural network training time. The image preprocessing steps were implemented in MATLAB, while the CNN algorithm was implemented in C language (using GCC compiler). The main purpose of this research is to develop a complete system of face recognition. A Graphical User Interface (GUI) in MATLAB links all the steps starting from image preprocessing to face identification process. Evaluation was carried out using the images of 40 subjects from AT & T database and 10 subjects from JAFFE database producing 100% accuracy with less than 1 minute average training time when inserting 1 to 10 new subjects into the system.

**Received date: August 05, 2015**
**Accepted date: January 29, 2016**
**Published date: February 01, 2016**

**Citation:** Syafeeza, A.R., et al. A MATLAB-Based Convolutional Neural Network Approach for Face Recognition System (2016) Bioinfo Proteom Img Anal 2(1): 71-75 .

**Keywords:** Face recognition; Convolution neural network; MATLAB; Graphical user interface

**DOI:** 10.15436/2381-0793.16.009

## Introduction

Now a day, people are using combinations of alphabets and numbers as their secret code to access their account. Although the passwords are unique, the safety cannot be guaranteed as it can easily be forgotten or stolen by identity fraud criminals. Biometrics is a method to identify individuals by unique biological traits that individual possesses such as features of face, finger print and finger vein, iris, blood type, DNA and many more. Recognition using these biometric traits provides high security since an account cannot be accessed by any other individuals without the presence of the account owner. Face recognition identify individuals by the features of the face i.e., eyes, nose, mouth, cheekbones, jaws or skin color. Face recognition remains a challenging biometric problem since no technique could provide a robust solution to all situations such as variation of facial expression, pose invariant, illumination invariant, and occlusion among others. Biometric has a wide range application especially in surveillance, security monitoring, immigration, any other applications for identification and recognition[1].

Neural network is a robust machine learning method in pattern recognition field. Neural network has the ability to deal with non-linear problems of data samples. Typical neural network technique is multilayer perceptron (MLP). However, designing a pattern recognition system using this type of neural network ends up with massive interconnection nodes that produce a totally flat structure with inputs are fully connected to the subsequent layers in the architecture. In addition, the topology of the input data is completely ignored, yielding similar training results for all permutations of the input vector[2].

Another variant of MLP is called Convolutional Neural Network (CNN) that is proposed by Yann Lecun in 1998 through LeNet-5 architecture. CNN has been applied to wide range of applications including face detection[3] and recognition[2], gender recognition[4], object recognition[5], etc. CNN method has been used as a face recognition technique through several existing works. Most of the works produce complex CNN design that incurs in additional cost and increasing the training time.

There are few existing works of face recognition using CNN. In 2012, Cheung proposed a 6 layers of CNN[6] and used CAPTCHA database with 10 subjects. This CNN is more complex as it has more than 5 layers. Meanwhile, Khalajzadeh et al. proposed a 4 layers CNN in 2013[2]. The work is done based on AT&T database with 40 subjects. It however, has low accuracy. The remainder of this paper is organized as follows. Section 2 discusses on the related theories. This is followed by the proposed methodology and system development in Section 3. Experimental results and analysis are discussed in Section 4. Ultimately, the final section concludes the work.

## Theory

### The difference between face verification, identification and authentication

Verification is a process to determine whether the two inputs of the system belong to the same group or identity[7]. Face verification is the process of verifying whether two faces are of the same person or not. There are some challenges in this process, such as variation of pose, hairstyle and face expression[8]. There are two types of face verification approach which are face matching and face representation[9]. The face verification involves two stages of classifier. When two faces are set as input for verification, the classifiers are applied to each face. The outputs of this classifier are used as features for the second-stage classifier. At the second stage, the 'same or different' verification decision occurs[10].

Identification is the process of providing a user identity, which normally provided in the form of user ID. A face identification is a technique to identify a person based on his physical characteristic and/or personal unique traits[11]. Face Authentication is the process of determining and validating user identity. Authentication always considered has two phases; which are identification and authentication. It verifies user-provided evidence to ascertain claimed user identity.

