



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 13, April 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

9940 572 462

6381 907 438

ijirset@gmail.com

www.ijirset.com

Face Emotion Recognition Using CNN Classifier

Mrs.Indhumathi.N¹, Juwairiya.MF², Rashidha Begam.I³, Saleeha.S⁴

Assistant Professor , Department of Computer Science and Engineering , Dhaanish Ahmed Institute of Technology ,
Veerappanur , K.G Chavadi , Coimbatore, Tamil Nādu, India¹

U.G Student, Department of Computer Science and Engineering, Dhannish Ahmed Institute of Technology Veerappanur ,
K.G Chavadi , Ciombatore, Tamil Nādu, India^{2,3,4}

ABSTRACT: Speech emotion recognition from speech signal based on features analysis and NN-classifier. Automatic Face emotion recognition (SER) plays an important role in HCI systems for measuring people's emotions and has dominated psychology by linking expressions to group of basic emotions (i.e., anger, disgust, fear, happiness, sadness, and surprise). The recognition system involves Face emotion detection, features extraction and selection and finally classification. These features are useful to distinguish the maximum number of samples accurately and the NN classifier based on discriminate analysis is used to classify the six different expressions. The simulated results will be shown that the filter based feature extraction with used classifier gives much better accuracy with lesser algorithmic complexity than other Face emotion expression recognition approaches. Recognition is based on the stored image data of the different group of persons. Input images are of any type can be used for recognition,

1. Still images.
2. Video frames or video stills.
3. Video

OBJECTIVE:The objective of this project is to develop Automatic Facial Expression Recognition System which can take human facial images containing some expression as input and recognize and classify it into seven different expression classes such as happy, sad, anger, disgusted

I. INTRODUCTION

With the advent of modern technology our desires went high and it binds no bounds. In the present era a huge research work is going on in the field of digital image and image processing. The way of progression has been exponential and it is ever increasing. Image Processing is a vast area of research in the present day world and its applications are very widespread. Image processing is the field of signal processing where both the input and output signals are images. One of the most important applications of Image processing is Facial expression recognition. Our emotion is revealed by the expressions in our face. Facial Expressions plays an important role in interpersonal communication. Facial expression is a non verbal scientific gesture which gets expressed in our face as per our emotions. Automatic recognition of facial expression plays an important role in artificial intelligence and robotics and thus it is a need of the generation. Some application related to this includes Personal identification and Access control, Videophone and Teleconferencing, Forensic application, Human-Computer Interaction, Automated Surveillance, Cosmetology and so on.

Scope of the project:

The main scope of the project is to detect the human facial expression as image and then image will do the process the filtering which gave human face emotions and then goes on to discrete Wavelet transform is used to compress the image and train the image in neural network .image quality plays an important role in this process. In this we can capture the face image as a real time, what human can express in face. Face emotion detection is used to predict the emotion state of the person based on their face expressions.

Digital image processing

The identification of objects in an image would probably start with image processing techniques such as noise removal, followed by (low-level) feature extraction to locate lines, regions and possibly areas with certain textures.

The clever bit is to interpret collections of these shapes as single objects, e.g. cars on a road, boxes on a conveyor belt or cancerous cells on a microscope slide. One reason this is an AI problem is that an object can appear very different when viewed from different angles or under different lighting. Another problem is deciding what features belong to what object and which are background or shadows etc. The human visual system performs these tasks mostly unconsciously but a computer requires skillful programming and lots of processing power to approach human performance. Manipulating data in the form of an image through several possible techniques. An image is usually interpreted as a two-dimensional array of brightness values, and is most familiarly represented by such patterns as those of a photographic print, slide, television screen, or movie screen. An image can be processed optically or digitally with a computer.

To digitally process an image, it is first necessary to reduce the image to a series of numbers that can be manipulated by the computer. Each number representing the brightness value of the image at a particular location is called a picture element, or pixel. A typical digitized image may have 512×512 or roughly 250,000 pixels, although much larger images are becoming common. Once the image has been digitized, there are three basic operations that can be performed on it in the computer. For a point operation, a pixel value in the output image depends on a single pixel value in the input image. For local operations, several neighbouring pixels in the input image determine the value of an output image pixel. In a global operation, all of the input image pixels contribute to an output image pixel value.

These operations, taken singly or in combination, are the means by which the image is enhanced, restored, or compressed. An image is enhanced when it is modified so that the information it contains is more clearly evident, but enhancement can also include making the image more visually appealing.

An example is noise smoothing. To smooth a noisy image, median filtering can be applied with a 3×3 pixel window. This means that the value of every pixel in the noisy image is recorded, along with the values of its nearest eight neighbours. These nine numbers are then ordered according to size, and the median is selected as the value for the pixel in the new image. As the 3×3 window is moved one pixel at a time across the noisy image, the filtered image is formed.

Another example of enhancement is contrast manipulation, where each pixel's value in the new image depends solely on that pixel's value in the old image; in other words, this is a point operation. Contrast manipulation is commonly performed by adjusting the brightness and contrast controls on a television set, or by controlling the exposure and development time in printmaking. Another point operation is that of pseudo colouring a black-and-white image, by assigning arbitrary colours to the gray levels. This technique is popular in thermograph (the imaging of heat), where hotter objects (with high pixel values) are assigned one color (for example, red), and cool objects (with low pixel values) are assigned another color (for example, blue), with other colours assigned to intermediate values.

Recognizing object classes in real-world images is a long standing goal in Computer vision. Conceptually, this is challenging due to large appearance variations of object instances belonging to the same class. Additionally, distortions from background clutter, scale, and viewpoint variations can render appearances of even the same object instance to be vastly different. Further challenges arise from interclass similarity in which instances from different classes can appear very similar. Consequently, models for object classes must be flexible enough to accommodate class variability, yet discriminative enough to sieve out true object instances in cluttered images. These seemingly paradoxical requirements of an object class model make recognition difficult. This paper addresses two goals of recognition are image classification and object detection. The task of image classification is to determine if an object class is present in an image, while object detection localizes all instances of that class from an image. Toward these goals, the main contribution in this paper is an approach for object class recognition that employs edge information only. The novelty of our approach is that we represent contours by very simple and generic shape primitives of line segments and ellipses, coupled with a flexible method to learn discriminative primitive combinations. These primitives are complementary in nature, where line segment models straight contour and ellipse models

curved contour. We choose an ellipse as it is one of the simplest circular shapes, yet is sufficiently flexible to model curved shapes. These shape primitives possess several attractive properties. First, unlike edge-based descriptors they support abstract and perceptually meaningful reasoning like parallelism and adjacency. Also, unlike contour fragment features, storage demands by these primitives are independent of object size and are efficiently represented with four parameters for a line and five parameters for an ellipse.

