https://dx.doi.org/10.22161/ijaers/3.9 *ISSN: 2349-6495(P) | 2456-1908(O)*

Emotion Based Music Player – XBeats

Aditya Gupte, Arjun Naganarayanan, Manish Krishnan

Department of Computer Engineering, SIES GST, Mumbai University, Navi Mumbai, India

Abstract— *This paper showcases the development of an Android platform based application named XBeats which acts as a Music Player working on Image Processing fundamentals to capture, analyze and present music as per the emotion or mood of the user using this application. The Android application was developed using the Android SDK software and OpenCV software was used to implement facial recognition algorithms and cascades. The unique aspect of this project is that it focuses on facial recognition on the Android platform unlike that on Computer systems which use commonly available softwares for the same. This paper also provides comparison between use of various classification algorithms used for facial detection.*

Keywords— XBeats, Android SDK, OpenCV, Facial Recognition, Image Processing.

I. INTRODUCTION

Facial expressions give an important clue about the person's emotion/mood. Computer systems using this phenomenon could effectively work upon human machine interaction (HMI) fundamentals to further bridge the gap between humans and technology. Facial emotions also could prove to be an effective tool in areas of security and entertainment. The work describes the development of Emotion Based Music Player named XBeats, which is a mobile application developed on Android platform meant for users to minimize their efforts in managing large playlists and listening to music. Generally, people happen to own a large number of songs in their Music Player and usually play the song which happens to be in front of them rather than searching for the song in the enormous library. This activity most often results in disapproval because the song may not map to the users' current emotion. Moreover, there is no commonly used application in the Google Play Store which is able to play songs based on the current emotions of the user. In our project, once the emotion is detected, playlist matching the emotion will be presented to the user. The following model contains two moods viz. Happy and Sad. The input to the model is still images of user which are further processed to determine the mood of user. The images are captured using front-camera of the mobile device. The system will capture the image of the user when the user clicks on the **SenseMe** option. **SenseMe** is the button provided in the music player application which enables the front-camera to identify the emotion of the user.

II. OBJECTIVE OF RESEARCH

In the existing music players, the user has to manually select the songs from the device to listen to them. This proves to be problematic if the user has many tracks (which is usually the case). It is really very difficult for the user to manage such large playlist. Thus, it would be inconvenient for the user to search the song that would suit to his mood and create a playlist. The traditional music player is time consuming and non-reliable if the playlist is too large. If the user gets bored of choosing the songs manually he could choose the shuffle option. The uncertainty here is that the song played randomly as a result of using the **Shuffle** button may not suit the current mood of the user. The other way is that the user needs to manually classify songs according to different types of emotions which is hectic and can be totally rendered unnecessary by the work of this project.

2.1 Limitations of current systems

1. It requires user to manually find and play the song from the playlist.
2. The current music player is hence not reliable and is time consuming when the number of songs exceeds 100.
3. Even if user uses the '**Shuffle'** button, the resultant song may not match the current mood of the user leading to the possibility of user not listening to music at all!.
4. User needs to manually classify the songs as per the emotions and then select a particular emotion to listen to songs of that category. This always yields unpleasant user reactions.

III. LITERATURE REVIEW

3.1 Skin Color based Face Detection Algorithm

Skin color based Face Detection algorithms is one of the most popular and robust face detection algorithms to be used in the field of Image Processing and was the major catalyst for providing valuable information for achieving the completion of our project. The Skin color based Face

https://dx.doi.org/10.22161/ijaers/3.9 ISSN: 2349-6495(P) | 2456-1908(O)

Detection algorithms typically work on three color domains.

3.1.1 RGB Color Space

The RGB Color space is primarily composed of three primary additives: Red, Green and Blue. Spectral components of these colors combine additively to produce a resultant color. The RGB model is represented by a 3-dimensional cube with red green and blue at the corners on each axis. Black is at the origin. White is at the opposite end of the cube. The gray scale follows the line from black to white. In a 24-bit color graphics system with 8 bits per color channel, red is (255, 0, 0). On the color cube, it is (1, 0, 0). The RGB model simplifies the design of computer graphics systems but is not ideal for all applications. The red, green and blue color components are highly correlated. This makes it difficult to execute some image processing algorithms. Many processing techniques, such as histogram equalization, work on the intensity component of an image only.