### Face recognition

Face recognition method is a method of identifying an individual based on the biological features of that person. This method, however, has many challenges.

**Illumination invariant:** The illumination invariant that might due to the direction of light source, will affects the brightness of an image. Ambient illumination causing performance degradation in face recognition[12].

**Facial expression:** Facial expression is an expression of one's emotions. Facial expression is to display messages and sign judgement. For example, anger may cause a frown that pulls the eyebrows closer together. A good face recognition algorithm must be able to recognize with variability of facial expression[13].

**Pose invariant:** Invariant in facial pose will cause some feature of an individual's face to be occluded in the image[14].

**Partial occlusion:** Partial occlusion means presence/absence of structural component. The structural components are challenging factor because they have variability of size, shape and color. For example moustaches, beards and spectacles[14].

There are three different approaches for existing face-recognition which are holistic matching (appearance-based) approach, feature-based (structural) approach and hybrid approach[15]. These approaches are differentiated based on the method of feature extraction[14].

i. In appearance-based approach, the whole face is the input data to the face recognition system. There are a number of methods categorized under this approach, including eigenface, frequency domain, fisherface, support vector machines, independent component analysis (ICA), Laplacian and probabilistic decision based neural network method[14].

ii. In feature-based approach, the features of face; for example nose and eyes are segmented, and used as input data[1]. There are a number of methods lie under this category, including geometrical feature[16], elastic bunch graph matching (EBGM)[17], and convolutional neural network (CNN)[18].

iii. In hybrid approach, combines both appearance-based approach and feature-based approach. In this approach, both features of face and the whole face are taken into account as the input to the system. For example combining a convolutional neural network (CNN) and a logical regression classifier (LRC)[19].

### Convolutional neural network method

Convolutional Neural Network is a unique method. It combines segmentation, feature extraction and classification in one processing module. Most of CNN design is derived from LeNet-5. LeNet-5 consists of 7 layers that is formed by 4 feature extraction layers and 3 layers of MLP. Feature extraction layer consists of convolution and subsampling layers. Convolution layer removes noise and detect lines, borders or corners of an image. In subsampling layer, it reduces the resolution of an image to prevent image distortions. CNN has built-in invariance as compared to typical neural network (MLP). CNN is superior in producing high accuracy in identification process although the algorithm is complex. In training the network, LeNet-5 applies Stochastic Diagonal Levenberg Marquadt (SDLM) learning algorithm. Unlike other neural network, when other complex database is applied, the system has to go through minimal redesigning process[20].

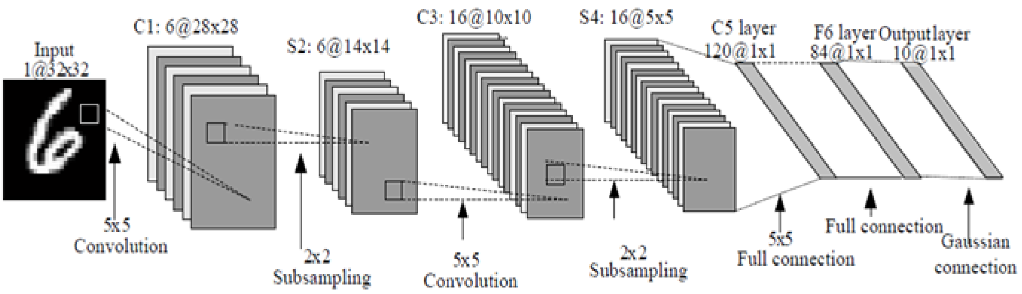


**Figure 1:** Example of LeNet-5 CNN architecture.

## Methodology

### Fusion of convolution and subsampling layers

In this paper, simplified version of CNN is used which is computationally efficient. The convolution and sub sampling layer are fused together as shown in Figure 2(b). Figure 2(a) shows the normal convolution and sub sampling approach.

