Music Controller using Gesture Recognition to Control Music Playback

Aditya R. Parab[1] Janhavi R. Ghosalkar[2] V. Akshay B. Medge[3] Prof. Mamta Meena[4]
[1,2,3,4]Department of Computer Engineering
[1,2,3,4]Atharva College of Engineering

Abstract— By keeping in mind the similarities of human hand shape with four fingers and one thumb, this paper aims to present a real time system for hand gesture recognition on the basis of detection of some meaningful shape based features like orientation, status of fingers, thumb in terms of raised or folded fingers of hand and their respective location in image. A primary goal of gesture recognition research is to create a system which can identify specific human gestures and use them to convey information or for device control. The proposed system mainly focuses on scenarios where we are multitasking that is working on many applications at a time or running various programs at a time on our desktop and along with that listening to music in background that is music being played in one of the windows. At such times if we wish to pause or switch a particular music track we have to make some movements like switching to the music window and doing the desired operation. This process is bit long and time consuming as well. Suppose if you can do this operation without switching to the media player and by doing just one hand movement, it will save some time and also keep you linked with the work you doing currently. With the help of Music Controller, one can simply wave or do a simple gesture of hand movement in front of the webcam which will in turn switch or pause the particular music track that was being played.

Key words: Control Music Playback, Music Controller using Gesture Recognition

I. INTRODUCTION

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Controlling a computer using hand gestures has long been the stuff of science fiction, the most memorable occasion being the movie Minority Report, where Chief John Anderton (played by Tom Cruise) controls a computer by deftly moving his hands around in a 3D space.

It is now possible to control the music on your computer simply by making a few hand gestures. Through the use of a few simple hand movements, you can play, pause, forward or replay your music. Gesture-based technology has been booming the past several years, especially with the creation of the Microsoft Kinect for PCs and the Xbox 360. Computer-integrated programs can be something for the future, especially because it simplifies tasks such as changing a song while you're reading something important.

II. LITERATURE REVIEW

Hand Gesture recognition system provides us an innovative, natural, user friendly way of interaction with the computer which is more familiar to the human beings.

Hand gesture has the natural ability to represents ideas and actions very easily, thus using these different hand shapes, being identified by gesture recognition system and interpreted to generate corresponding event, has the potential to provide a more natural interface to the computer system.

By keeping in mind the similarities of human hand shape with four fingers and one thumb, this paper aims to present a real time system for hand gesture recognition on the basis of detection of some meaningful shape based features like orientation, centre of mass (centroid), status of fingers, thumb in terms of raised or folded fingers of hand and their respective location in image.

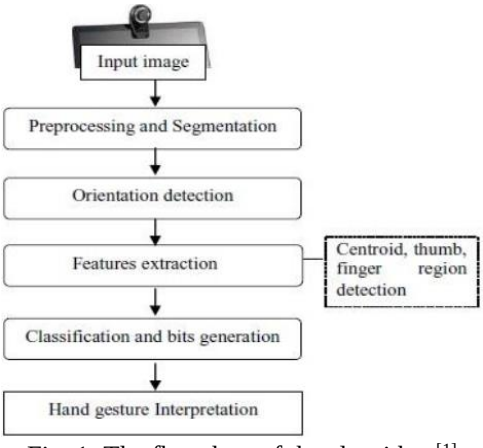

Fig. 1: The flowchart of the algorithm [1]

Gesture recognition does not consider any skin color and/or shape of fingers. Once fed in to the system, the data is stored as per the given instruction and thus can be given the necessary commands to act according to the gestures.

The algorithm "Weighted Averaging Analysis" can be used in our project which gives more efficient result. About 96% of the signs are correctly classified. Such results are so efficient that it does not need any training like in neural networks. The processing is quite fast given that no sophisticated calculus is required while running the program. The memory requirement is also very less as we are doing statistical analysis and not the database matching like in traditional image processing.[1]

III. IMPLEMENTATION

A. Collecting the pictures:

First of all, it will be necessary to collect pictures of all the gestures that will be supported by the Gesture recognition system.

B. Finding the hand in the picture:

According to the requirements, the web camera is not supposed to move. This piece of information gives a huge advantage that allows simplifying the zooming process. Indeed, it implies that the background is more or less always the same. In all what follows, it will be supposed that in the picture, we can just find the hand and the background: no other object should be present. After processing noise removal, the resulting picture will be black almost everywhere except where the hand is. So, zooming can then be easily realized by cropping areas whose pixel values are close to 0.

