**Music Recommendation Using Emotions Recognition In python**

**Abstract-**

In this project we have made an emotion recogniseras in todays world stress management and for oneselves is very important , so with the help of this project , what we do is playing songs[STRESS BUSTORS] in accordance to the mood of the user which can be taken as input from various sources like cameras or to be more concise and precise a voice input as well.

**KEYWORDS :**

Song Recommendations, OpenCV , Emotion Recognition , mood/stress , FisherFace Classifier *.*

1. **INTRODUCTION TO IMAGE PROCESSING:**

Image processing is a method to analyze a image at its depth and extract some useful information out of the image provided. Image processing is very easy way through which an image is converted into digital form to perform various operations on it which in turn is quite similar to signal processing. The type of input given is a 2 D image It is a collection of numbers ranging from 0 - 255 which denotes pixel value.

|  |  |  |
| --- | --- | --- |
| 130 | 22 | 109 |
| 68 | 37 | 81 |
| 35 | 7 | 15 |

TABLE 1:This is a 2D array depicting the pixels of a image.

There are basically 3 steps to convert an image into a 2D matrix:

1. **Image scanning:** a image is taken as input and processed. It is then converted or processed in form of pixels as shown in the table.
2. **Processing and Enhancing it:**digitizer is used to convert the image into digital form it quantizes the input signals. For good resolutions the rate of sampling should be high and the quantization level should be high for human perception of different shades using different gray scale.
3. The result we obtain describes the properties of the image and further classifies it.
4. Eventually the image is converted into grayscale to be processed further.
5. **AboutOpen CV :**

OpenCV(Open Source Computer Vision)[12] is a popular computer vision library. Algorithms used in Open CV for facial recognition: The currently available algorithms are:Eigenfaces, Fisherfaces, Local Binary Patterns Histograms. Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 14 million. Usage ranges from interactive art, to mines inspection, stitching maps on the web or through advanced robotics.The whole series will take you through setting up OpenCV and python on your machine, carrying out some simple tasks on images and videos, and lastly, we will advance to detecting objects.

1. **Installation of necessary packages**

## Importing Libraries

The packages of Python 2.7 is downloaded and installed: Python 2.7, NumPy, Glob[12] and Random. Import all the packages and start working.

* 1. **Database Used**

This is the databases[2] that can be used for emotion detection. It is a database having various types of emotions[324] . CK+ database[2] has become one of the most widely used test-beds for algorithm development and evaluation. As it consists of extended-cohn-kanade-images, excel file of Landmarks, FACS\_labels, Emotion\_Labels. Classifier is trained with the help of this dataset.

1. **FISHERFACE CLASSIFIER :**

As we all now we cannot just directly process the image in its orginal format so we have to convert it into some pattern or reduce it to make it operatable.

So fisher face recognition is based on the reduction of the face space dimension using the principal of component analysis (PCA) and then will prefer applying the fishers linear discriminant(FDL) also known as linear Discriminant analysis(LDA) method to obtain feature of image characterstics. This algorithm uses minimum euclidean.

While the main goal of this algorithm is to make the image uinform in terms of size and fromat so that the image is ready to be used by the system.According to our analysis if the image in input is from the training dataset then obviosuly it will give the accuracy of 100% otherwise on an account of 73 faces test case 70 faces are recignised correctly and 3 faces are recognised incorrectly, there by it gives an accuracy of 93%.