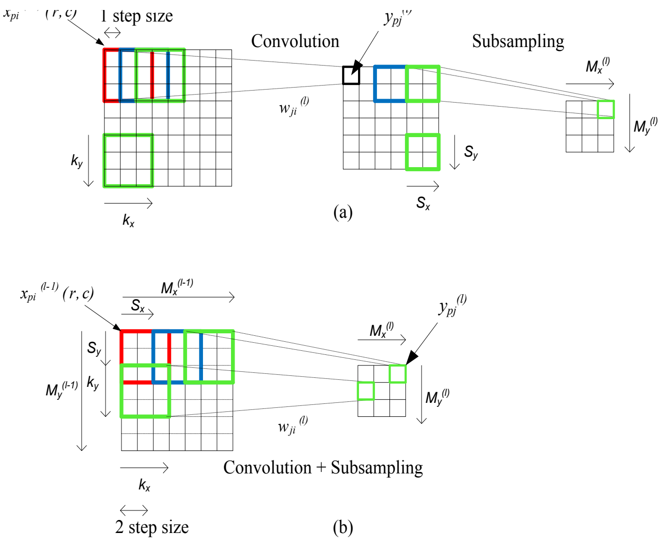


**Figure 2:** (a) Formation of convolution and subsampling layers without fusion (b) Fusion of convolution and subsampling layers

### CNN architecture and training process

The proposed CNN architecture proposed in Figure 3 consists of only four layers; C1, C2, C3 and F4 (output) layer. In C1 layer, there are 5 feature maps, 14 feature maps in C2 layer and 60 feature maps in C3 layer. In the F4 layer, it has 40 feature maps due to classification of 40 subjects contained in the ORL database. The design has a reduced size of feature maps as it does not require padding in the convolution process. It fused convolution and subsampling together, hence reduces the number of layers required. The 40 number of subjects from ORL database is trained using the mentioned CNN architecture.

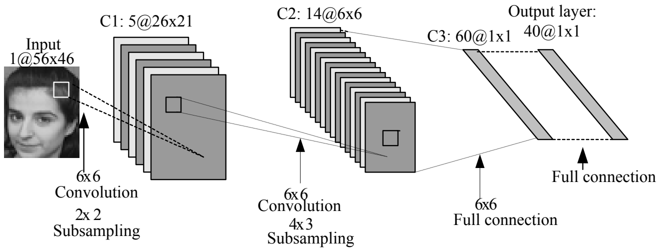


**Figure 3:** The proposed CNN architecture for face recognition

The result obtained after training is optimal weights that are interconnected at each layer. These optimal weights from layer C1 to C3 act like a feature extractor and is kept fixed when new subject(s) from JAFFE database is/are inserted into the system. Only the weights interconnected from layer C3 to output layer is randomly initialized according to the number of new subjects. Layer C3 becomes the new input to the two layers CNN. Figure 4 visualize the idea with 1 new subject inserted into the system. The detail information of this method can be referred in[20].

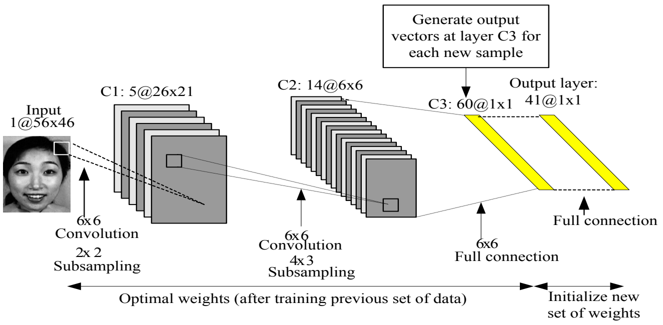


**Figure 4:** The weights in between layer C3 and output layer is randomly initialize according to the number of subjects

### Data preparation

The face image database used in this system is ORL and JAFFE databases. Each of the databases has 40 and 10 subjects respectively with 10 images for each subject. Those 10 images will be divided for training and test in the ratio of 8:2. The preprocessing step for ORL database involves image resizing to 56×46 only while for JAFFE database, cropping and resizing is needed. Figure 5 and Figure 6 depict examples of AT&T and JAFFE database respectively.