C. Detection of Hand:

- The detection of the Hand in the image can be summarized as follows:
- Firstly, the hand is detected using the background subtraction method and the result of hand detection is transformed to a binary image.
- Then, the fingers and palm are segmented so as to facilitate the finger recognition. Moreover, the fingers are detected and recognized. Last, hand gestures are recognized using a simple rule classifier.
- The images are captured with a web camera. These hand images are taken under the same condition. Since the background of these images is identical, it is easy and effective to detect the hand region from the original image using the background subtraction method.
- Background subtraction focusses on removing the existential image background so as to focus more on the gesture extraction and detection.
- The skin color can be used to discriminate the hand region from the other moving objects and/or the background. The color of the skin is measured with the HSV model.
- The HSV (hue, saturation, and value) value of the skin color is 315, 94, and 37, respectively.

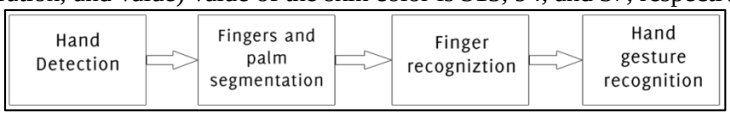

Fig. 2: Implementation Flow

Fig. 3: Background Subtraction

Finally, the image of the detected hand is resized to make the gesture recognition invariant to image scale.

- The image detection process can be shown as:

D. Counting Fingers:

- Counting fingers is another important part of gesture recognition as every gesture will contain a particular number of fingers to carry out a specific action.
- Thus images will be needed to scan the fingers and carry out the action assigned to the gesture.
- These fingers, when scanned and counted will match with the predefined gesture and will be used for carrying out the task.
- In this case, the fingers and the thumb are counted in different ways. Each have a different method for scanning and recognizing the fingers/thumb.

Scanning images will be done through image sensing algorithms like:

E. Finger Counting Algorithm [2]:

- Finger Counting Algorithm implements in a different way of counting the number of fingers.
- Initially the RGB image is converted into binary as seen. To get the desired gesture the image is cropped and zoomed.
- After this when the pixel value changes from 1 to 0 and then 0 to 1 we consider it as one finger.

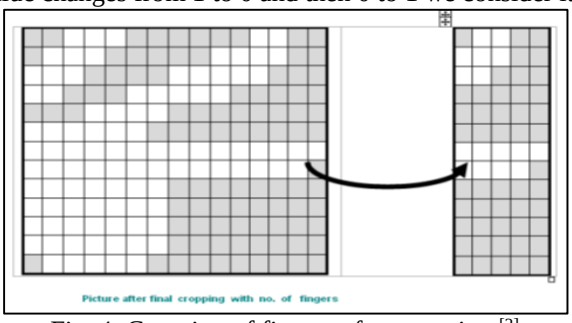

Fig. 4: Counting of fingers after cropping [2]

F. Creating a Binary Picture:

To make all the pre-processing easier, it is better to create a binary picture. To do so, it is necessary to choose a threshold: pixels with value lower than this threshold will be set to 0 (black) and others will be set to 1. The choice of this threshold depends on the web camera properties. Then it is necessary to execute noise-removal functions, else every noisy pixel that its value is too high may be considered as part of the hand and will be included in the zoom-in picture.

Zooming in the Binary Picture Let's suppose that the picture is completely black (0), except in the area of the hand in which it is completely white (1), according to the previous few examples. The problem now is how to crop the area completely black. The fastest method is to compute a line vector in which the i-th element is the sum of the elements of the i-th column of the picture, and a column vector in which the i-th element is the sum of the elements of the i-th line of the picture.

G. Counting through X-Y Axis [2]

- In this algorithm the thumb and fingers are detected separately.
- X-axis is used for detection of thumb and Y-axis is used for detection of fingers.