Additionally, matching between primitives can be efficiently computed (e.g., with geometric properties), unlike contour fragments, which require comparisons between individual edge pixels. Finally, as geometric properties are easily scale normalized, they simplify matching across scales. In contrast, contour fragments are not scale invariant, and one is forced either to rescale fragments, which introduces aliasing effects (e.g., when edge pixels are pulled apart), or to resize an image before extracting fragments, which degrades image resolution.

In recent studies it is shown that the generic nature of line segments and ellipses affords them an innate ability to represent complex shapes and structures. While individually less distinctive, by combining a number of these primitives, we empower a combination to be sufficiently discriminative. Here, each combination is a two-layer abstraction of primitives: pairs of primitives (termed shape tokens) at the first layer, and a learned number of shape tokens at the second layer. We do not constrain a combination to have a fixed number of shape-tokens, but allow it to automatically and flexibly adapt to an object class. This number influences a combination's ability to represent shapes, where simple shapes favor fewer shape-tokens than complex ones. Consequently, discriminative combinations of varying complexity can be exploited to represent an object class. We learn this combination by exploiting distinguishing shape, geometric, and structural constraints of an object class. Shape constraints describe the visual aspect of shape tokens, while geometric constraints describe its spatial layout (configurations). Structural constraints enforce possible poses/structures of an object by the relationships (e.g., XOR relationship) between shape-tokens.

CLASSIFICATION OF IMAGES:

There are 3 types of images used in Digital Image Processing. They are

1. Binary Image
2. Gray Scale Image
3. Colour Image

BINARY IMAGE:

A binary image is a digital image that has only two possible values for each pixel. Typically the two colours used for a binary image are black and white though any two colours can be used. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color.

Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1). This name black and white, monochrome or monochromatic are often used for this concept, but may also designate any images that have only one sample per pixel, such as grayscale images

Binary images often arise in digital image processing as masks or as the result of certain operations such as segmentation, thresholding, and dithering. Some input/output devices, such as laser printers, fax machines, and bi-level computer displays, can only handle bi-level images

GRAY SCALE IMAGE

A grayscale Image is digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray (0-255), varying from black (0) at the weakest intensity to white (255) at the strongest.



Organized by

Dhaanish Ahmed Institute of Technology, KG chavadi, Coimbatore, India

Grayscale images are distinct from one-bit black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white (also called bi-level or binary images). Grayscale images have many shades of gray in between. Grayscale images are also called monochromatic, denoting the absence of any chromatic variation.

Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic proper when only a given frequency is captured. But also they can be synthesized from a full color image; see the section about converting to grayscale.

COLOUR IMAGE:

A (digital) color image is a digital image that includes color information for each pixel. Each pixel has a particular value which determines it’s appearing color. This value is qualified by three numbers giving the decomposition of the color in the three primary colours Red, Green and Blue. Any color visible to human eye can be represented this way. The decomposition of a color in the three primary colours is quantified by a number between 0 and 255. For example, white will be coded as R = 255, G = 255, B = 255; black will be known as (R,G,B) = (0,0,0); and say, bright pink will be : (255,0,255).

In other words, an image is an enormous two-dimensional array of color values, pixels, each of them coded on 3 bytes, representing the three primary colours. This allows the image to contain a total of $256 \times 256 \times 256 = 16.8$ million different colours. This technique is also known as RGB encoding, and is specifically adapted to human vision

CLASSIFICATION OF IMAGES:

There are 3 types of images used in Digital Image Processing. They are

1. Binary Image
2. Gray Scale Image
3. Colour Image

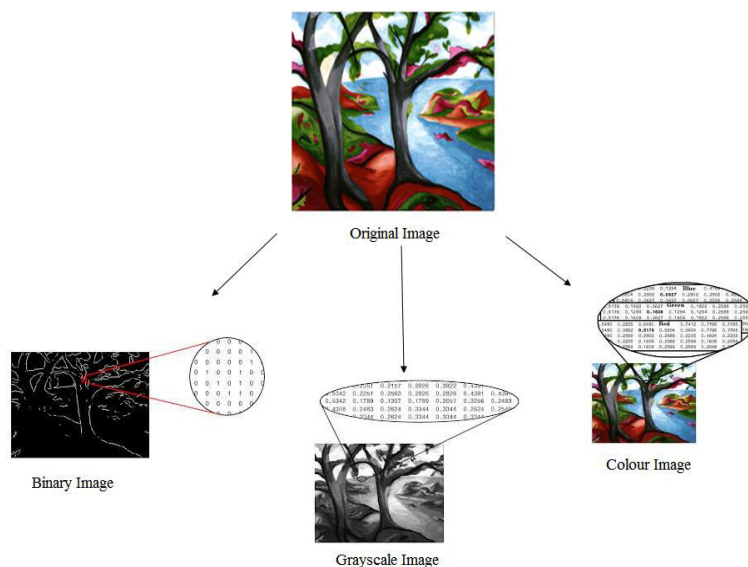


Figure 1.1 Representation of a image in binary, grayscale and colour form.

Binary image:

A binary image is a digital image that has only two possible values for each pixel. Typically the two colours used for a binary image are black and white though any two colours can be used. The colour used for the objects in the image is the foreground colour while the rest of the image is the background colour.

Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1) [Refer figure 1.1]. This name black and white, monochrome or monochromatic are often used for this concept, but also designate any images that have only one sample per pixel, such as grayscale images.

Binary images often arise in digital image processing as masks or as the result of certain operations such as segmentation, thresholding and dithering. Some input/output devices, such as laser printers, fax machines, and bi-level computer displays, can only handle bi-level images.

Gray scale image:

A grayscale Image is digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray (0-255), varying from black (0) at the weakest intensity to white (255) at the strongest [Refer figure 1.1].

Grayscale images are distinct from one-bit black-and-white images, which in the context of computer imaging are images with only the two colours, black, and white (also called bi-level or binary images). Grayscale images have many shades of gray in between. Grayscale images are also called monochromatic, denoting the absence of any chromatic variation.

Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic in nature when only a given frequency is captured.

Colour image:

A colour image is a digital image that includes colour information for each pixel. Each pixel has a particular value which determines its appearing colour. This value is qualified by three numbers giving the decomposition of the colour in the three primary colours Red, Green and Blue. Any colour visible to human eye can be represented this way. The decomposition of a colour in the three primary colours is quantified by a number between 0 and 255. For example, white will be coded as $R = 255, G = 255, B = 255$; black will be known as $(R,G,B) = (0,0,0)$ [Refer figure 1.1]. In other words, an image is an enormous two-dimensional array of colour values, pixels, each of them coded on 3 bytes, representing the three primary colours. This allows the image to contain a total of $256 \times 256 \times 256 = 16.8$ million different colours. This technique is also known as RGB encoding, and is specifically adapted to human vision.

II. APPLICATIONS

Digital camera images

Digital cameras generally include dedicated digital image processing chips to convert the raw data from the image sensor into a colour-corrected image in a standard image file format. Images from digital cameras often receive further processing to improve their quality, a distinct advantage that digital cameras have over film cameras. The digital image processing typically is executed by special software programs that can manipulate the images in many ways. Many digital cameras also enable viewing of histograms of images, as an aid for the photographer to understand the rendered brightness range of each shot more readily.

Intelligent transportation systems

Digital image processing has wide applications in intelligent transportation systems, such as automatic number plate recognition and traffic sign recognition.

III. SYSTEM ANALYSIS

Existing system

- Principal Component Analysis
- Geometric methods
- Support vector machine.

Drawbacks

- Low discriminatory power and high computational load
- In geometric based methods, the geometric features like distance between speech signals.

Proposed method

- Face Emotion recognition for transform features system through textural analysis and NN classifier. The system involves like anger, happy, sad disgust and anger.

Advantage:

- It will easily detect the face and recognise the human expression.
- In this image quality is essential for recognising the face.

IV. BLOCK DIAGRAM

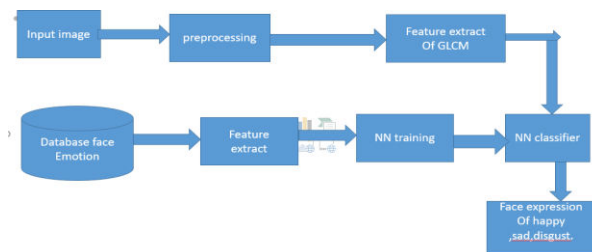


Fig. BlockDiagram

Modules description

- Input video
- Pre-processing
- Discrete wavelet transform
- Gcm feature extraction
- Database face emotion signals
- NN CLASSIFIER

V. PREPROCESSING

Image Pre-processing is a common name for operations with images at the lowest level of abstraction. Its input and output are intensity images. The aim of pre-processing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing.



Image restoration is the operation of taking a corrupted/noisy image and estimating the clean original image. Corruption may come in many forms such as motion blur, noise, and camera misfocus. Image restoration is different from image enhancement in that the latter is designed to emphasize features of the image that make the image more pleasing to the observer, but not necessarily to produce realistic data from a scientific point of view. Image enhancement techniques (like contrast stretching or de-blurring by a nearest neighbour procedure) provided by "Imaging packages" use no a priori model of the process that created the image. With image enhancement noise can be effectively be removed by sacrificing some resolution, but this is not acceptable in many applications. In a Fluorescence Microscope resolution in the z-direction is bad as it is. More advanced image processing techniques must be applied to recover the object. De-Convolution is an example of image restoration method. It is capable of: Increasing resolution, especially in the axial direction removing noise increasing contrast.

GLCM FEATUES

To create a GLCM, use the graycomatrix function. The graycomatrix function creates a gray-level co-occurrence matrix (GLCM) by calculating how often a pixel with the intensity (gray-level) value i occurs in a specific spatial relationship to a pixel with the value j . By default, the spatial relationship is defined as the pixel of interest and the pixel to its immediate right (horizontally adjacent), but you can specify other spatial relationships between the two pixels. Each element (i,j) in the resultant GLCM is simply the sum of the number of times that the pixel with value i occurred in the specified spatial relationship to a pixel with value j in the input image. Because the processing required to calculate a GLCM for the full dynamic range of an image is prohibitive, graycomatrix scales the input image. By default, graycomatrix uses scaling to reduce the number of intensity values in gray scale image from 256 to eight. The number of gray levels determines the size of the GLCM. To control the number of gray levels in the GLCM and the scaling of intensity values, using the Num Levels and the Gray Limits parameters of the gray comatrix function. See the graycomatrix reference page for more information.

The gray-level co-occurrence matrix can reveal certain properties about the spatial distribution of the gray levels in the texture image. For example, if most of the entries in the GLCM are concentrated along the diagonal, the texture is coarse with respect to the specified offset. To illustrate, the following figure shows how graycomatrix calculates the first three values in a GLCM. In the output GLCM, element $(1,1)$ contains the value 1 because there is only one instance in the input image where two horizontally adjacent pixels have the values 1 and 1, respectively.

GLCM $(1,2)$ contains the value 2 because there are two instances where two horizontally adjacent pixels have the values 1 and 2. Element $(1,3)$ in the GLCM has the value 0 because there are no instances of two horizontally adjacent pixels with the values 1 and 3. graycomatrix continues processing the input image, scanning the image for other pixel pairs (i,j) and recording the sums in the corresponding elements of the GLCM.