**Figure 5:** Samples from database

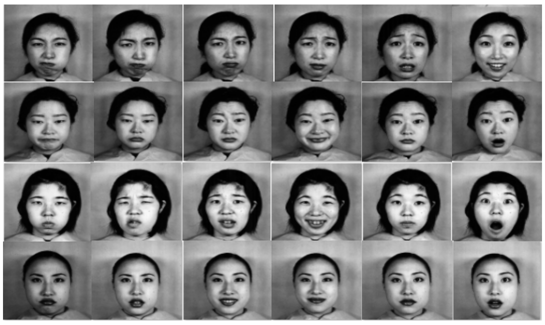
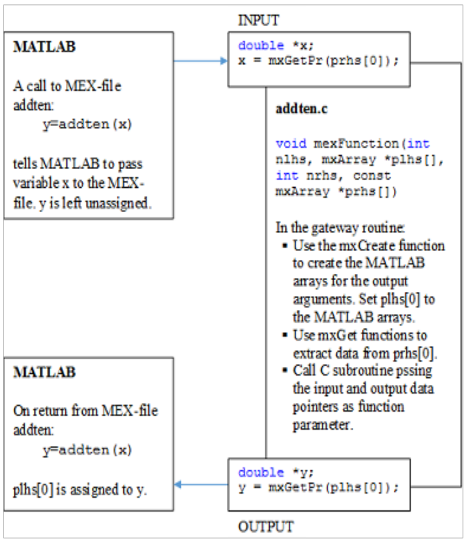


**Figure 6:** Samples from JAFFE database

e image database used in this system is ORL and JAFFE databases. Each of the atabases has 40 and 10 subjects respectively with 10 images for each subject. Those 10 images will be divided for training and test in the ratio of

### MEX-files

As mentioned before, the preprocessing stage is conducted in MATLAB Window-based environment while the CNN is developed in C language using LINUX based environment. Initially, both parts have been developed separately. However, in order to create a complete system, integration of both parts is needed. The integration is carried out in MATLAB Window-based environment using MEX-files. MEX-files call the C language based algorithms in MATLAB R2014a. MEX-files are dynamically linked subroutines that the MATLAB interpreter loads and executes. Figure 7 shows how the data flow in and out of MATLAB using MEX-file including what functions the gateway routine perform. Figure 8 shows the example of MEX-file and how it works.



A call to MEX-file addten:
y=addten(x)

tells MATLAB to pass variable x to the MEX-file. y is left unassigned.

```
double *x;
x = mxGetPr(prhs[0]);
```

```
void mexFunction(int
nlhs, mxArray *plhs[],
int nrhs, const
mxArray *prhs[])
```

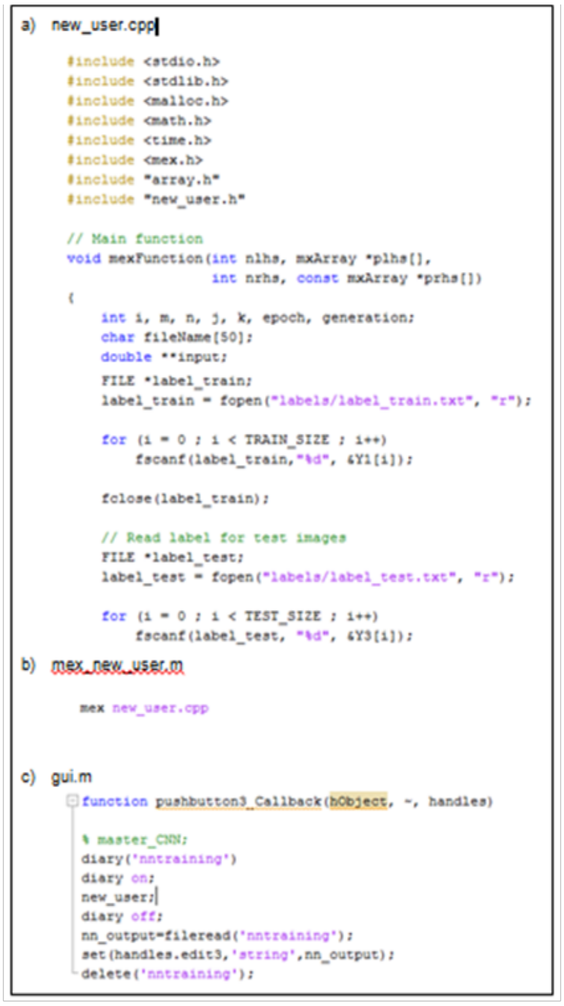
In the gateway routine:
- Use the mxCreate function to create the MATLAB arrays for the output arguments. Set plhs[0] to the MATLAB arrays.
- Use mxGet functions to extract data from prhs[0].
- Call C subroutine pssing the input and output data pointers as function parameter.