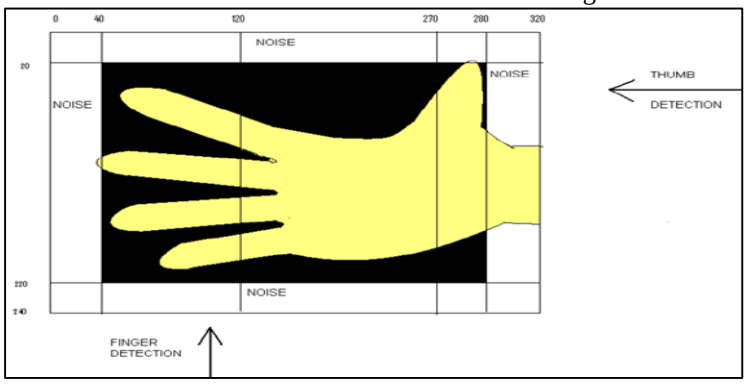

Fig. 5: Implementation of counting fingers with the pixels of 320x240

H. Thumb Detection [2]:

Thumb detection step is performed in order to detect the presence or absence of thumb in hand gesture. Thumb is considering as a significant shape feature to classify various hand gestures in this approach. We know that thumb can either be reside at right most side of all finger of the hand or at left most side of the hand in general . To detect the presence of thumb in hand, we proceed with the previously calculated bounding box and consider the left side and right side of this bounding box. By taking 30 pixels width from each side of the bounding box we crop this bounding box in to two regions, one which is represented by green boundary is left box and another is right box represented by blue boundaries in the image shown below. After having these two boxes we count the total number of white pixels presents in binary image which represent the hand object. Then we count number of white pixels present in each box i.e. green and blue box. If there is less than 7% of total white pixels exist in any of the right box or left box, we consider that thumb is present in that box only. If both boxes having more than 7% percent of total

White pixels in the image, then thumb are not present in any of the box. And if both boxes having less than 7% of total number of white pixels exist in image then thumb is not present in any of the box because thumb is only one and it cannot be detected at both side of the bounding box for the same hand gesture. The percentage of white pixels set as 7% is chosen experimentally on testing more than 400 images. In our previous approach discussed in paper, the parameter was

taken as 0.69%, but for getting accurate result we set the percentage of white pixel for thumb detection as 7%. This method is applicable to both categories of hand. The results will be highly influenced by variation in orientation.

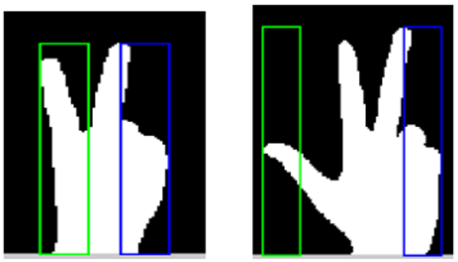

Fig. 6(a) : Fig. 6(b):

Fig. 6: Thumb detection [3]

Fig.6 shows the partition of bounding box in two boxes represented by green and blue box. In Fig. 6(b), It detect thumb at the left hand side and in the green box , in which percentage of white pixels is counted less than 7 % of total white pixels.

I. Matching:

- After the gestures are recorded and fingers are counted, every gesture is scanned for the exact same gesture in the database.
- In this process the predefined gesture is matched with the current recorded gesture and the required action is carried out.
- Algorithms like Point Pattern Matching are used to find a matching gesture in orientation differences, as follows:

1) The points on fingers and boundaries are matched with the database.
2) It is scanned for the perfect fit.
3) Once match is found out, the match is reported to the application.

J. Operations:

- Once a match is found, the software reports it to the OS.
- The code block will specify which gesture will be used to do what action.

The action specified in the database is then performed so as to end the task of the first gesture.

