JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# FACIAL EXPRESSION RECOGNITION SYSTEM USING CNN

**MAHESWARAN M, ANGEL SHALINI M**

Hindusthan College Of Arts and Science

## CHAPTER 1

**INTRODUCTION**:

A Facial expression is the visible manifestation of the affective state, cognitive activity, intention, personality and psychopathology of a person and plays a communicative role in interpersonal relations. Human facial expressions can be easily classified into 7 basic emotions:happy, sad, surprise, fear, anger, disgust, and neutral. Our facial emotions are expressed through activation of specific sets of facial muscles. These sometimes subtle, yet complex, signals in an expression often contain an abundant amount of information about our state of mind.

Automatic recognition of facial expressions can be an important component of natural human-machine interfaces; it may also be used in behavioral science and in clinical practice. Ithave been studied for a long period of time and obtaining the progress recent decades. Thoughmuch progress has been made, recognizing facial expression with a high accuracy remains to be difficult due to the complexity and varieties of facial expressions.

On a day to day basics humans commonly recognize emotions by characteristic features, displayed as a part of a facial expression. For instance happiness is undeniably associated with a smile or an upward movement of the corners of the lips. Similarly other emotions are characterized by other deformations typical to a particular expression. Research into automatic recognition of facial expressions addresses the problems surrounding the representation and categorization of static or dynamic characteristics of these deformations offace pigmentation.

In machine learning, a convolutional neural network (CNN, or ConvNet) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons isinspired by the organization of the animal visual cortex. Individual cortical neurons respond tostimuli in a restricted region of space known as the receptive field. The receptive fields of different neurons partially overlap such that they tile

the visual field. The response of an individual neuron to stimuli within its receptive field can be approximated mathematically by a convolution operation. Convolutional networks were inspired by biological processes and are variations of multilayer perceptron designed to use minimal amounts of preprocessing.

They have wide applications in image and video recognition, recommender systems and natural language processing. The convolutional neural network is also known as shift invariant or space invariant artificial neural network (SIANN), which is named based on its shared weights architecture and translation invariance characteristics.

### 1.1 Problem definition

Human emotions and intentions are expressed through facial expressions and derivingan efficient and effective feature is the fundamental component of facial expression system. Facial expressions convey non-verbal cues, which play an important role in interpersonal relations. Automatic recognition of facial expressions can be an important component of natural human-machine interfaces; it may also be used in behavioral science and in clinical practice. An automatic Facial Expression Recognition system needs to solve the following problems: detection and location of faces in a cluttered scene, facial feature extraction, and facial expression classification.

### 1.2 Scope of the Project

In this project facial expression recognition system is implemented using convolution neural network. Facial images are classified into seven facial expression categories namely Anger, Disgust, Fear, Happy, Sad, Surprise and 'Neutral. Kaggle dataset is used to train and test the classifier.

### 1.3 MODULE DESCRIPTION:

#### 1. Dataset

The dataset from a Kaggle Facial Expression Recognition Challenge (FER2013) is usedfor the training and testing. It comprises pre-cropped, 48-by-48-pixel grayscale images of faceseach labeled with one of the 7 emotion classes: anger, disgust, fear, happiness, sadness, surprise, and neutral. Dataset has training set of 35887 facial images with facial expression labels.. The dataset has class imbalance issue, since some classes have large number of examples while some has few.

#### 2. Facial Feature Extraction:

Facial Features extraction is an important step in face recognition and is defined as the process of locating specific regions, points, landmarks, or curves/contours in a given 2-D imageor a 3D range image. In this feature extraction step, a numerical feature vector is generated from the resulting registered image. Common features that can be extracted area.

a) Lips
b) Eyes
c) Eyebrows

d) Nose tip

**3. Emotion Classification:**

In the third step, CNN Classifier is then used to classify image taken from webcam in Laptop. Face is detected in webcam frames using Haar cascade classifier from MATLAB. Thendetected face is cropped and normalized and fed to CNN Classifier.

# CHAPTER 2

## SYSTEM SPECIFICATIONS:

### 2.1 HARDWARE SPECIFICATIONS:

1. Pentium IV Processor and above

2. 2 GB RAM and above

3. 40GB HDD and above

4. 1024 * 768 Resolution Color Monitor

### 2.2 SOFTWARE SPECIFICATIONS:

1. OS: Windows 7 and above

2. MATLAB

### 2.3 SOFTWARE CONFIGURATION:

**MATLAB**:

MATLAB® is a programming platform designed specifically for engineers and scientists to analyse and design systems and products that transform our world. The heart of MATLAB is the MATLAB language, a matrix-based language allowing the most natural expression of computational mathematics.

