**Hand Gesture Recognition Using Depth Camera**

*6th SEMESTER END-SEM MINI PROJECT REPORT*

*FOR THE DEGREE OF*

**BACHELOR OF TECHNOLOGY**

*IN*

**INFORMATION TECHNOLOGY**

*BY*

Ankita Nasipuri (LIT2015012)

Puja Kumari (LIT2015017)

Samriddhi Niranjan (LIT2015021)

Kajol (LIT2015022)

*UNDER THE SUPERVISION OF*

**Prof. Anupam Agrawal**

IIIT-ALLAHABAD

**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,**

**LUCKNOW**

(An Institute Established Under Public Private Partnership (PPP) Model By MHRD, UP Govt. & UPLC)

**MAY, 2018**

**DECLARATION BY THE CANDIDATES**

We hereby declare that the work presented in this end semester project report of B.Tech (IT) 6th Semester titled “**HAND GESTURE RECOGNITION USING DEPTH CAMERA**” submitted by us at Indian Institute of Information Technology, Lucknow is an authenticated record of our original work carried out from January 2018 to May 2018 under the guidance of **Prof. Anupam Agrawal**.

Due acknowledgements have been made in the text to all other material used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Allahabad

Date: 4 May, 2018

Ankita Nasipuri (LIT2015012)

Puja Kumari (LIT2015017)

Samriddhi Niranjan (LIT2015021)

Kajol (LIT2015022)

i

**CERTIFICATE FROM SUPERVISOR**

I do hereby recommend that the mini project report prepared under my supervision by Ankita Nasipuri (LIT2015012), Puja Kumari (LIT2015017), Samriddhi Niranjan (LIT2015021) and Kajol (LIT2015022) titled “Hand Gesture Recognition Using Depth Camera” be accepted in the partial fulfillment of the requirements of VI semester of Bachelor of Technology in Information Technology for Examination.

Date: Prof. Anupam Agrawal

Place: Allahabad IIIT-Allahabad

ii

**ACKNOWLEDGEMENT**

We would like to express our sincere gratitude to the Department of Information Technology, IIIT Allahabad for providing valuable help, discussion, suggestions and resources. We are especially thankful to **Prof. Anupam Agrawal** for his valuable support and helping us figure out approach to the problem.

AnkitaNasipuri

Puja Kumari

SamriddhiNiranjan

Kajol

iii

**Table of Contents**

1. Abstract..............................................................................................1
2. Introduction…………………………….…...………………....…..2
3. Motivation….…………………………………………......………..4
4. Problem Statement, Scope and Objective …………….............……5
5. Literature Review …………………………….…........…………....6
6. Methodology………………….........................................................14
7. Software and Hardware Requirements…………………………….15
8. Implementation ...........…..……………………….......……...…….16
9. Results....................................................................….............………20
10. Activity Time Chart…...……………………………......………….23
11. Conclusion and Future Scope……...................……………………24
12. References…………………………………………….............…….25

iv

**ABSTRACT**

Human Computer Interaction can acquire several advantages with the introduction of different natural forms of device free communication. Gestures are a natural form of actions which we often use in our daily life for interaction, therefore to use it as a communication medium with computers generates a new paradigm of interaction with computers. In this paper, we propose to recognize dynamic hand gestures using Microsoft Kinect and control the VLC media player through those gestures. The user stands in front of Kinect sensor and performs appropriate gestures. The depth information is captured by Kinect. The depth image is used to segment the hand from the background using depth thresholding. The depth video is divided into frames and HoG feature extraction is applied to the extracted frames. To recognize the hand gestures using HoG features, we train a multi-class SVM classifier which will classify hand gestures. Finally, the VLC command corresponding to the predicted gesture will be executed.

1

**1.INTRODUCTION**

Since the last decade of the 20th century many efforts are being made to improve the interaction between human beings and computers. To improve the way in which we interact with computers efforts are being made to make the interaction more and more similar to the way human beings interact with each other. Gestures are an important means of communicating in our day-to-day life [7]. Often we communicate by the movement of body parts like hands and head rather than speaking. So for a successful machine-human interaction consideration of these gestures is inevitable. The scope of application for hand gesture recognition is enlarging over the years and includes sign language recognition [9], handwriting recognition [3] and medical systems [8].