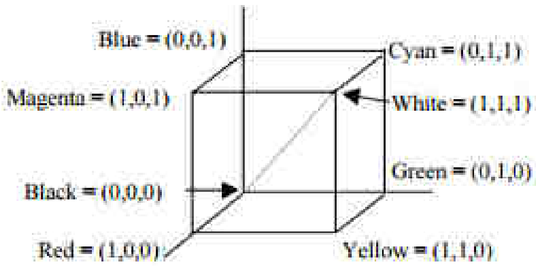

Fig 1: The RGB Color cube

3.1.1.1 Skin Color based Face Detection in RGB Domain

Crowley and Coutaz [1] said one of the simplest algorithms for detecting skin pixels is to use skin color algorithm. The perceived human color varies as a function of the relative direction to the illumination. The pixels for skin region can be detected using a normalized color histogram, and can be further normalized for changes in intensity on dividing by luminance. And thus converted an [R, G, B] vector is converted into an [r, g] vector of normalized color which provides a fast means of skin detection. This gives the skin color region which localizes face. As in [1], the output is a face detected image which is from the skin region. This algorithm fails when there some more skin region like legs, arms, etc.

Table 1: Result of classifier in RGB Domain

No. of Images	False Detection Rate (%)	False Dismissal Rate (%)
1100	25.45	18.09

The above table shows that 1100 images were used in training the skin color classifier for detecting the face under RGB Domain and results show that the classifier does not work that well in the RGB domain because the overall accuracy of the classifier is 56.46%. This is due to high False Detection and False Dismissal rates.

3.1.2 YCbCr Color Space

YCbCr color space has been defined in response to increasing demands for digital algorithms in handling video information, and has since become a widely used model in a digital video. It belongs to the family of television transmission color spaces. The family includes others such as YUV and YIQ. YCbCr is a digital color system, while YUV and YIQ are analog spaces for the respective PAL and NTSC systems. These color spaces separate RGB (Red-Green-Blue) into luminance and chrominance information and are useful in compression applications however the specification of colors is somewhat unintuitive. The Recommendation 601 specifies 8-bit (i.e. 0 to 255) coding of YCbCr, whereby the luminance component Y has an excursion of 219 and an offset of +16. This coding places black at code 16 and white at code 235. In doing so, it reserves the extremes of the range for signal processing footroom and headroom. On the other hand, the chrominance components Cb and Cr have excursions of +112 and offset of +128, producing a range from 16 to 240 inclusively.

3.1.2.1 Skin color based Face Detection in YCbCr Domain

Studies have found that pixels belonging to skin region exhibit similar Cb and Cr values. Furthermore, it has been shown that skin color model based on the Cb and Cr values can provide good coverage of different human races. The thresholds be chosen as [Cr1, Cr2] and [Cb1, Cb2], a pixel is classified to have skin tone if the values [Cr, Cb] fall within the thresholds [3] The skin color distribution gives the face portion in the color image. This algorithm is also having the constraint that the image should be having only face as the skin region. Below is the table which shows the result when 1100 images are used to train the classifier.

Table 2: Result of classifier in YCbCr Domain

No. of Images	False Detection Rate (%)	False Dismissal Rate (%)
1100	12.82	3.27

The above table depicts the result of skin color classifier for detecting faces in YCbCr domain. Similar working as in the above case of RGB but here the accuracy is found to be 83.91% which is far better than that of RGB. The above table also shows that Rate of False Detection is high and Rate of False Dismissal is low which leads to bit low accuracy in facial detection.

3.1.3 HSI Color Space

Since hue, saturation and intensity are three properties used to describe color, it seems logical that there be a

https://dx.doi.org/10.22161/ijaers/3.9 ISSN: 2349-6495(P) | 2456-1908(O)

corresponding color model, HSI. [4] When using the HSI color space, you don't need to know what percentage of blue or green is required to produce a color. You simply adjust the hue to get the color you wish. To change a deep red to pink, adjust the saturation. To make it darker or lighter, alter the intensity. Many applications use the HSI color model. Machine vision uses HSI color space in identifying the color of different objects. Image processing applications such as histogram operations, intensity transformations and convolutions operate only on an intensity image. These operations are performed with much ease on an image in the HSI color space. The hue (H) is represented as the angle 0, varying from 0 to 360. Saturation (S) corresponds to the radius, varying from 0 to 1. Intensity (I) varies along the z axis with 0 being black and 1 being white. When S = 0, color is a gray value of intensity 1. When S = 1, color is on the boundary of top cone base. The greater the saturation, the farther the color is from white/gray/black (depending on the intensity). Adjusting the hue will vary the color from red at 0, through green at 120, blue at 240, and back to red at 360. When I = 0, the color is black and therefore H is undefined. When S = 0, the color is grayscale. H is also undefined in this case. By adjusting I, a color can be made darker or lighter.