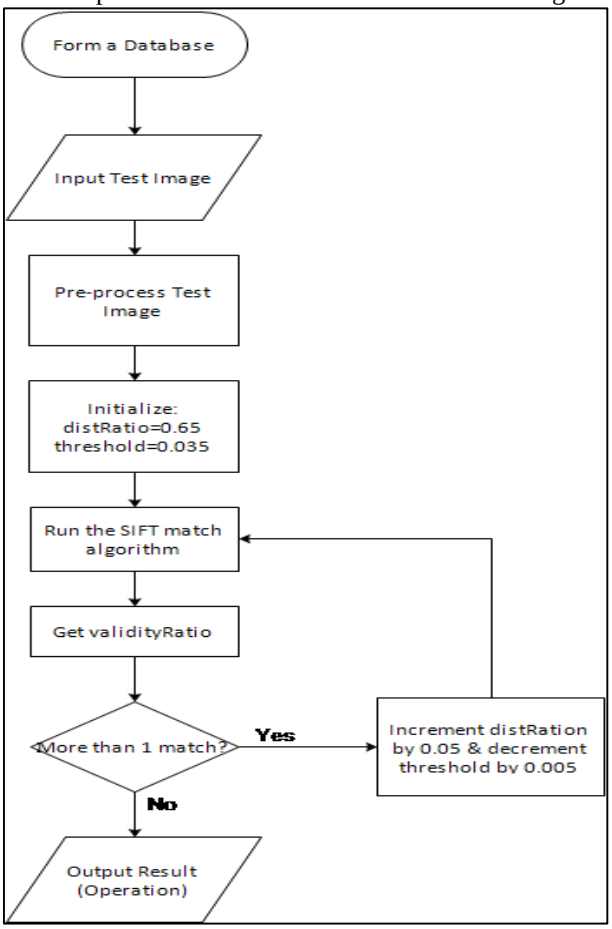

Fig. 7: Matching Algorithm

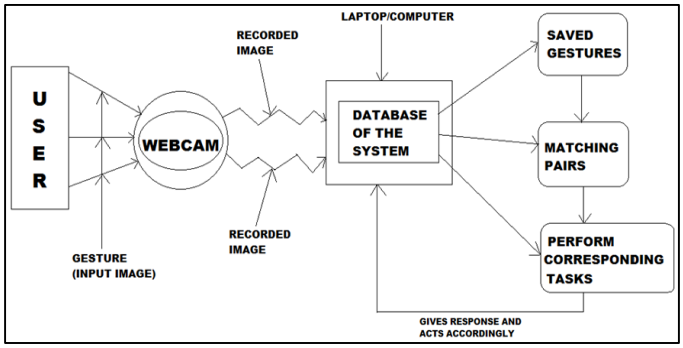

Fig. 8: Process diagram

IV. ARCHITECTURE DIAGRAM

The architecture of our system will be as shown in the figure.

- Firstly the webcam will capture the image of the gesture by the user,
- This image will then go through various stages of background subtraction and skin tone detection.
- Then the image will be sent to the database in which the predefined gestures are stored.
- In the database the gesture recognition will play its part and number of functions will be counted.
- Along with the counting of fingers, orientation detection will take place and the actual gesture will be sent for matching.
- Once the gesture processed gesture is received, the pattern matching algorithm will carry out its function to find the match by finding the pattern which is predefined in the database.
- When the match is found, the report will be sent to the database which in turn will report to the system.
- The system will then carry out the task as specified in the database for that particular gesture.
- Once the action is performed, the task loop of the application is over and it is set to receive the next gesture and perform the task required by the user.

V. CONCLUSION

We have thus implemented Gesture recognition using various algorithms and the best method/algorithm will be used in the further development of the project.

The advantages of this approach include easy detection, simplicity and ease of understanding.

Since this method does not need to be trained, the time usage is reduced thus lowering the complexity of the program. It also eliminates the need for post processing as simple matching of gesture is carried out to find the suitable task and carry it out.

Additionally these algorithms can detect the orientation of the gesture which can prove helpful as similar gesture can be carried out in various orientations to provide different tasks for every gesture.

The advantages of this algorithm thus result in a lower amount of computation time and hence improving the efficiency of the program and the project.

ACKNOWLEDGEMENT

We would like to thank our project guide Prof. Mamta Meena for her major co-operation in our project work. Working under her guidance has proved more than helpful for the development of this project.

We are also thankful to Hon. Shri Sunil Rane Sir, our Principal Dr. S.P. Kallurkar, our Head of Department Prof. Mahendra Patil along with our Project co-ordinator Prof. Deepali Maste and all the staff members of the Computers department who have provided us various facilities and guided us to develop a very good project idea.

We would also like to thank all other staff members to provide us with enormous support and access to college facilities.

And finally we would like to thank our friends to help us keep up our efforts.

REFERENCES

[1] Gesture recognition system Mrs Ayesha Butalia (MIT-College of Engineering), Ms Divya Shah (MIT-College of Engineering), Doctor R.V Dharaskar (GHRCE, Nagpur).
[2] Hand gesture recognition based on shape parameters Meenakshi Panwar Centre for Development of Advanced Computing Noida, Uttar Pradesh, India.
[3] www.ijrcct.org/index.php/ojs/article/view/706
[4] http://in.mathworks.com/matlabcentral/fileexchange/30447-point-pattern-matching-algorithm-for-hand-gesture---american-sign-language--asl--recognition
[5] https://www.hindawi.com/journals/tswj/2014/267872/