MATLAB® combines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly.It includes the Live Editor for creating scripts that combine code, output, and formatted text inan executable notebook.

**Origins**:

MATLAB was invented by mathematician and computer programmer Cleve Moler. The idea for MATLAB was based on his 1960s PhD thesis. Moler became a math professor at the University of New Mexico and started developing MATLAB for his students as a hobby. He developed MATLAB's initial linear algebra programming in 1967 withhis one-time thesis advisor, George Forsythe. This was followed by Fortran code for linear equations in 1971.

In the beginning (before version 1.0) MATLAB "was not a programming language; it was a simple interactive matrix calculator. There were no programs, no toolboxes, no graphics.And no ODEs or FFTs.

The first early version of MATLAB was completed in the late 1970s. The software wasdisclosed to the public for the first time in February 1979 at the Naval Postgraduate School inCalifornia. Early versions of MATLAB were simple matrix calculators with 71 pre-built functions. At the time, MATLAB was distributed for free to universities. Moler would leave copies at universities he visited and the software developed a strong following in the math departments of university campuses.

In the 1980s, Cleve Moler met John N. Little. They decided to reprogram MATLAB inC and market it for the IBM desktops that were replacing mainframe computers at the time. John Little and programmer Steve Bangert re-programmed MATLAB in C, created the MATLAB programming language, and developed features for toolboxes.

**Commercial development**:

MATLAB was first released as a commercial product in 1984 at the Automatic ControlConference in Las Vegas. MathWorks, Inc. was founded to develop the software and the MATLAB programming language was released. The first MATLAB sale was the following year, when Nick Trefethen from the Massachusetts Institute of Technology bought ten copies.

By the end of the 1980s, several hundred copies of MATLAB had been sold to universities for student use. The software was popularized largely thanks to toolboxes createdby experts in various fields for performing specialized mathematical tasks. Many of the toolboxes were developed as a result of Stanford students that used MATLAB in academia, then brought the software with them to the private sector.

Over time, MATLAB was re-written for early operating systems created by Digital Equipment Corporation, VAX, Sun Microsystems, and for Unix PCs. Version 3 was released in 1987. The first MATLAB compiler was developed by Stephen C. Johnson in the 1990s.

In 2000, MathWorks added a Fortran-based library for linear algebra in MATLAB 6, replacing the software's original LINPACK and EISPACK subroutines that were in
C. MATLAB's Parallel Computing Toolbox was released at the 2004 Supercomputing Conference and support for graphics processing units (GPUs) was added to it in 2010

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation
- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics

- Application development, including Graphical User Interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especiallythose with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects, which together represent the state-of-the-art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high- productivity research, development, and analysis.

MATLAB features a family of application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to *learn* and *apply* specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

The MATLAB system consists of five main parts:

**The MATLAB language.**

This is a high-level matrix/array language with control flow statements, functions, datastructures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

**The MATLAB working environment.**

This is the set of tools and facilities that you work with as the MATLAB user or programmer. It includes facilities for managing the variables in your workspace and importingand exporting data. It also includes tools for developing, managing, debugging, and profiling M-files, MATLAB's applications.

**Handle Graphics.**

This is the MATLAB graphics system. It includes high-level commands for two- dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands that allow you to fully customize the appearance of graphics as well as to build complete Graphical User Interfaces on your MATLAB applications.

**The MATLAB mathematical function library.**

This is a vast collection of computational algorithms ranging from elementary functionslike sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

**The MATLAB Application Program Interface (API).**

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It include facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

# CHAPTER 3

## SYSTEM STUDY:

### 3.1 EXISTING SYSTEM:

In the existing system, we have used appearance based methods for extracting featuresfrom a region of interest (ROI). The ROI's, were detected using the Viola-Jones face detectionmethod. The appearance based features, are extracted by applying a predefined bank of Gaborfilters with different orientation and scales. A pre-processing step normalizes the window of region of interest to the same size and proportions. This process results a feature vector of 40690 values, which we split to 10 different sub feature vectors per each image. The collectionsof the sub feature vectors extracted for each image, were used to train a Radial Bases Function Network resulting 10 RBFN's. The output of these RBFN's, is used as an input to an additionallayer of ANN in order to define the final classification result. The last Layer weights were trained using the gradient descent method and back propagated to the previous layer for schematic flow of the process.