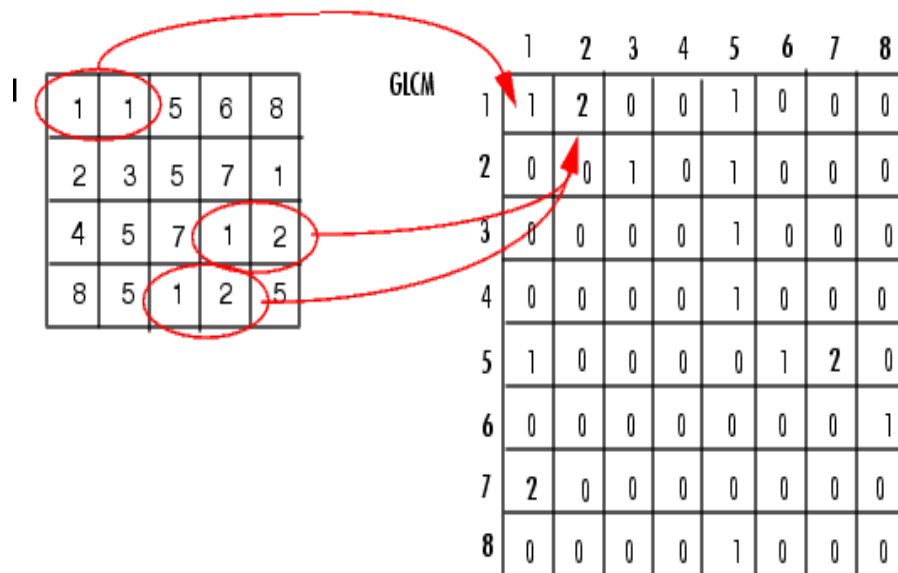


Fig. elements of the GLCM

To create multiple GLCMs, specify an array of offsets to the graycomatrix function. These offsets define pixel relationships of varying direction and distance. For example, you can define an array of offsets that specify four directions (horizontal, vertical, and two diagonals) and four distances. In this case, the input image is represented by 16 GLCMs. When you calculate statistics from these GLCMs, you can take the average.

You specify these offsets as a p-by-2 array of integers. Each row in the array is a two-element vector, [row_offset, col_offset], that specifies one offset. Row_offset is the number of rows between the pixel of interest and its neighbour. Col_offset is the number of columns between the pixel of interest and its neighbour. This example creates an offset that specifies four directions and 4 distances for each direction. After you create the GLCMs, you can derive several statistics from them using the graycoprops function. These statistics provide information about the texture of an image. Statistic such as a as Contras, Correlation, Energy, Homogeneity gives information about image.

VI. FEATURE EXTRACTION

Texture analysis

Texture is that innate property of all surfaces that describes visual patterns, each having properties of homogeneity. It contains important information about the structural arrangement of the surface, such as; clouds, leaves, bricks, fabric, etc. It also describes the relationship of the surface to the surrounding environment. In short, it is a feature that describes the distinctive physical composition of a surface.

Texture properties include:

- Coarseness
- Contrast
- Directionality
- Line-likeness
- Regularity
- Roughness



Texture is one of the most important defining features of an image. It is characterized by the spatial distribution of gray levels in a neighborhood [8]. In order to capture the spatial dependence of gray-level values, which contribute to the perception of texture, a two-dimensional dependence texture analysis matrix is taken into consideration. This two-dimensional matrix is obtained by decoding the image file; jpeg, bmp, etc.

Methods of Representation

There are three principal approaches used to describe texture; statistical, structural and spectral...

- Statistical techniques characterize textures using the statistical properties of the grey levels of the points/pixels comprising a surface image. Typically, these properties are computed using: the grey level co-occurrence matrix of the surface, or the wavelet transformation of the surface.
- Structural techniques characterize textures as being composed of simple primitive structures called “texels” (or texture elements). These are arranged regularly on a surface according to some surface arrangement rules.
- Spectral techniques are based on properties of the Fourier spectrum and describe global periodicity of the grey levels of a surface by identifying high-energy peaks in the Fourier spectrum [9].

For optimum classification purposes, what concern us are the statistical techniques of characterization... This is because it is these techniques that result in computing texture properties... The most popular statistical representations of texture are:

Co-occurrence Matrix

Tamura Texture

Wavelet Transform

Co-occurrence Matrix

Originally proposed by R.M. Haralick, the co-occurrence matrix representation of texture features explores the grey level spatial dependence of texture [2]. A mathematical definition of the co-occurrence matrix is as follows [4]:

- Given a position operator $P(i,j)$,
- let A be an $n \times n$ matrix
- whose element $A[i][j]$ is the number of times that points with grey level (intensity) $g[i]$ occur, in the position specified by P , relative to points with grey level $g[j]$.
- Let C be the $n \times n$ matrix that is produced by dividing A with the total number of point pairs that satisfy P . $C[i][j]$ is a measure of the joint probability that a pair of points satisfying P will have values $g[i], g[j]$.
- C is called a co-occurrence matrix defined by P .

Examples for the operator P are: “ i above j ”, or “ i one position to the right and two below j ”, etc.

This can also be illustrated as follows... Let t be a translation, then a co-occurrence matrix C_t of a region is defined for every grey-level (a, b) by [1]:

$$C_t(a,b) = \text{card} \{ (s, s+t) \in R^2 \mid A[s] = a, A[s+t] = b \}$$

Here, $C_t(a, b)$ is the number of site-couples, denoted by $(s, s+t)$ that are separated by a translation vector t , with a being the grey-level of s , and b being the grey-level of $s+t$.

For example; with an 8 grey-level image representation and a vector t that considers only one neighbour, we would find [1]:



Organized by

Dhaanish Ahmed Institute of Technology, KG chavadi, Coimbatore, India

1	2	1	3	4
2	3	1	2	4
3	3	2	1	1

Figure: Image example

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	0	0	0	0	0
2	0	1	0	2	0	0	0	0
3	0	0	1	1	0	0	0	0
4	0	1	0	0	1	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Figure: Classical Co-occurrence matrix

At first the co-occurrence matrix is constructed, based on the orientation and distance between image pixels. Then meaningful statistics are extracted from the matrix as the texture representation. Haralick proposed the following texture features:

1. Energy
2. Contrast
3. Correlation
4. Homogeneity
5. Entropy

Hence, for each Haralick texture feature, we obtain a co-occurrence matrix. These co-occurrence matrices represent the spatial distribution and the dependence of the grey levels within a local area. Each $(i,j)^{th}$ entry in the matrices, represents the probability of going from one pixel with a grey level of 'i' to another with a grey level of 'j' under a predefined distance and angle. From these matrices, sets of statistical measures are computed, called feature vectors .