3.1.3.1 Skin Color based Face Detection in HSI Domain

Kjeldson and Kender defined a color predicate in HSV color space to separate skin regions from background [2]. Skin color classification in HSI color space is the same as YCbCr color space but here the responsible values are hue (H) and saturation (S). Similar to above the threshold be chosen as [H1, S1] and [H2, S2], and a pixel is classified to have skin tone if the values [H, S] fall within the threshold and this distribution gives the localized face image. Similar to above two algorithms, this algorithm is also having the same constraint. Thus, the skin color based face detection algorithm is developed by combining the before mentioned three algorithms. It has been stated that the above three algorithms work very well under the condition that there is only one face is present in the image. In the implementation of the algorithms there are three main steps are -

1. Classify the skin region in the color space.
2. Apply threshold to mask the skin region.
3. Draw bounding box to extract the face image.

Table 3: Result of classifier in HSI Domain

No. of Images	False Detection Rate (%)	False Dismissal Rate (%)
1100	14.55	3.18

The above table showcases that HSI color space is also good in classifying skin color region. The accuracy found in HSI Color space is 82.27% which is nearly equivalent to that in YCbCr color space. Similar to YCbCr, in this color space also false detection rate is high and dismissal rate is low.

3.2 Facial Detection using Haar cascade

Object Detection using Haar feature-based cascade classifiers is an effective object detection method. It is a machine learning based approach. [6] In this approach a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. First of all, lot of positive images (images of faces) and negative images (images without faces) are used to train the classifier. Then features are extracted from it. For this, Haar features shown in below image are used. They are just like our convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle. [5] Now all possible sizes and locations of each kernel are used to calculate plenty of features. But among all these features we calculated, most of them are irrelevant. In the figure below top row shows two good features. The first feature selected seems to focus on the property that the region of the eyes is often darker than the region of the nose and cheeks. The second feature selected relies on the property that the eyes are darker than the bridge of the nose. Now, common Haar features are shown in the diagram below. They are classified into Edge, Line and Four-Rectangle features.

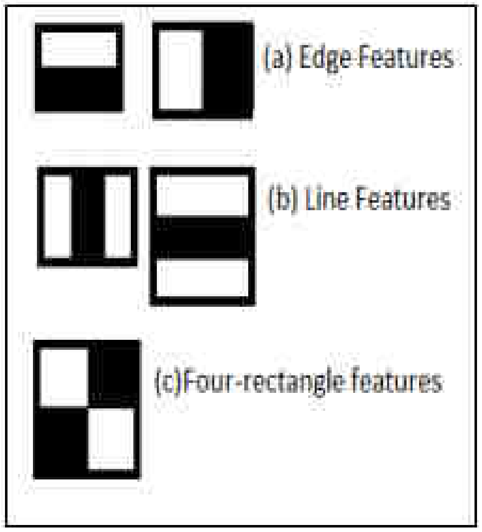

Fig 2: Common Haar Feature

3.2.1 Extracting Effective features

In this module, first System will capture the image from webcam or suitable device. Then the input image is first checked for the facial features. [7] In case if the image does not contain human features, then it does not detect it. If the input image contains Human features, then it detects the features. Face is detected from image as shown in Fig3.

https://dx.doi.org/10.22161/ijaers/3.9 ISSN: 2349-6495(P) | 2456-1908(O)

Fig 3: Face Extraction

3.2.1.1 Lip feature extraction

Up till now we will have an image which contains lip portion of the face. Now the next step is to extract the expression features from lip. To extract the feature we just have to measure the distance between upper lip & lower lip. Also the system will consider the position of contour points of lip. By lip feature we can significantly determine two features happy & surprise mood. The lip from lower face is detected as shown in Fig 4. This feature is particularly useful to detect the precise mood of the user but still difficulties arise because the exact location and movement of lips vary from person to person, so this feature greatly has its drawbacks.