On return from MEX-file addten:
y=addten(x)

plhs[0] is assigned to y.

```
double *y;
y = mxGetPr(plhs[0]);
```

**Figure 7:** C MEX Cycle



a) new_user.cpp

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <math.h>
#include <time.h>
#include <mex.h>
#include "array.h"
#include "new_user.h"

// Main function
void mexFunction(int nlhs, mxArray *plhs[],
                 int nrhs, const mxArray *prhs[])
{
    int i, m, n, j, k, epoch, generation;
    char fileName[50];
    double **input;
    FILE *label_train;
    label_train = fopen("labels/label_train.txt", "r");

    for (i = 0 ; i < TRAIN_SIZE ; i++)
        fscanf(label_train,"%d", &Y1[i]);

    fclose(label_train);

    // Read label for test images
    FILE *label_test;
    label_test = fopen("labels/label_test.txt", "r");

    for (i = 0 ; i < TEST_SIZE ; i++)
        fscanf(label_test, "%d", &Y3[i]);
```

b) mex_new_user.m

```
 mex new_user.cpp
```

c) gui.m

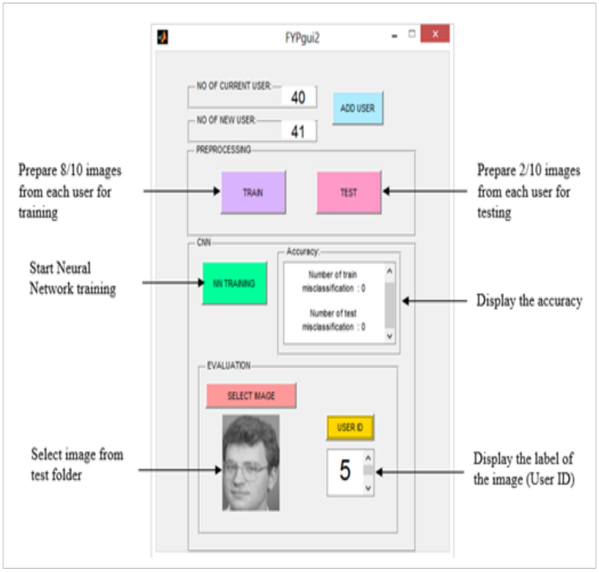
```
function pushbutton3_Callback(hObject, ~, handles)