Energy: It is a gray-scale image texture measure of homogeneity changing, reflecting the distribution of image gray-scale uniformity of weight and texture..

$$E = \sum_x \sum_y p(x,y)^2$$

p(x,y) is the GLC M

Contrast: Contrast is the main diagonal near the moment of inertia, which measure the value of the matrix is distributed and images of local changes in number, reflecting the image clarity and texture of shadow depth.

$$\text{Contrast} \quad I = \sum \sum (x - y)^2 p(x,y)$$

Entropy: It measures image texture randomness, when the space co-occurrence matrix for all values are equal, it achieved the minimum value.

$$S = -\sum_x \sum_y p(x, y) \log p(x, y)$$

Correlation Coefficient: Measures the joint probability occurrence of the specified pixel pairs.

$$\text{Correlation} = \frac{\sum(\sum((x - \mu_x)(y - \mu_y)p(x, y)/\sigma_x\sigma_y))}{\sqrt{\sum(x - \mu_x)^2 p(x, y) \sum(y - \mu_y)^2 p(x, y)}}$$

Homogeneity: Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.

$$\text{Homogeneity} = \sum(\sum(p(x, y)/(1 + |x - y|)))$$

Neural Network:

Neural networks are predictive models loosely based on the action of biological neurons.

The selection of the name “neural network” was one of the great PR successes of the Twentieth Century. It certainly sounds more exciting than a technical description such as “A network of weighted, additive values with nonlinear transfer functions”. However, despite the name, neural networks are far from “thinking machines” or “artificial brains”. A typical artificial neural network might have a hundred neurons. In comparison, the human nervous system is believed to have about 3×10^{10} neurons. We are still light years from “Data”.

The original “Perceptron” model was developed by Frank Rosenblatt in 1958. Rosenblatt’s model consisted of three layers, (1) a “retina” that distributed inputs to the second layer, (2) “association units” that combine the inputs with weights and trigger a threshold step function which feeds to the output layer, (3) the output layer which combines the values. Unfortunately, the use of a step function in the neurons made the perceptions difficult or impossible to train. A critical analysis of perceptrons published in 1969 by Marvin Minsky and Seymour Paper pointed out a number of critical weaknesses of perceptrons, and, for a period of time, interest in perceptrons waned.

Interest in neural networks was revived in 1986 when David Rumelhart, Geoffrey Hinton and Ronald Williams published “Learning Internal Representations by Error Propagation”. They proposed a multilayer neural network with nonlinear but differentiable transfer functions that avoided the pitfalls of the original perceptron’s step functions. They also provided a reasonably effective training algorithm for neural networks.

Types of Neural Networks:

- 1) Artificial Neural Network
- 2) Probabilistic Neural Networks
- 3) General Regression Neural Networks

DTREG implements the most widely used types of neural networks:

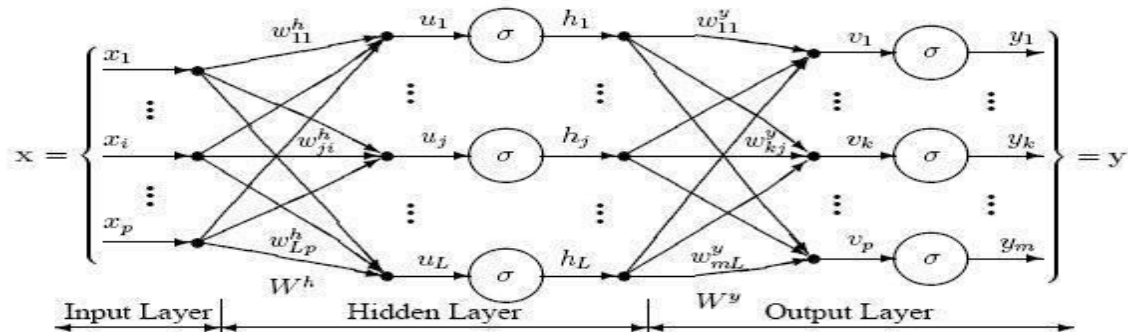
- a) Multilayer Perceptron Networks (also known as multilayer feed-forward network),
- b) Cascade Correlation Neural Networks,
- c) Probabilistic Neural Networks (NN)
- d) General Regression Neural Networks (GRNN).

Radial Basis Function Networks:

- a) Functional Link Networks,
- b) Kohonen networks,
- c) Gram-Charlier networks,
- d) Hebb networks,
- e) Adaline networks,
- f) Hybrid Networks.

The Multilayer Perceptron Neural Network Model

The following diagram illustrates a perceptron network with three layers:



This network has an **input layer** (on the left) with three neurons, one **hidden layer** (in the middle) with three neurons and an **output layer** (on the right) with three neurons.

There is one neuron in the input layer for each predictor variable. In the case of categorical variables, $N-1$ neurons are used to represent the N categories of the variable.

Input Layer — A vector of predictor variable values (x_1, \dots, x_p) is presented to the input layer. The input layer (or processing before the input layer) standardizes these values so that the range of each variable is -1 to 1 . The input layer distributes the values to each of the neurons in the hidden layer. In addition to the predictor variables, there is a constant input of 1.0 , called the *bias* that is fed to each of the hidden layers; the bias is multiplied by a weight and added to the sum going into the neuron.

Hidden Layer — Arriving at a neuron in the hidden layer, the value from each input neuron is multiplied by a weight (w_{ji}), and the resulting weighted values are added together producing a combined value u_j . The weighted sum (u_j) is fed into a transfer function, σ , which outputs a value h_j . The outputs from the hidden layer are distributed to the output layer.

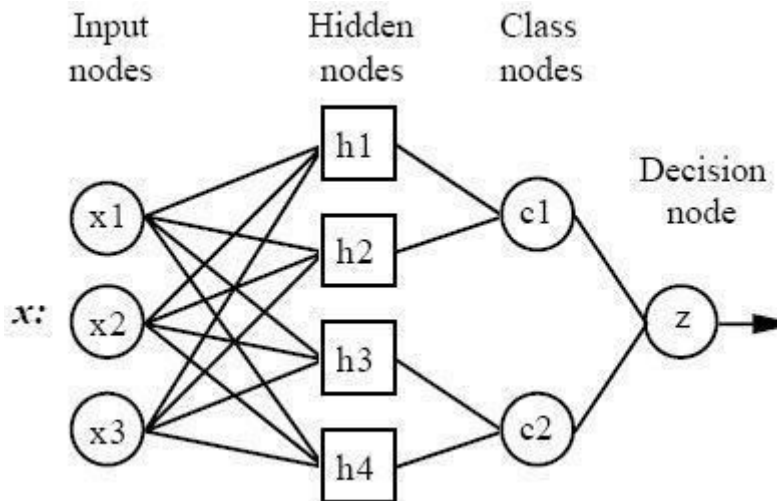
Output Layer — Arriving at a neuron in the output layer, the value from each hidden layer neuron is multiplied by a weight (w_{kj}), and the resulting weighted values are added together producing a combined value v_j . The weighted sum (v_j) is fed into a transfer function, σ , which outputs a value y_k . The y values are the outputs of the network.