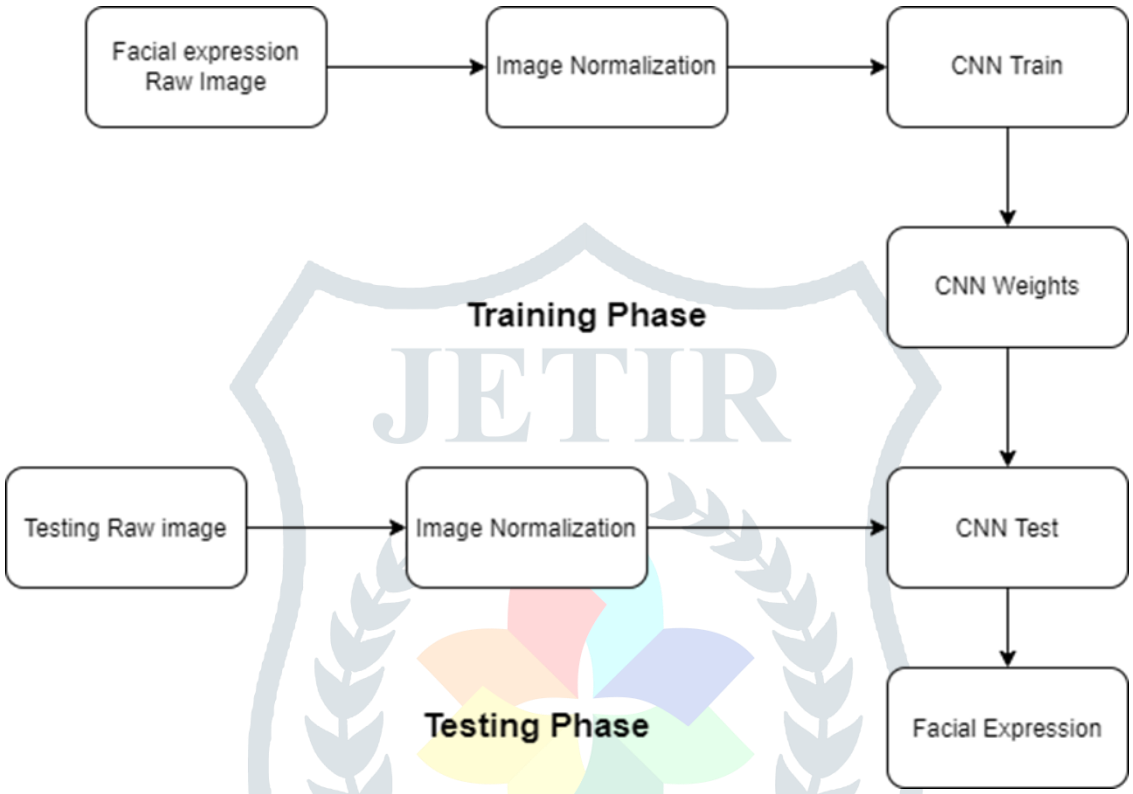
### 3.2 PROPOSED SYSTEM:

Facial Expression Recognition usually performed in four-stages consisting of pre- processing, face detection, feature extraction, and expression classification. In this project we applied various deep learning methods (convolutional neural networks) to identify the key seven human emotions: anger, disgust, fear, happiness, sadness, surprise and neutrality. Facialexpression recognition system is implemented using Convolution Neural Network (CNN). CNN model of the project is based on LeNet Architecture. Kaggle facial expression dataset with seven facial expression labels as happy, sad, surprise, fear, anger, disgust, and neutral is used in this project.

# CHAPTER 4

## SYSTEM DESIGN

### 4.1 SYSTEM FLOW DIAGRAM:




### 4.2 INPUT DESIGN:

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Welldesigned input forms and screens have following properties −

- It should serve specific purpose effectively such as storing, recording, and retrievingthe information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.
- All these objectives are obtained using the knowledge of basic design principlesregarding −
    - What are the inputs needed for the system?

- How end users respond to different elements of forms and screens.

**Objectives for Input Design**

The objectives of input design are −

- To design data entry and input procedures
- To reduce input volume
- To design source documents for data capture or devise other data capture methods
- To design input data records, data entry screens, user interface screens, etc.
- To use validation checks and develop effective input controls.

**4.3 OUTPUT DESIGN:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls andprototype report layouts.

**Objectives of Output Design**

The objectives of input design are −

- To develop output design that serves the intended purpose and eliminates theproduction of unwanted output.
- To develop the output design that meets the end users requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.
- To make the output available on time for making good decisions.

# CHAPTER 5

## SYSTEM TESTING

Testing is a process of executing a program with the indent of finding an error. Testingis a crucial element of software quality assurance and presents ultimate review of specification,design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error.