In this project we intend to develop an application for recognizing hand gestures for controlling the VLC media player. The application uses the Microsoft Kinect for acquisition of depth images. The functions which we intend to control are: Play/Pause, Volume up, Volume down, Next video, Previous video, and Quit.

We first plan to capture depth information using Microsoft Kinect. We will then use the depth information for hand segmentation. Then frame by frame feature extraction using HoG [8] will be performed and extracted features will be sent to a multi class SVM [10] for classification. Finally, the recognized gestures will be sent to VLC and intended action will be generated.

**1.1 MICROSOFT KINECT**

The Kinect sensor incorporates several advanced sensing hardware. Most notably, it contains a depth sensor, a colour camera, and a four-microphone array that provides full-body 3D motion capture, facial recognition, and voice recognition capabilities [10]. The depth sensor consists of the IR projector combined with the IR camera, which is a monochrome complementary metal oxide semiconductor (CMOS) sensor. The IR projector is an IR laser that passes through a diffraction grating and turns into a set of IR dots. In Fig. 1.1: Microsoft Kinect and in Fig. 1.2: (a) Depth image provided by Kinect; (b) RGB image provided by Kinect are shown as follows:

2

|  |
| --- |
| https://lh3.googleusercontent.com/3KSrZQFV6O3quRMJBYhWve3BgkJqpeJh9irxIkpos3XFmRpqYmyWQaF2tUWaaN9Naij8-2Y-fQr1yLlwz7h6s7b8j5aUWhQlSC8nL0OpimkGWSk28ahtLFDBn46iMdG5NSbfrSp7oBg |

Fig, 1.1: Microsoft Kinect

|  |
| --- |
| https://lh6.googleusercontent.com/uRsHHAjpNLSRTrcRHbKBWcOmO3Yoy2wRquQkCPiN4GBxAlj-Cn8mYXZecfS2gh_jdjeBilfeQcuwUE4a7PWvlFPEQjP7Re2p0mJnFkC7consTidBguVxAlvekIQlbH6mh6DdcYdZKG4 |

**(a) (b)**

Fig. 1.2: (a) Depth image provided by Kinect; (b) RGB image provided by Kinect

3

**2.MOTIVATION**

Use of a physical controller like mouse, keyboard for human computer interaction hinders natural interface as there is a strong barrier between the user and computer. The project aims to provide the user with a natural and less restricted way to communicate with the computer. The drawbacks faced earlier when using webcams were reduction in accuracy in case of cluttered background and sensitivity to lighting conditions. Kinect sensor provides the depth information of an image which is useful for extracting the hand from the real scene and as a result the unnecessary background information is removed easily. So we intend to use the depth information provided by Kinect to achieve higher accuracy even in cluttered background.

4

**3.PROBLEM OBJECTIVE AND SCOPE**

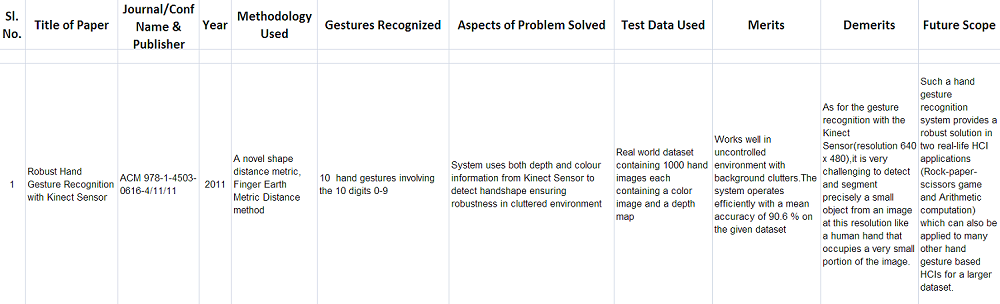
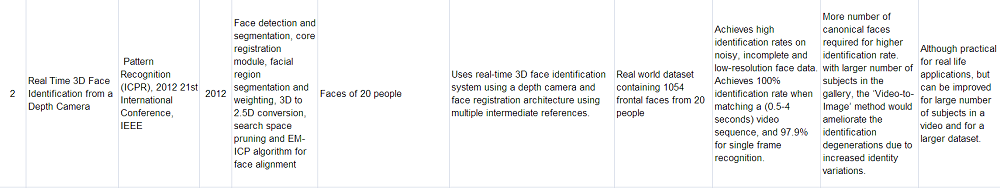
The objective of our project is to develop a software system which takes visual input from Microsoft Kinect depth camera and recognizes dynamic hand gestures for controlling VLC media player application. The input will be the real time video stream of the person performing a particular hand gesture. As output the appropriate commands will be executed on VLC media player.