If a regression analysis is being performed with a continuous target variable, then there is a single neuron in the output layer, and it generates a single y value. For classification problems with categorical target variables, there are N neurons in the output layer producing N values, one for each of the N categories of the target variable.

Neural Networks (NN):

Neural Network (NN) and General Regression Neural Networks (GRNN) have similar architectures, but there is a fundamental difference: networks perform classification where the target variable is categorical, whereas general regression neural networks perform regression where the target variable is continuous. If you select a NN/GRNN network, DTREG will automatically select the correct type of network based on the type of target variable.

Architecture of a NN:



All NN networks have four layers:

1. **Input layer** — There is one neuron in the input layer for each predictor variable. In the case of categorical variables, $N-1$ neurons are used where N is the number of categories. The input neurons (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.
2. **Hidden layer** — This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the x vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron’s center point and then applies the RBF kernel function using the sigma value(s). The resulting value is passed to the neurons in the pattern layer.
3. **Pattern layer / Summation layer** — The next layer in the network is different for NN networks and for GRNN networks. For NN networks there is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron’s category. The pattern neurons add the values for the class they represent (hence, it is a weighted vote for that category).

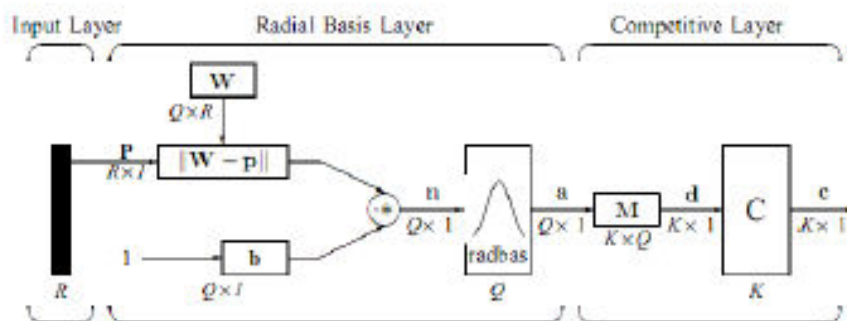
For GRNN networks, there are only two neurons in the pattern layer. One neuron is the denominator summation unit the other is the numerator summation unit. The denominator summation unit adds up the weight values coming from

each of the hidden neurons. The numerator summation unit adds up the weight values multiplied by the actual target value for each hidden neuron.

4. **Decision layer** — The decision layer is different for NN and GRNN networks. For NN networks, the decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

For GRNN networks, the decision layer divides the value accumulated in the numerator summation unit by the value in the denominator summation unit and uses the result as the predicted target value.

The following diagram is actual diagram or propose network used in our project



1) Input Layer:

The input vector, denoted as \mathbf{p} , is presented as the black vertical bar. Its dimension is $R \times 1$. In this paper, $R = 3$.

2) Radial Basis Layer:

In Radial Basis Layer, the vector distances between input vector \mathbf{p} and the weight vector made of each row of weight matrix \mathbf{W} are calculated. Here, the vector distance is defined as the dot product between two vectors [8]. Assume the dimension of \mathbf{W} is $Q \times R$. The dot product between \mathbf{p} and the i -th row of \mathbf{W} produces the i -th element of the distance vector $\|\mathbf{W} - \mathbf{p}\|$, whose dimension is $Q \times 1$. The minus symbol, “-”, indicates that it is the distance between vectors. Then, the bias vector \mathbf{b} is combined with $\|\mathbf{W} - \mathbf{p}\|$ by an element-by-element multiplication. The result is denoted as $\mathbf{n} = \|\mathbf{W} - \mathbf{p}\| \cdot \mathbf{b}$. The transfer function in NN has built into a distance criterion with respect to a center. In this paper, it is defined as $radbas(n) = 2 n e^{-n}$ (1) Each element of \mathbf{n} is substituted into Eq. 1 and produces corresponding element of \mathbf{a} , the output vector of Radial Basis Layer. The i -th element of \mathbf{a} can be represented as $a_i = radbas(\|\mathbf{W}_i - \mathbf{p}\| \cdot b_i)$ (2) where \mathbf{W}_i is the vector made of the i -th row of \mathbf{W} and b_i is the i -th element of bias vector \mathbf{b} .



Some characteristics of Radial Basis Layer:

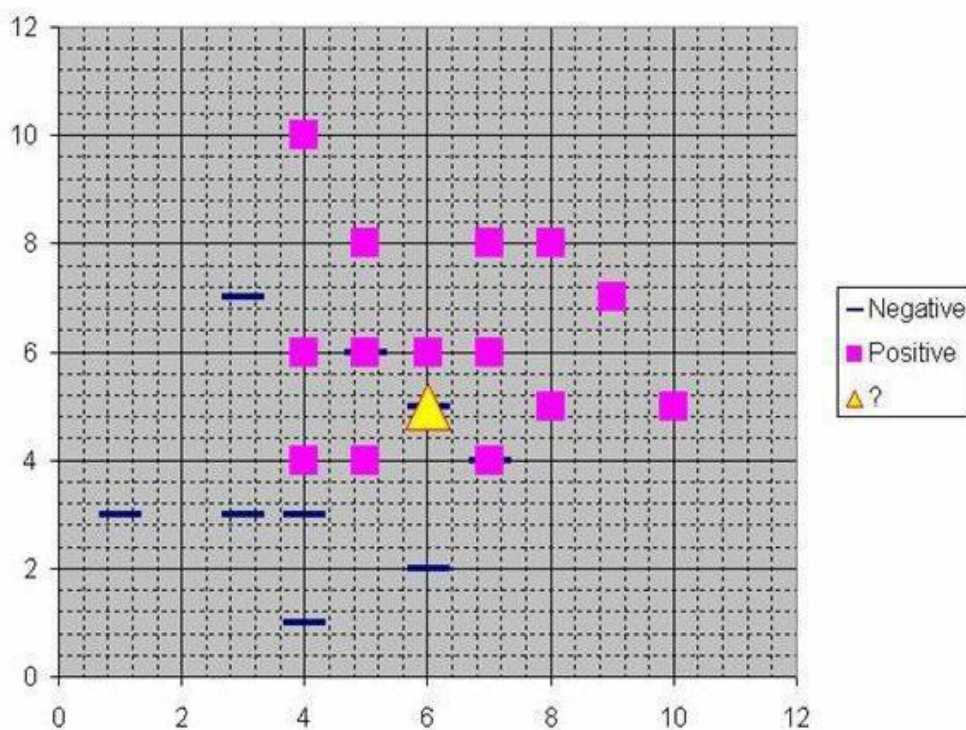
The i -th element of \mathbf{a} equals to 1 if the input \mathbf{p} is identical to the i th row of input weight matrix \mathbf{W} . A radial basis neuron with a weight vector close to the input vector \mathbf{p} produces a value near 1 and then its output weights in the competitive layer will pass their values to the competitive function. It is also possible that several elements of \mathbf{a} are close to 1 since the input pattern is close to several training patterns.