% master_CNN;
diary('nntraining')
diary on;
new_user;
diary off;
nn_output=fileread('nntraining');
set(handles.edit3,'string',nn_output);
delete('nntraining');
```

**Figure 8:** (a) The new_user.cpp (b) The mex_new_user.m (c) gui.m

This tool enables the system to run continuously without requiring the user to manually open each file to run all the steps. The generated files will be automatically saved in a folder and are accessible whenever the system needs them.

## Experimental Results and Analysis

The proposed system run on a 2.4GHz Intel i3 3110M quad core processor, 8GB RAM, with Windows 8 Pro. A general user interface (GUI) is developed in MATLAB for the system to interact with users as shown in Figure 9. The GUI consists of 4 components. The first part is to add new users into the system. The second part is for the image preprocessing step. During this process, the images of each user will be resized to 56×46. Those images are then divided into the ratio of 8:2 for training and test samples respectively. The third part is for CNN training. In this step, the training will stop when new accuracy is gained. After the new trained weights are saved, evaluation process will take place. In the evaluation part, user can browse to select an image to be tested. Then, the user ID of the image will be displayed after the evaluation button is pressed. The user ID should be identical with the ID label attached for each subject.



NO OF CURRENT USER: 40

NO OF NEW USER: 41

Number of train misclassification : 0

Number of test misclassification : 0

5

Prepare 8/10 images from each user for training

Prepare 2/10 images from each user for testing

Start Neural Network training

Display the accuracy

Select image from test folder

Display the label of the image (User ID)

**Figure 9:** The system GUI

The system is tested with different number of new subject(s), to evaluate the number of epochs and the average training time. The result is shown in Table 1.

**Table 1:** Result obtained after inserting 1 to 10 new subjects into the system

| Number of new subject(s) | Total Number of Subjects | Number of Epochs | Average time to train (s) | Accuracy (%) |
|---|---|---|---|---|
| 1 | 41 | 2 | 8.795699 | 100 |
| 2 | 42 | 2 | 9.031064 | 100 |
| 3 | 43 | 2 | 9.623741 | 100 |
| 4 | 44 | 2 | 9.72347 | 100 |
| 5 | 45 | 4 | 10.64926 | 100 |
| 6 | 46 | 3 | 10.92308 | 100 |
| 7 | 47 | 2 | 48.74828 | 100 |
| 8 | 48 | 5 | 32.19402 | 100 |
| 9 | 49 | 4 | 16.75137 | 100 |
| 10 | 50 | 5 | 52.25557 | 100 |

From the table, it is shown that only a few epochs are required to train new number of subjects that is less than 6 epochs. The average training time for 41 to 46 subjects are consistently increasing. However, the average training time for 47 to 50 subjects is not as expected. In conclusion, the average training time increases as the network grow which means that the proposed design is suitable for less than 100 subjects.

The proposed CNN architecture is suitable for 'moderate' type of image challenge such as ORL and JAFFE. Moderate challenge means moderate degree of variation in poses (up to 20 degrees), lighting (dark homogenous background), facial expressions and head positions. If the images of type 'complex' challenges such as AR and FERET database, bigger CNN architecture is needed as discussed in[22].

The results showed that the system has 100% accuracy in recognizing the images, with number of epochs is less than 6. However, the system has inconsistent average time to train when the number of new subjects added is more than 7. The inconsistency happened due to the complexity of the data set, as the number of new subject is increasing. The accuracy of 100% is possible since AT&T database is a moderate challenge type of face database. The average time to train is also longer compared to the C language platform as reported in[21].

## Conclusion

An offline face recognition system with GUI is developed. The MEX function is successfully used to call the C algorithm into MATLAB environment. The system is developed based on four-layers CNN. However, only 2-layers of CNN are involved in retraining new incoming subjects into the system. This face recognition system has a faster retraining process, compared to retraining with 4-layers CNN. The epoch used for training in this system is less than 15 epochs. The accuracy of the system is 100% for all 50 subjects. However, MATLAB platform is not suitable for this system as there was result degradation where the average time to train was not consistent. For future work, this face recognition system could be developed in other platform; such as C, and could be extended as a real-time system. The system should have the ability to capture new images and saved into the image database.