The gestures and corresponding commands are: Clap - Play/Pause, Palm - Exit, Both hands Up - Volume high, Both hands down - Volume low, Clockwise Rotation - Play next video and Anticlockwise Rotation - Play previous video.

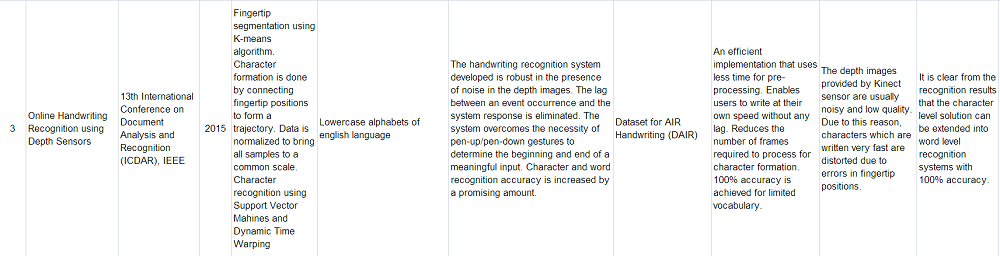
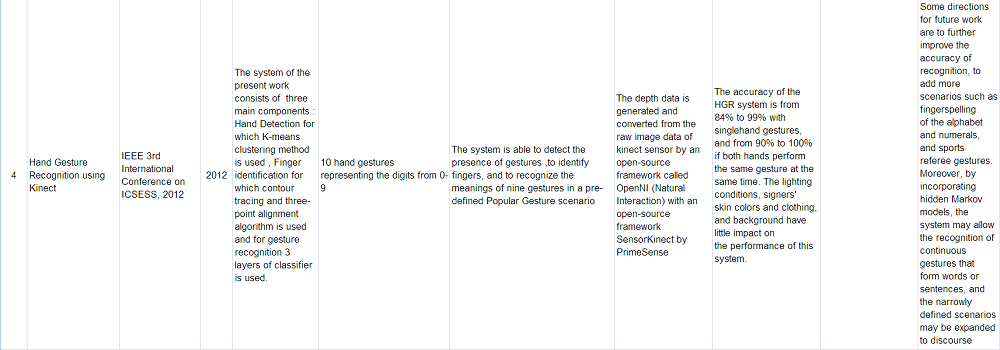
**3.1 Scope:**