A successful test is one that uncovers an as undiscovered error.

**5.1 TESTING OBJECTIVES:**

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3. A successful test is one that uncovers an undiscovered error

**5.2 TESTING PRINCIPLES:**

- All tests should be traceable to end user requirements
- Tests should be planned long before testing begins
- Testing should begin on a small scale and progress towards testing in large
- Exhaustive testing is not possible
- To be most effective testing should be conducted by a independent third party

The primary objective for test case design is to derive a set of tests that has the highestlivelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

- **White box testing.**
- **Black box testing.**

**White-box testing:**

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

**Block-box testing:**

Black box testing is designed to validate functional requirements without regard to theinternal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

**Testing strategies:**

A strategy for software testing must accommodate low-level tests that are necessary toverify that all small source code segment has been correctly implemented as well as high-leveltests that validate major system functions against customer requirements.

## 5.3 TESTING FUNDAMENTALS:

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects,it can only show that software defects present.

## 5.4 TESTING INFORMATION FLOW:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

**Unit testing:**

Unit testing is essential for the verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed designdescription as a guide, important paths are tested to uncover errors with in the boundary of themodules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

**Integration testing:**

Integration testing focuses on unit tested modules and build the program structure thatis dictated by the design phase.

**System testing**:

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and it's original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire systemis working properly or not will be tested here, and specified path ODBC connection will corrector not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

**Acceptance Testing:**

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end userwith the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implementedsystem satisfies its requirements.

Tools to special importance during acceptance testing include:

Test coverage Analyzer – records the control paths followed for each test case.

Timing Analyzer – also called a profiler, reports the time spent in various regions of the codeare areas to concentrate on to improve system performance.

Coding standards – static analyzers and standard checkers are used to inspect code fordeviations from standards and guidelines.

### 5.5 TEST CASES:

Test cases are derived to ensure that all statements in the program have been executedat least once during testing and that all logical conditions have been executed.

Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structure to assure their validity.

The test case specification for system testing has to be submitted for review before systemtesting commences.

## CHAPTER 6

## IMPLEMENTATION

### 6.1 ARCHITECTURE OF CNN

A typical architecture of a convolutional neural network contains an input layer, someconvolutional layers, some fully-connected layers, and an output layer. CNN is designed withsome modification on LeNet Architecture. It has 6 layers without considering input and output.The architecture of the Convolution Neural Network used in the project is shown in the following figure.

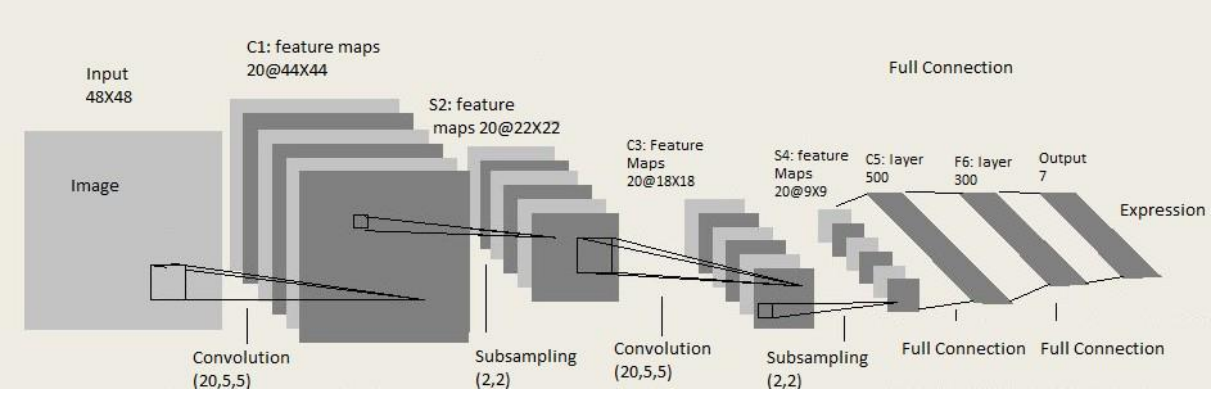



Fig: Architecture of CNN ( Modified from LeNet Architecture)

### 6.2 INPUT LAYER:

The input layer has pre-determined, fixed dimensions, so the image must be pre- processed before it can be fed into the layer. Normalized gray scale images of size 48 X 48 pixels from Kaggle dataset are used for training, validation and testing. For testing propose laptop webcam images are also used, in which face is detected and cropped using OpenCV Haar Cascade Classifier and normalized.