3) Competitive Layer:

There is no bias in Competitive Layer. In Competitive Layer, the vector \mathbf{a} is firstly multiplied with layer weight matrix \mathbf{M} , producing an output vector \mathbf{d} . The competitive function, denoted as \mathbf{C} in Fig. 2, produces a 1 corresponding to the largest element of \mathbf{d} , and 0's elsewhere. The output vector of competitive function is denoted as \mathbf{c} . The index of 1 in \mathbf{c} is the number of tumor that the system can classify. The dimension of output vector, K , is 5 in this paper.

How NN network work:

Although the implementation is very different, neural networks are conceptually similar to *K-Nearest Neighbor* (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure:





Organized by

Dhaanish Ahmed Institute of Technology, KG chavadi, Coimbatore, India

Assume that each case in the training set has two predictor variables, x and y . The cases are plotted using their x,y coordinates as shown in the figure. Also assume that the target variable has two categories, *positive* which is denoted by a square and *negative* which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values $x=6, y=5.1$. Should we predict the target as positive or negative?

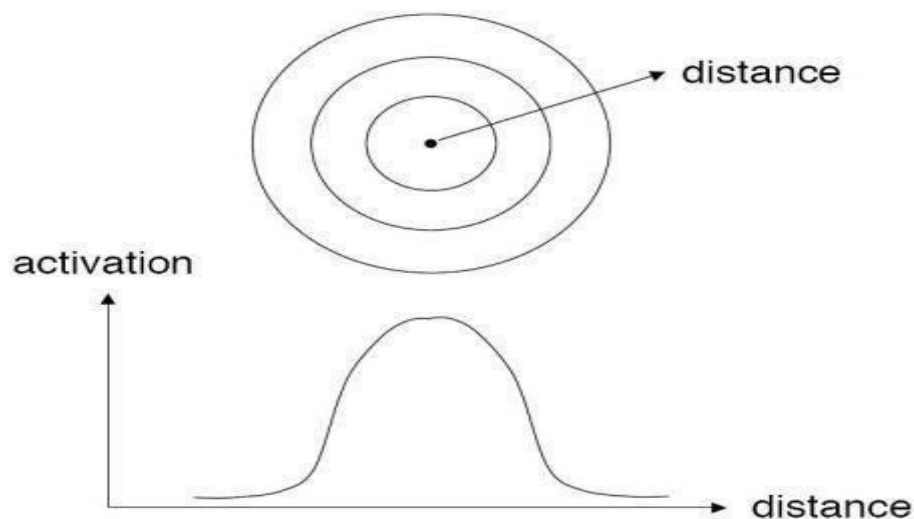
Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbour classification performed for this example depends on how many neighbouring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

A neural network builds on this foundation and generalizes it to consider all of the other points. The distance is computed from the point being evaluated to each of the other points, and a *radial basis function* (RBF) (also called a *kernel function*) is applied to the distance to compute the weight (influence) for each point. The radial basis function is so named because the radius distance is the argument to the function.

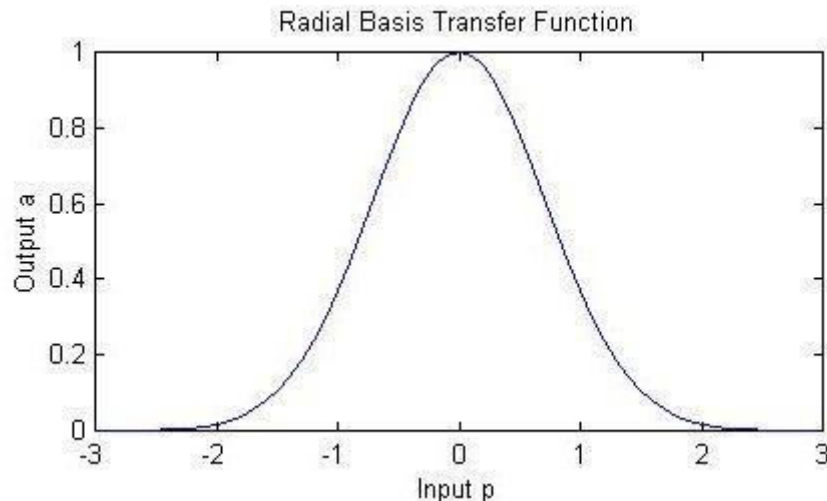
$$\text{Weight} = \text{RBF}(\text{distance})$$

The further some other point is from the new point, the less influence it has.



Radial Basis Function

Different types of radial basis functions could be used, but the most common is the Gaussian function:



Advantages and disadvantages of NN networks:

- 1) It is usually much faster to train a NN/GRNN network than a multilayer perceptron network.
- 2) NN/GRNN networks often are more accurate than multilayer perceptron networks.
- 3) NN/GRNN networks are relatively insensitive to outliers (wild points).
- 4) NN networks generate accurate predicted target probability scores.
- 5) NN networks approach Bayes optimal classification.
- 6) NN/GRNN networks are slower than multilayer perceptron networks at classifying new cases.
- 7) NN/GRNN networks require more memory space to store the model.

Removing unnecessary neurons

One of the disadvantages of NN models compared to multilayer perceptron networks is that NN models are large due to the fact that there is one neuron for each training row. This causes the model to run slower than multilayer perceptron networks when using scoring to predict values for new rows.