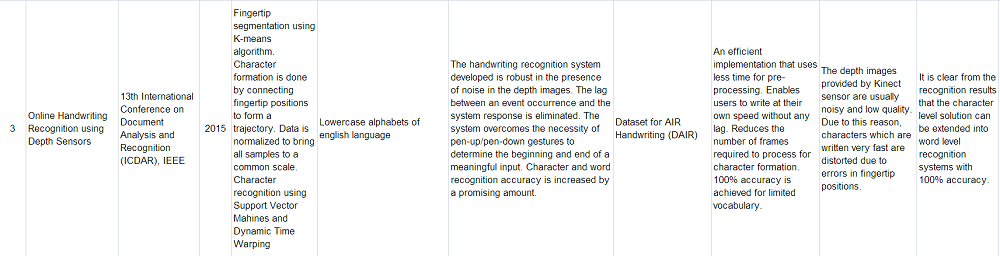
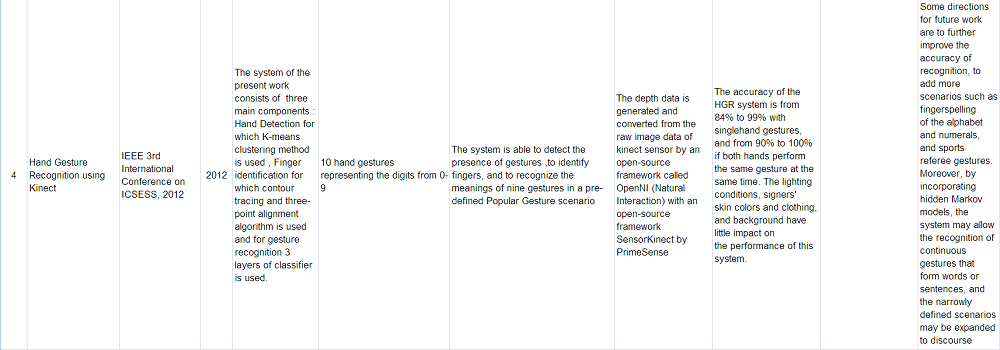
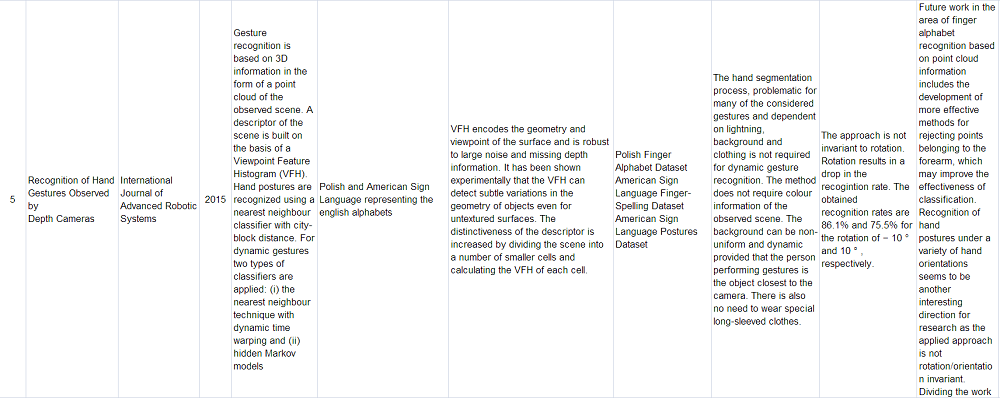
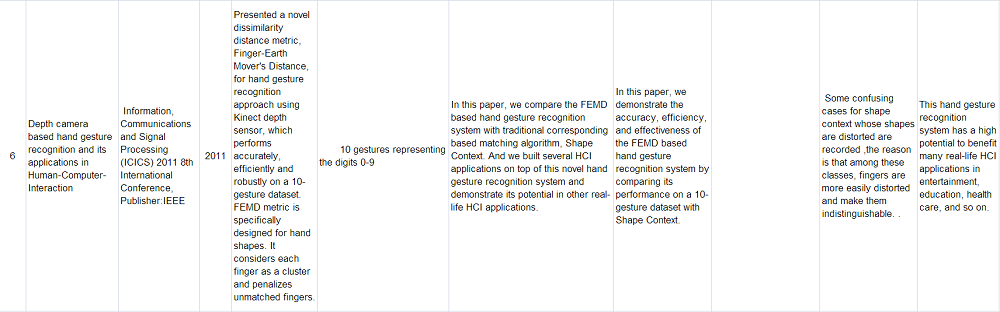
* The software system will take visual input from Microsoft Kinect depth camera.
* The software system should work for dynamic hand gestures.
* Gestures may involve the use of one or both hands.
* Background environment may be static or dynamic and cluttered.
* Gestures will be classified into: Play/Pause, Quit, Volume Up, Volume Down, Next video, Previous video.
* It is assumed that the user will be nearest to the depth camera and facing it. There will be no obstacle between the user and the camera.
* We plan to use HoG for feature extraction and classify gestures using SVM classifier.