***1* INTRODUCTION :**

No explanation should be required in that why to recognise face in as it is the most often used to distinguish the identity of individuals. Generally , the image recognition system is divide into 2 types , namely : feature based system and image beased system. In first features extracted from the face such as eye components , nose , mouth , etc. Which are the somehow modeled geometrically to compre and give the relationship between them,. while in the secind system image pixels which are represented n pca approach etc , are sued for training and image identification classification . feature extraction is definately an magical concept as It determines the emotions or diffnrelatioshiops between a unique face and datasets face all of these things as a whole can be explained using PRINCIPAL COMPONENTANANLYSIS METHOD(PCA) . the main aim of PCA method is that it projects data in the dircetion that has tyhegratest variations corresponding to largest eigenvalues fro the covariance matrix . but always there are 2 faces of coins so it got some darker face also that is it is less optimal in seperation between the classes. So in 1991 there is a solution fro this comes thatr is LDA method for face recognition ,ths method try to fnd a linear subspace that maximises the separatoin of 2 patterns classes according to fisher face criterion , this can be obtained by minimising the distaance of within the class distributuin matrix and maximising the split matrix spacing between the sub classes which actually results in maximum fisher criteron JF. Also with the lda method there is a limitation that if the data dimension is much higher than the number of training samples will cause Sw to be singular. So till now we are up with two brilliance in there own fields that is LDA and PCA but both of them got some limitattion also , So we definately requires some other mechanism to do it all more productively and efficiently. So , in 1997 Belheumer introduces the fisherface method used by our project for face recognition ,this method is the combonation of both PCA and LDA methods. But it styill have some problens that is complex computation process because of the conditio if the face image that is light ,postion like factors . the data used int his study is just general face samples so that we can make this project more practical and operatable.

***2.*RESEARCH METHODS :**

*2.1.* **The Design System:**

The results of feature extraction is used for matching the results with the face image in Fisherface .The correct recognition of the image isExpected to be done by the system.

* 1. **Process Design:**

2.2.1. **Data retrieval process:**

In this the main aim is to collect the data. The position the position of the face facing towards the upright and front position. Almost 50 photographs were taken with distance of +-100 c

***2.2.2.* Image Processing Process**:

**The design of process is divided into 2 phases:**

* **Image Preprocessing stage:**

In this stage the acquisition and RGB image is

converted into grayscale.The acquisition of image can be a 24-bit RGB image in JPG

format conversion is the done from 24 bit RGB

To 8 bit grayscale, BMP format.

The face data is divided into 2 parts that is for

Training purpose(**training dataset**) and test purpose(**test dataset**).

* **Image Processing stage:**

In this stage we use fisherface method to

generate feature vector of image data used by

system and then after by using the Euclidian

Distance formula we match the vector of traits

of training data set with vector of characteristic

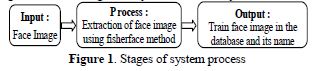
of test image data set.

***2.2.3.* Feature generation process:**

We have extracted the features of the image.The method used is the mixed from of PCA and LDA methods the merging of PCA and LDA results in fisherface Algorithm.

1. **Result and Discussion:**

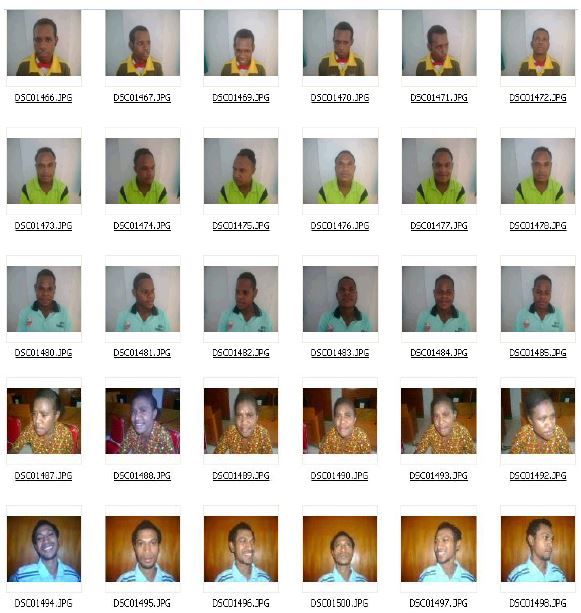
In this section we will discuss about the results produced by the fisherface algorithm. In the figure below we have shown how the algorithm works

**

*3.1* **Image Data**

*3.1.1* Image of the photograph result.

Given below is the sample photos of each individual represented by minimum of 5 sample images.



**Figure 2.**Example of an unprocessed image set

3.1.2. **Image Data Training**:

To ensure the success of the system, the system will be trained with several images



**Figure 3.**Some training set example

*3.1.3. Image Data testing:*



Figure 4.