DTREG provides an option to cause it remove unnecessary neurons from the model after the model has been constructed.

Removing unnecessary neurons has three benefits:

1. The size of the stored model is reduced.
2. The time required to apply the model during scoring is reduced.
3. Removing neurons often improves the accuracy of the model.

The process of removing unnecessary neurons is an iterative process. Leave-one-out validation is used to measure the error of the model with each neuron removed. The neuron that causes the least increase in error (or possibly the largest reduction

in error) is then removed from the model. The process is repeated with the remaining neurons until the stopping criterion is reached.

When unnecessary neurons are removed, the “Model Size” section of the analysis report shows how the error changes with different numbers of neurons. You can see a graphical chart of this by clicking Chart/Model size.

There are three criteria that can be selected to guide the removal of neurons:

1. Minimize error – If this option is selected, then DTREG removes neurons as long as the leave-one-out error remains constant or decreases. It stops when it finds a neuron whose removal would cause the error to increase above the minimum found.
2. Minimize neurons – If this option is selected, DTREG removes neurons until the leave-one-out error would exceed the error for the model with all neurons.
3. # of neurons – If this option is selected, DTREG reduces the least significant neurons until only the specified number of neurons remain.

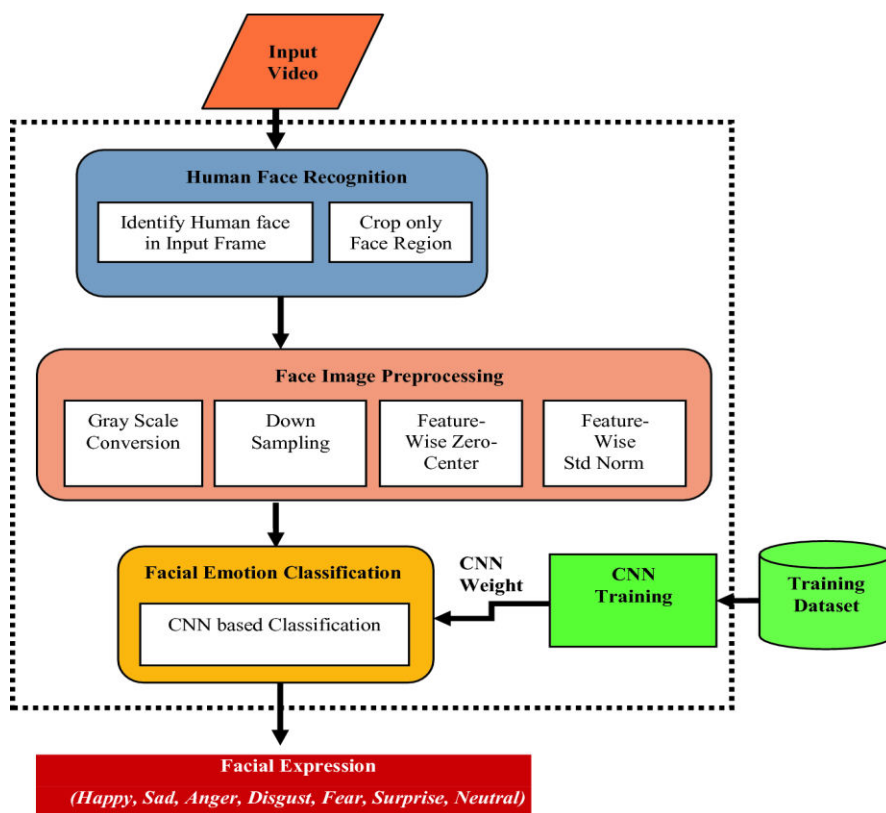


Fig No 3: Image Classification



Fig No:4 Face Recognition Techniques

VII. CONCLUSION AND FUTURE SCOPE

The aim of the project is to face recognition and face emotion by using deep learning technique. In this we proposed real time video surveillance, what human face expresses it in front of camera and they were recognising the face. This paper shows the high performance of classifier and feature extraction method that enhances the efficiency of system and improved the accuracy of facial emotion recognition. In this, seven universal emotions from different set of static images is analysed. In future we have increase the accuracy rate based on facial expression.

REFERENCES

- [1] Dubey, M., Singh, P. L., "Automatic Emotion Recognition Using Facial Expression: A Review," International Research Journal of Engineering and Technology, 2016.
- [2] W. Swinkels, L. Claesen, F. Xiao and H. Shen, "SVM point-based real-time emotion detection," 2017 IEEE Conference on Dependable and Secure Computing, Taipei, 2017.
- [3] H. Ebine, Y. Shiga, M. Ikeda and O. Nakamura, "The recognition of facial expressions with automatic detection of the reference face," 2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era (Cat.No.00TH8492), Halifax, NS, 2000, pp. 1091-1099 vol.2.
- [4] Arca, S., Campadelli, P., Lanzarotti, R.: An automatic feature-based face recognition system. In: Proceedings of the 5th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'04) (2004).
- [5] Face Profile Image Sequences," IEEE Trans. Systems, Man, and Cybernetics Part B, vol. 36, no. 2, pp. 433-449, 2006.
- [6] Emotion recognition from natural conversations based on cooccurrence matrix and frequency domain energy distribution features. In Proceedings of the 4th international conference on Affective computing and intelligent interaction-Volume Part II, ACII'11, pages 407-14, Berlin, Heidelberg, 2011. Springer-Verlag.
- [7] T. Senechal, V. Rapp, H. Salam, R. Segurier, K. Bailly, and L. Prevost. Facial action recognition combining heterogeneous features via multikernel learning. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 42(4):993-1005, 2012.
- [8] Geoffrey E. Hinton, Salakhutdinov RR. Reducing the dimensionality of data with neural networks, Science, 313(5786):504-7, 2006.



- [9] Happy, S. L., George, A., & Routray, A. (2012, December). A real time facial expression classification system using Local Binary Patterns. In Intelligent Human Computer Interaction (IHCI), 2012 4th International Conference on (pp. 1-5). IEEE.
- [10] Zhang, S., Zhao, X., & Lei, B. (2012). Facial expression recognition based on local binary patterns and local fisher discriminant analysis. WSEAS Trans. Signal Process, 8(1), 21-31.
- [11] Michel, P., & El Kaliouby, R. (2005). Facial expression recognition using support vector machines. In The 10th International Conference on Human-Computer Interaction, Crete, Greece.



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details