5

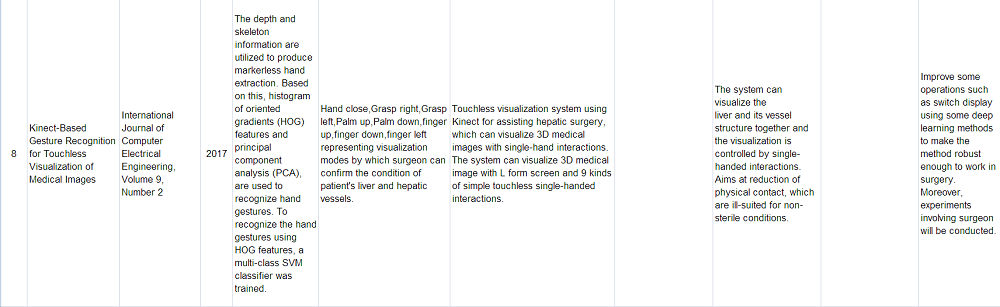
**4.LITERATURE REVIEW**

6

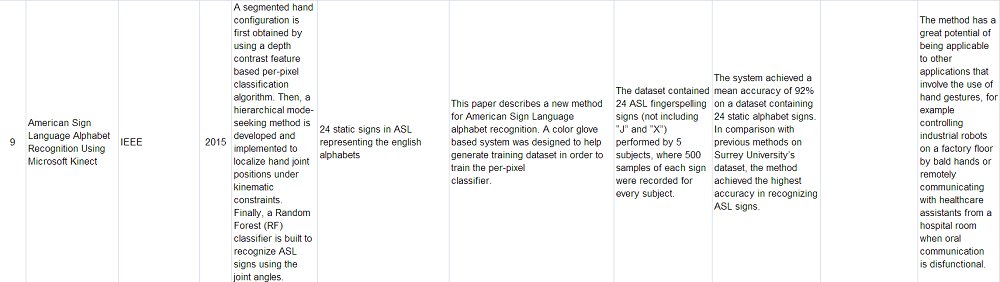
****7

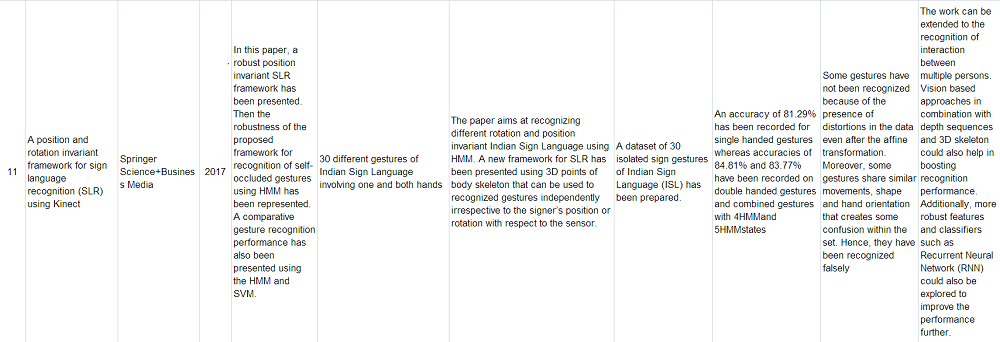
**** 8

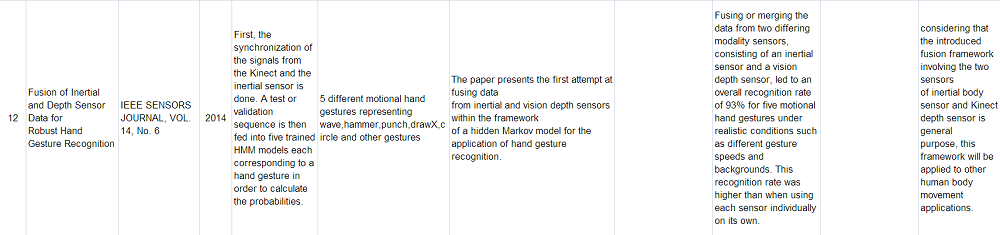
****

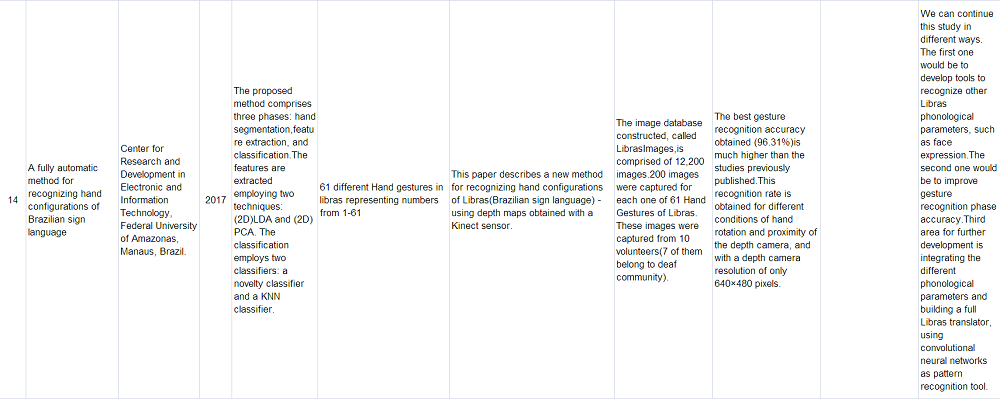
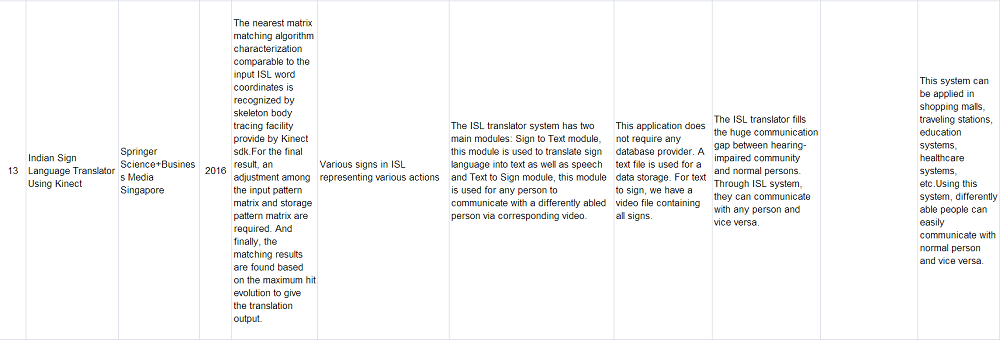


9

 10







12



13

**5.METHODOLOGY**

In this system we have used image preprocessing techniques, feature extraction and classification to recognize the gestures in real time and give the appropriate commands to VLC media player. The phases of our system are described as follows:

* Depth Input: Depth information of the real time video input is obtained from Kinect.
* Hand Segmentation: The hand of the subject is segmented out using depth thresholding.
* Feature Extraction: Histogram of Oriented Gradients (HoG) is used for feature extraction.
* Classification: A multiclass SVM classifier is used to classify the different gestures.
* VLC Interaction: Appropriate command is given to the VLC player according to the recognized gesture.