**Acknowledgment:** This work is supported by Universiti Teknikal Malaysia Melaka (UTeM) under the grant PJP/2014/FKEKK (5B)/S01334

## References

1. Chandra, E., Kanagalakshmi, K. Cancelable biometric template generation and protection schemes: A review. (2011) Electro Comput Technol 5: 15-20.
2. Khalajzadeh, H., Mansouri, M., Teshnehlab, M. Hierarchical Structure Based Convolutional Neural Network for Face Recognition. (2013) Int J Comput Intell Applicat 12(03).
3. Li, H., Lin, Z., Shen, X., et al. A Convolutional Neural Network Cascade for Face Detection. (2015) Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
4. Shan Sung Liew, Mohamed, K.H., Syafeeza, A.R., et al. Gender classification: A convolutional neural network approach. (2014) Turkish J Elect Eng Comput Sci 1-17.
5. Ghosh, R., et al. Real-time object recognition and orientation estimation using an event-based camera and CNN. (2014) Biomedical Circuits and Systems Conference (BioCAS), IEEE Pg: 544-547.
6. Cheung, B. Convolutional Neural Networks Applied to Human Face Classification. (2012) 11th Int Conf Mach Learn Applicat (ICMLA) 2: 580-583.
7. Taigman, Y., Ming, Y., Marc, A.R., et al. Deepface: Closing the gap to human-level performance in face verification. (2014) Comput Vision Pattern Recognit (CVPR).
8. Kumar, N., Berg, A.C., Nayar, S.K., et al. Describable visual attributes for face verification and image search. (2011) IEEE Trans Pattern Anal Mach Intell 33(10): 1962-1977.
9. Cao, Z., Yin, Q., Jian, S., et al. Face recognition with learning-based descriptor. (2010) Computer Vision and Pattern Recognition (CVPR) 2707-2714.
10. Berg, T., Belhumeur, P.N. Tom-vs-Pete Classifiers and Identity-Preserving Alignment for Face Verification. (2012) BMVC 1-11.
11. Li, Z., Nagasaka, T., Kurihara, T., et al. A hybrid biometric system using touch-panel-based finger-vein identification and deformable-registration-based face identification. (2014) Systems, Man and Cybernetics (SMC) 69-74.
12. Ramaiah, N.P., Ijjina, E.P., Mohan, C.K. Illumination Invariant Face Recognition Using Convolutional Neural Networks. Signal Processing, Informatics, Communication and Energy Systems (SPICES) 19-21.
13. Valstar, M.F., Mehu, M., Pantik, M., et al. Meta-analysis of the first facial expression recognition challenge. (2012) IEEE Transactions Systems Man and Cybern B Cybern 42(4): 966-979.
14. Ahmad Radzi, S., Khalil, M.H., Liew, S.S., et al. Convolutional Neural Network for Face Recognition with Pose and Illumination Variation. (2014) Int J Eng Technol (IJET) 6(1): 44-57.
15. Parmar, D.N., Mehta, B.B. Face Recognition Methods & Applications. (2014) Int j comp tech applicant 4(1): 84-86.
16. Deboeverie, F., Veelaert, P., Philips, W. Best view selection with geometric feature based face recognition. (2012) Image Processing (ICIP) 1461-1464.
17. Salan, T., Iftekharuddin, K.M. Large pose invariant face recognition using feature-based recurrent neural network. (2012) Neural Networks (IJCNN) 1-7.
18. Lawrence, S., Giles, C.L., Back, A.D., et al. Face Recognition: A Convolutional Neural Network Approach. (1997) IEEE Trans Neural Netw 8(1): 98–113.
19. Khalajzadeh, H., Mansouri, M., Teshnehlab, M. Face Recognition Using Convolutional Neural Network and Simple Logistic Classifier. (2014) Soft Computing in Industrial Applications. Springer p: 197-207.
20. LeCun, Y., Leon, B., Patrick, H., et al. Gradient-Based Learning Applied to Document Recognition. (1998) Proceedings of the IEEE 1-46.
21. Syafeeza, A., Khalil, H.M., Saad, N.M., et al. An Improved Retraining Scheme for Convolutional Neural Network. (2015) J Telecom Elect Comput Eng (JTEC) 7(1): 5-9.
22. Syafeeza, A.R., Liew, S.S., Khalil, H.M., et al. Convolutional Neural Network for Face Recognition with Pose and Illumination Variation. (2014) Int J Eng Technol 6(1): 44-57.

Ommega Online Publisher
Bioinformatics, Proteomics and Immaging Analysis
Short Title : Bioinfo Proteom Img Anal

ISSN: 2381-0793
E-mail : bioinfo@ommegaonline.org
website: www.ommegaonline.org