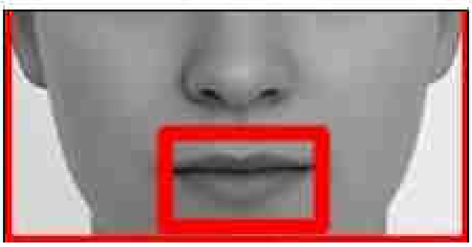

Fig 4: Lip Detection

3.2.1.2 Eye Feature Detection

Even with an upper face we can obtain certain facial feature like sleepy and surprised. For this again we have to calculate distance major axis and minor axis considering that eye as an ellipse. The eye is detected as shown in Fig 5.

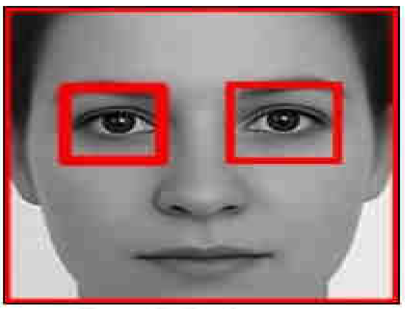

Fig 5: Eye feature Detection

IV. PROPOSED METHODOLOGY

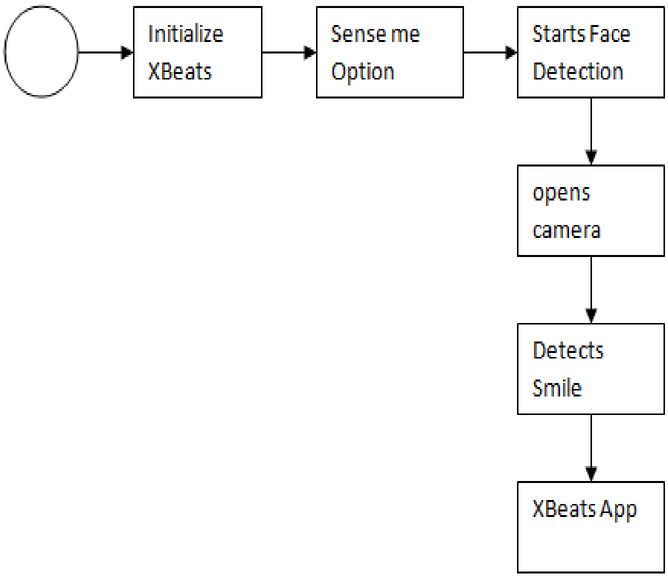

Fig 6: The proposed flow diagram of execution.

The above figure shows how our project works. It works and produces the correct output in 6 steps which are thoroughly explained below.

1. Initialize XBeats

In this step, the user taps on the XBeats icon on his App Tray (Home Screen) where the app is located alongwith other apps which are displayed to start the overall process.

2. SenseMe option

In this part, the user clicks on the SenseMe button provided in the app. This SenseMe button is the most crucial aspect of this entire project as it enables the Front-camera of the device and the module of Face Detection starts therein.

3. Starts Face Detection

This module is the link between the UI (The Music Player) and the Back-end processing (Face Detection) of the project. The Face Detection module runs alongwith the camera interface code defined in the Android SDK. This module was coded using OpenCV v. 3.1.0

4. Opens Camera

The front camera is opened at it looks for faces and once face is detected, it looks whether the user is smiling or not i.e. it checks whether the user is happy or sad. This module was coded using OpenCV v. 3.1.0

5. Detects Smile

If the user is smiling i.e. if the user is happy while using this app, it draws a green rectangle of dimensions specified in the code, along the border of the lips to show the user that happy emotion is detected. This module also completely works on OpenCV v. 3.1.0

https://dx.doi.org/10.22161/ijaers/3.9 ISSN: 2349-6495(P) | 2456-1908(O)

6. XBeats App

Once the Haar cascade detects smile, it presents a list of happy songs to the user and automatically plays the first song on the list for the user back in the XBeats App where it functions as a Music Player. In this activity, the camera is closed.

V. ANALYSIS OF PROPOSED SYSTEM

The proposed model is able to extract user's expressions and thus detect user's emotion. This music player will then play the songs according to the category of emotion detected. It is aimed to provide a satisfaction to music lovers to enrich their music experience. This application uses face detection and mood recognition to identify the user's face and then determine the user's mood and based on this, the application plays songs suitable to the user's mood. The face detection and feature extraction is based on OpenCV and Haar cascade classifier. Both algorithms are combined in order to give better accuracy. These implementations are designed in order to generate a playlist according to the user moods. This proposed system is of great use for the user having a large playlist. This system requires only the user expressions to play the songs.