*3.2.Feature generation process with fisherface method:*

**ASSUMPTION :**

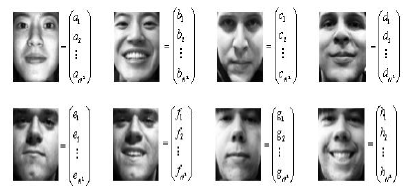
Size of rectangular face image with height = N and width = N and consists of h samples image

and C class {x1,x2,x3,…..xn}

3.2.1 **PCA Algorithm:**

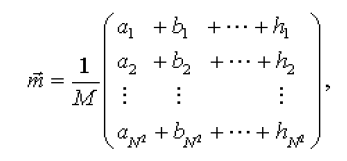
* In this algorithm the training images of size

N x N is converted into vector form of length size N^2



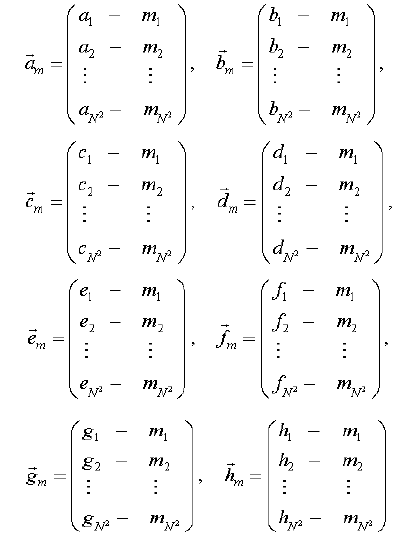
**Figure -** Training images converted into matrix form

* Calculating Average



Or in other words

Calculate Matrix A with the formula

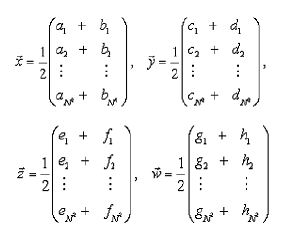


or written as

* Vector eigen (***eigVecs***) and value eigen (***eigenVals***) by using the methods svd of the matrix A.
* After that the ***eigenVecs*** are sorted and by PCA methods , pe.pe will be ***eigenfaces***

***3.***2.2 ***LDA Algorithm***:

* + Calculating mean of each class/person



* **Constructing scatter matrices:**

and the matrix within class Scatter matrix

*ScatW = S1+S2+S3+S4*

* Obtain the multiplication of matrices transpose of pe , (**PeT**) with ScatW and ScatB we get:

***Sbb=PeT\*ScatB\* Pe***

***Sww=PeT\* Pe***

* Finding eigenvectors (Vesbb) and generalized eigenvalues(***NewSww***) of (***Sbb,Sww***) and then sorting them.
* Projection back *VeSbb with pe*eigenfaces then forms up (***Pe \* VeSbb***) Output as **Fisherfase**Normalization = ***Pe\*VeSbb\*N***
* Find transpose of ***Pe\*VeSbb\*Nt***
* Calculate weight of each training image matrixU = Pe \* VeSbb \* Nt \*A.

Result of this process will be when the weight of each and every training image is in the eigenvector form that we will use to find the similarity with the face image/the face shown and will recognize it.

*3.2.3* **Identification algorithm:**

* Face image tested by size of *N\*N* is hence converted into column vector form*r1N^2*
* Normalization of facial image input to the image of training by finding the value of different matrix *inp*by subtracting the average value of training image.

Weight of the test image is calculated by multiplying the eigenval transpose matrix with the matrix

Φ*inp*matrix **( U***inp***) ;***U inp=VT*Φ*inp*

Using Euclidian distance we calculate the distance between image testing with training image.

*3.3* **System testing** :

To check whether the system is working properly or not the following tests are done

**3.3.1 Training process**

In this stage the aim is to generate the weight val. of each training image

**3.3.2. Image Recognition Process**

In this stage we see how big our system can easily recognize the test image or testing properly. Below are the results produced by the image recognition process

* **Training image same as testing image***:*

In this test same images were used for recognition of faces which we used during the training process and test process in terms of expression, position and image condition. The images used were 250 and the percent accuracy was 100%.