### 6.3 CONVOLUTION AND POOLING (CONVPOOL) LAYERS:

Convolution and pooling is done based on batch processing. Each batch has N images and CNN filter weights are updated on those batches. Each convolution layer takes image batchinput of four dimension N x Color-Channel x width x height. Feature map or filter for convolution are also four dimensional (Number of feature maps in, number of feature maps out, filter width, filter height). In each convolution layer, four dimensional convolution is calculated between image batch and feature maps. After convolution only parameter that change is image width and height.

New image width = old image width – filter width + 1 New image

height = old image height – filter height + 1

After each convolution layer downsampling / subsampling is done for dimensionality reduction. This process is called Pooling. Max pooling and Average Pooling are two famous pooling method. In this project max pooling is done after convolution. Pool size of (2x2) is taken, which splits the image into grid of blocks each of size 2x2 and takes maximum of 4 pixels. After pooling only height and width are affected.

Two convolution layer and pooling layer are used in the architecture. At first convolution layer size of input image batch is Nx1x48x48. Here, size of image batch is N, number of color channel is 1 and both image height and width are 48 pixel. Convolution with feature map of 1x20x5x5 results image batch is of size Nx20x44x44. After convolution poolingis done with pool size of 2x2, which results image batch of size Nx20x22x22. This is followedby second convolution layer with feature map of 20x20x5x5, which results image batch of sizeNx20x18x18. This is followed by pooling layer with pool size 2x2, which results image batchof size Nx20x9x9.

### 6.4 FULLY CONNECTED LAYER

This layer is inspired by the way neurons transmit signals through the brain. It takes a large number of input features and transform features through layers connected with trainableweights. Two hidden layers of size 500 and 300 unit are used in fully-connected layer. The weights of these layers are trained by forward propagation of training data then backward propagation of its errors. Back propagation starts from evaluating the difference between prediction and true value, and back calculates the weight adjustment needed to every layer before. We can control the training speed and the complexity of the architecture by tuning thehyper-parameters, such as learning rate and network density. Hyper-parameters for this layer include learning rate, momentum, regularization parameter, and decay.

The output from the second pooling layer is of size Nx20x9x9 and input of first hiddenlayer of fully-

connected layer is of size Nx500. So, output of pooling layer is flattened to Nx1620 size and fed to first hidden layer. Output from first hidden layer is fed to second hiddenlayer. Second hidden layer is of size Nx300 and its output is fed to output layer of size equal to number of facial expression classes.

### 6.5 OUTPUT LAYER

Output from the second hidden layer is connected to output layer having seven distinctclasses. Using Softmax activation function, output is obtained using the probabilities for eachof the seven class. The class with the highest probability is the predicted class.

## CONCLUSION

The facial expression recognition system presented in this research work contributes aresilient face recognition model based on the mapping of behavioral characteristics with the physiological biometric characteristics. The physiological characteristics of the human face with relevance to various expressions such as happiness, sadness, fear, anger, surprise and disgust are associated with geometrical structures which restored as base matching template for the recognition system. Facial expression recognition system is implemented using Convolution Neural Network (CNN). CNN model of the project is based on LeNet Architecture. In this project, a LeNet architecture based six layer convolution neural network is implemented to classify human facial expressions i.e. happy, sad, surprise, fear, anger, disgust, and neutral.

## FUTURE ENHANCEMENT

In the future work, the model can be extended to color images. This will allow to investigate the efficacy of pre-trained models such as AlexNet or VGGNet for facial emotion recognition.

## BIBILIOGRAPHY

**References**:

[1] Shan, C., Gong, S., & McOwan, P. W. (2005, September). Robust facial expression recognition using local binary patterns. In *Image Processing, 2005. ICIP 2005.* IEEE International Conference on (Vol. 2, pp. II-370). IEEE.

[2] Chibelushi, C. C., & Bourel, F. (2003). Facial expression recognition: A brief tutorial overview. CVonline: On-Line Compendium of Computer Vision, 9.

[3] "Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation". DeepLearning 0.1. LISA Lab. Retrieved 31 August 2013.

[4] Matusugu, Masakazu; Katsuhiko Mori; Yusuke Mitari; Yuji Kaneda (2003). "Subject independent facial expression recognition with robust face detection using a convolutional neural network" (PDF). Neural Networks. 16 (5): 555–559. doi:10.1016/S0893- 6080(03)00115-1. Retrieved 17 November 2013.

[5] LeCun, Yann. "LeNet-5, convolutional neural networks". Retrieved 16 November 2013

[6] C. Zor, “Facial expression recognition,” Master’s thesis, University of Surrey, Guildford, 2008.