5.1 Advantages of Proposed System

1. It requires only the user expressions to play the songs.
2. Reliable & User-friendly.
3. No trouble of troublesome selection of songs.
4. Use of Haar Cascade for detecting Smile is very accurate and presents very low false postitive detection rate.

Thus all the user needs to do is tap on the **SenseMe** button and the program will do the rest of presenting the user with a list of songs that match his/her current mood. Thus this project is very robust and appealing.

VI. RESULTS AND DISCUSSION

6.1 Screenshots

1. User selects the Music Player application "XBeats" from the App tray.

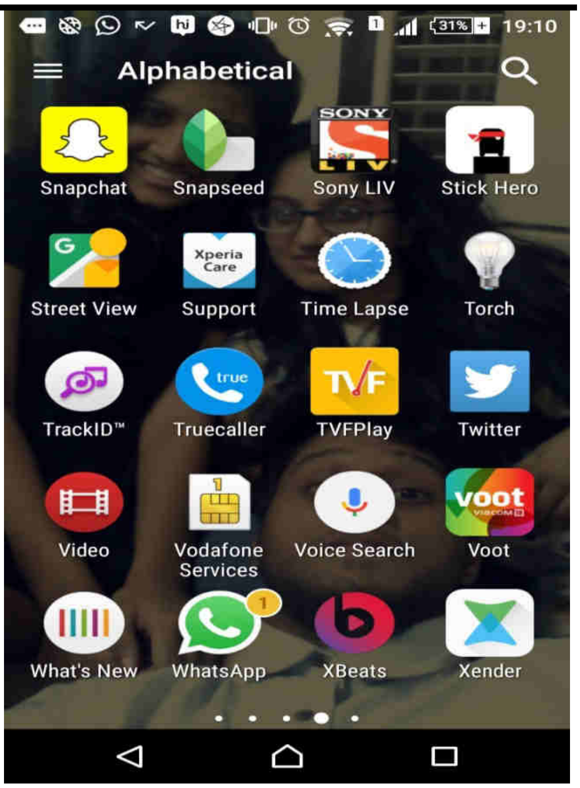

Fig 7: XBeats icon on the home screen

2. Once the app is opened in the mobile device, this is the front end of the music player. It displays all mp3 files present in the SD Card of the mobile device. User can either select a song manually or can click on the SenseMe Button placed in the media player to enable mood detection and let the app play a song for him according to his mood.

https://dx.doi.org/10.22161/ijaers/3.9 *ISSN: 2349-6495(P) | 2456-1908(O)*

Fig 8: Xbeats UI

3. SenseMe button lets the application detect the mood of the user according to his/her facial expression. It opens the front camera of the mobile device

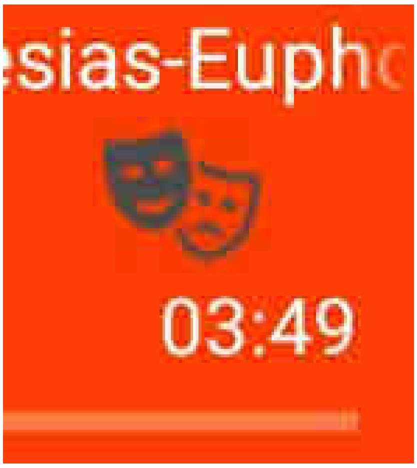

Fig 9: SenseMe button

4. The front camera of the mobile device is initialized. If the user smiles, then the application draws a rectangular box around the smile to detect the happy mood of the user.

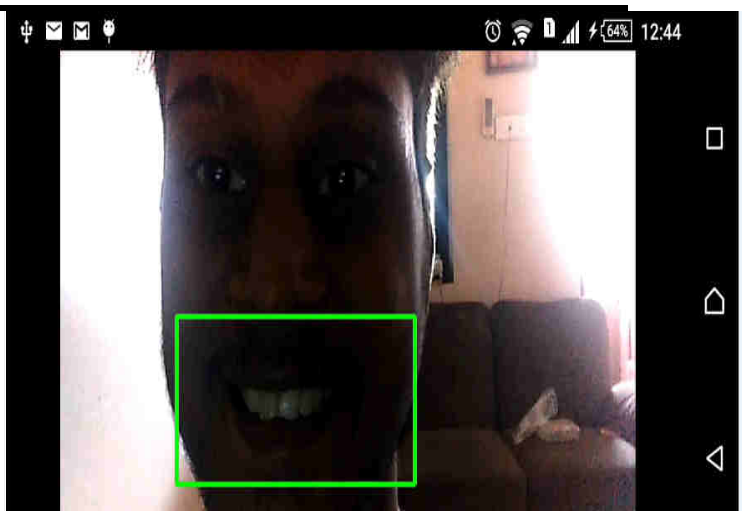

Fig 10: Smile Detection

5. The application identifies the happy/sad mood of the user and appropriately plays a song from their playlist to suit their mood.