* ***The training image is not the same as the testing image*:**

In this test same images were not used for training and for testing process during recognition whether in terms of expression, position and condition of image but the data of images comes from same person’s face.

The percent accuracy came out to be around 93%

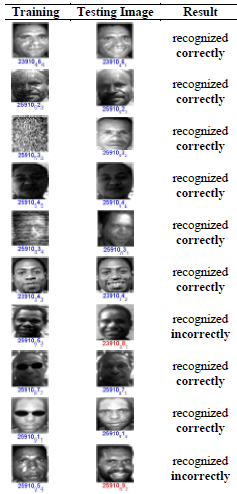
3.4. **Image recognition results:**

The following is an example of the results of facial recognition process with fisherface method can be

seen here.



Fig.4 Fisherface Reconstruction Opencv.[6]

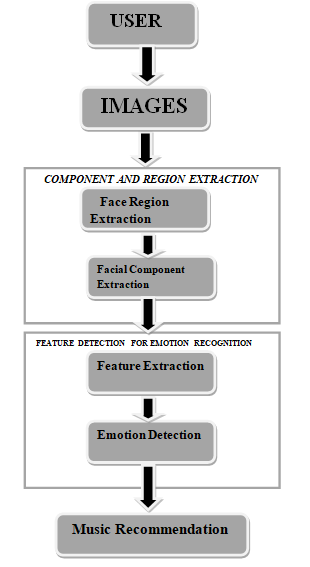


**Table 2**.example of fisher face image recognition results

**Conclusion:**

The fisher face model can accurately recognize faces upto accuracy of approximately >90%.

1. **Emotion Detection Procedure**

* Every possible required libray is imported. Dataset is downloaded. Add your own own expressive facial images if possible for better results or skip this part. Let the classifier have its fun.
* Dataset[2] is being arranged in emotions folder for training the classifier. Furthermore shape\_predictor\_68\_face\_landmarks.dat is used for facial feature extraction using FisherFace Classifier.
* Finally the dataset[2] is being split into training set and classification set. After accessing the trainer file (in form of .xml) training set is compiled using test set. To improve the efficiency either dataset[2] is incremented or increased training of data too contribute.
* Now during the project run first multiple images of user is being clicked for proficient classification then with help of trainer file image is classified happy, anger, sad, neutral, surprise.
* Then below self designed music recommendation recommends the song with alter the mood of user.

**Fig.Flowchart for Emotion recognition recommending music.**

# V. Different emotions that can be detected out of an image:

## A. Neutral

## B. Happy

## C. Anger

## D. Disgust

## E .Surprise

## F. Fear

## G. Sad

## H. Contempt

Below is the image of different emotions that can be recognized:



Fig.4Output of a sample image [9]

1. **Music Recommendation Algorithm Used:**

Song is recommended according to the mood of user. As described mood of user varies from happy, sad, disgust, angry, surprise. Dataset[2] consists of various states of minds songs. Songs are recommended on the algorithm that first holds a record of zero played songs assigned with value for its indication. The rest played ones consists of percentile of their ***respective*** regions (temper of mind). Main task of this recommender is to prioritize most likely as well as just published song.

Every region consists of two dimensional matrixes. Row one consists of songs of percentile greater than zero and other one consists of non-played songs. Row one elements are sorted in descending order.

Row one consists of most viewed songs whereas row second consists of equally likely or nearly equally viewed songs but still in descending order accordingly.

Now the questions arises how selected ones consists played ones with high percentile, low percentile and non-played ones? How recommender starts recommending songs at inception with biasing?

So at inception regional dataset is divided into equal halves. First half is assigned with static percentiles according to:-

Let percentile be say (p) then :

Let say no. of observations be (n) of regional dataset[2]. Dividing it into two halves. If n is even then into n/2, n/2 otherwise n/2, (n+1)/2. Considering in first half every element is arranged priority wised then percentiles assigned to elements from 1 to n/2 will be:-

Say n/2 be m then:

Percentile at index0, index1, index2, index, index 3, . . ,index(m-1) will be

*%*

*%*

*%*

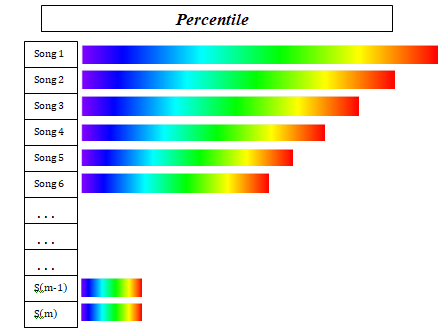
*%*

*. ..*

*. . .*

*. . .*

*%*

Figure: Depicts percentiles assigned percentiles.

Notations for array elements:

For first row:

a1, a2, a3, a4, . . . , a(m)

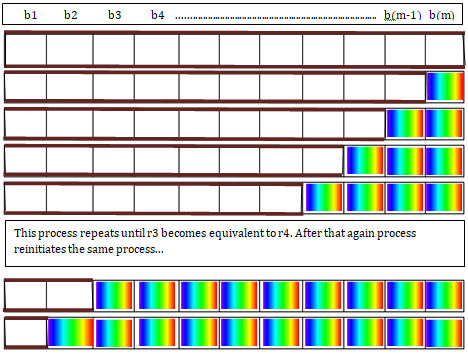
For second row:

b1, b2, b3, b4, . . . , b((m) or (m+1))

Now moving on to selection procedure of songs. Selection Array consists of four elements. First element is the a(1), Second element is a(m), third element is b1, fourth element is b((m) or (m+1)). And randomly one is selected between these four elements.

For first row selection procedure, if first element is chosen then percentile is divided the number of counts (times played). Second element i.e. lastelement is multiplied by the count factor (times played).

Once song is selected from row one then row alterations are to be made i.e. array is again sorted optimally O(n) complexity in descending order, as array is already sorted except one element after operation.

******

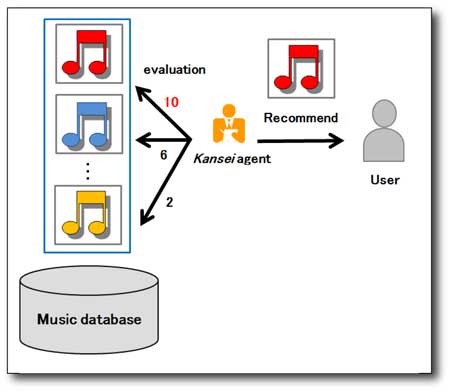
***Rainbow coloured regions shows discarded ones***

i.e***.*** played songs.

Whereas for second row it is as simple selection follows:

After every selection procedure, selected element is discarded for upcoming participation. So size of set decreases by one from chosen one.

If r3 == r4 then the process reinitiates with same credentials.



**Future scope :**

1. Definitely can be used as a start up plan.
2. What we want is just an high resolution camera , so we can make a device which actually contains a camera and just make a move to the market.
3. Why we let anything look into ourselves when we’re not in a good mood so here is the solution for that as well , most of us in any state of time love to see ourselves so it actually Can be embedded inside the mirrors.
4. Future scope also can contain iris recognition or emotion recognition according to pupil contraction and so on .

# Applications and Future Scope

Computer Vision is a very vast field which is still under developmental phase. Research work in this field is going at a rapid phase.

Emotion detection is an inseparable part of computer vision. Loads of tasks and processes can be performed if one can become aware about the intricacies and endless possibilities offered under the field of emotion detection.

Some common and widely used applications of emotion detection are:

## App and product development

Emotion recognition and music recommendation can play a huge role in optimizing various software engineering processes which comprises of testing of ease with which a product can be used. It’s a long-established fact that level of comfort with different software products depends hugely upon human mood. Thus, researching about different emotional states and making them feel stress free under difficult conditions can make humans make better choices by recommending them the best music .