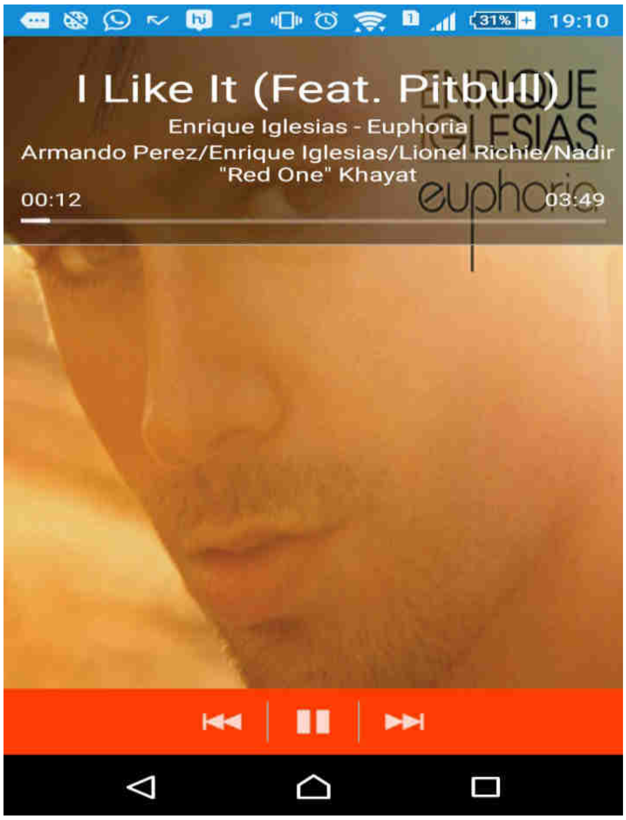

Fig 11: Music matching to the mood is played

VII. CONCLUSION

The existing systems do not provide the user with the facility to listen to songs according to his/her mood. The user has to manually select the songs that he/she wants to listen. Thus, the application developed will reduce the efforts of user in creating and managing playlist. It will provide better enjoyment to the music listeners by providing the most suitable or appropriate song to the user according to his/her current emotion. It will not only help user but also the songs are systematically sorted.

7.1 Future Scope for Implementation

1. Facial recognition can be used for authentication purpose.
2. Moods such as frustrated, amazed, ecstatic can also be detected.
3. Can detect sleepy mood while driving.
4. Can be used to determine mood of physically challenged & mentally challenged people.

REFERENCES

[1] Crowley, J. L. and Coutaz, J., “Vision for Man Machine Interaction,” Robotics and Autonomous Systems, Vol. 19, pp. 347-358 (1997).

[2] Kjeldsen, R. and Kender., J., “Finding Skin in Color Images,” Proceedings of the Second International Conference on Automatic Face and Gesture Recognition, pp. 312-317 (1996).

[3] Cahi, D. and Ngan, K. N., “Face Segmentation Using Skin-Color Map in Videophone Applications,” IEEE Transaction on Circuit and Systems for Video Technology, Vol. 9, pp. 551-564 (1999).

[4] A Robust Skin Color Based Face Detection Algorithm by Sanjay Kr. Singh , D. S. Chauhan , Mayank Vatsa , Richa Singh, Tamkang Journal of Science and Engineering, Vol. 6, No. 4, pp. 227-234 (2003).

[5] “Generating Music Playlist Based on Facial Expression” US 8094891 B2 Jan 10, 2012.

[6] Facial feature detection using Haar classifiers, Phillip Ian Wilson, Dr. John Fernandez

[7] S. Dornbush, K. Fisher, K. McKay, A. Prikhodko and Z. Segall ‖Xpod- A Human Activity and Emotion Aware Mobile Music Player‖, UMBC Ebiquity, November 2005.