**Improved learning practices**

Evidence suggests that part of emotional states vouches for better learning practices while the other part try to suppress them. The difference between the two groups possessing different emotional states is not so common to find. For example, positive emotional state is thought to be bad for learning purposes while slightly negative emotional state fosters analytical thinking and is also appropriate for carrying out critical tasks.

## Improvised web development

With such a large scale at which the internet and the demand of music ini music industry is expanding, service providers are interested in collecting tons and tons of data which can be extracted from the users. Correspondingly, all the content and advertisements are played based on the users’ profile. Subsequently, adding intricate details about the different human emotions and his/her choice of songs can provide much more precise behavioral models of different types of users and will make the music industry aware of wht kind of songs our current generation is into which will definitely help the singers to rise up and gain popularity.

## Immersive gaming

Video games constitute a large chunk of entertainment industry. Thus, in order to make these games much more intensive and intrusive, video game creators base their research on different types of human emotions and the gaming music realted to the type of games played. commonly found. In order to allure more and more players, video games are made in such a way that they incorporate human mood naturally into their game play.

**Challenges**

The main purpose is to detect various emotions in a given sample image. The most challenging part in this task was to determine the exact emotion when two emotions look quite similar, for e.g. “Disgust” being classified as “Sadness”, “Surprise” like “Happy” and so on. Now for eight different categories, the result was approximately 81.2% accurate which is quite well actually as our classifier learned quite a bit. So, we must see how can we increase its efficiency and accuracy.

If we look at our emotions list we can find out that we have only limited number of examples for “sad”, “fear” and “contempt”. By increasing the number of images for these emotions we can certainly increase optimization, or if we no longer consider these emotions in the list then optimization can be increased more than 90%.

The dataset that we used for our task was Cohn-Kanade (CK and CK+) Database. [6]

**Conclusion**

Artificial Intelligence can be used to solve intriguing tasks such as emotion detection, stress reliving although this task was quite convolute even more when using a great number of images. We humans also sometimes make a mistake while recognizing someone’s emotion so is our program. The optimum accuracy was nearly 83%.

In Present Work, we recognized different types of human emotions using Python 2.7, OpenCV& (CK and CK+) Database [6] and got some interesting insight about it

**REFRENCES:**.

[1] - Zhao, W., Chellappa, R., Phillips, P.J. and Rosenfeld, A., 2003. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, *35*(4), pp.399-458.

[2] - Kanade, T., Cohn, J. F., & Tian, Y. (2000). [Comprehensive database for facial expression analysis](http://ieeexplore.ieee.org/document/840611/?arnumber=840611). Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France, 46-53.

[3] - https://www.bytefish.de/blog/fisherfaces

(for green one)

[4] - https://www.ibmastery.com/blog?page=4(emotion tok)

[5] - https://www.slideshare.net/wolf/eigenfaces-and-fisherfaces

[6] - https://docs.opencv.org/2.4/ modules/ contrib/ doc/facerec/facerec\_tutorial.html.

[7] - M.Inoue, H.Takenouchi, M.Tokuamaru, ” Music Recommendation System Improvement Using Distributed Genetic Algorithm”, 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems and 2016 17th International Symposium on Advanced Intelligent Systems (SCIS&ISIS 2016), pp.627-630, 2016-08 (Sapporo, Hokkaido, Japan)

[8] - http://www.kis.kansai-u.ac.jp/ ?page\_id =185& lang=en

[9] - Kim & Joo, Young Hoon & Bae Park, Jin. (2015). Emotion Detection Algorithm Using Frontal Face Image

[10] - Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann and Shrikanth Narayanan (2004). Analysis of Emotion Recognition using Facial Expressions, Speech and Multimodal Information

[11] - https://docs.python.org/2/library/glob.html

[12] - https://opencv.org/

[13] - V Gent, P. (2016). Emotion Recognition with Python, OpenCV and a Face Dataset. *A tech blog about fun things with Python and embedded electronics.*  Retrieved from:  
http://www.paulvangent.com/2016/04/01/emotion-recognition-with-python-opencv-and-a-face-dataset/