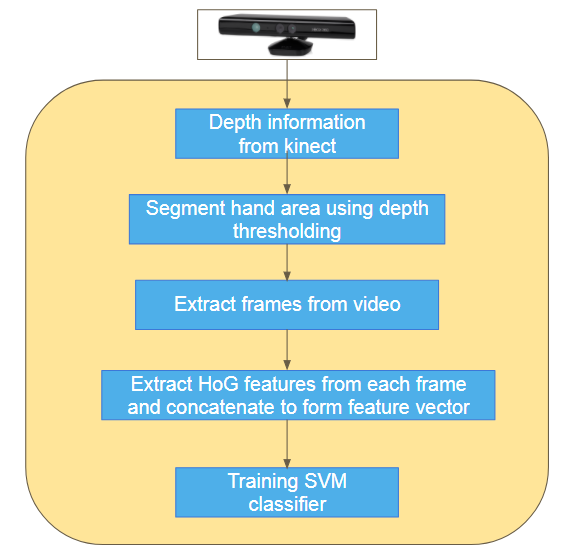


Fig. 5.1 : Flowchart depicting the training phase

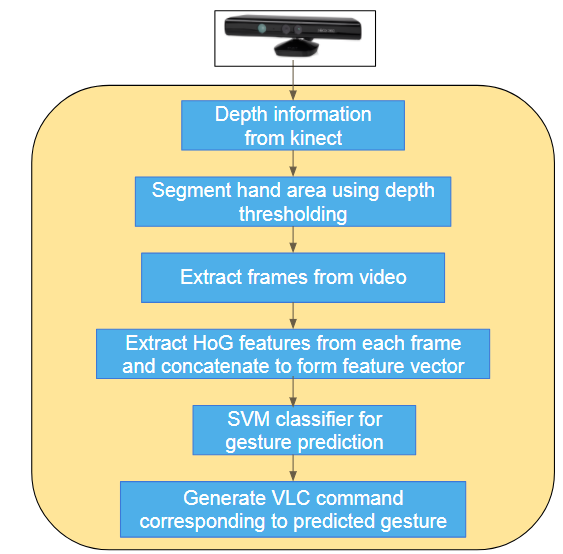


Fig. 5.2 : Flowchart depicting the testing phase

14

**6.SOFTWARE & HARDWARE REQUIREMENTS**

**Implementation Language:** Python 2.7

**Libraries:** NumPy, OpenCV3, Scikit-learn, Freenect, Scipy, SimpleCV.

**Operating System Used:** Ubuntu 16.04LTS.

**Hardware Requirements:** 64-bit environment, Intel Core i5 CPU @ 2.60 GHz

with 4 GB of RAM, Kinect camera(XBOX-360-version-1).

15

**7.IMPLEMENTATION**

In this section we will discuss the hand gesture recognition process in a detailed manner:

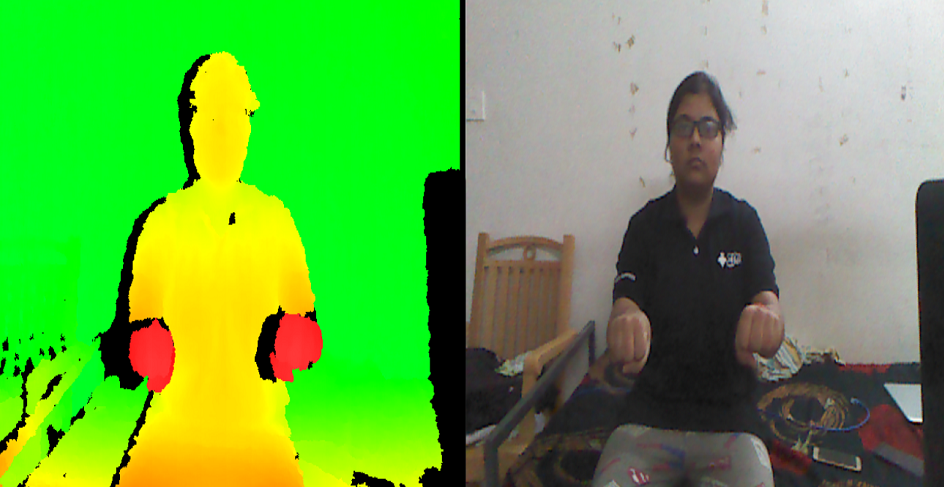
**7.Depth Input:**

The user sits in front of Kinect sensor and performs appropriate gestures. The depth information is captured by Kinect. It is assumed that the user will be nearest to the depth camera and facing it. There will be no obstacle between the user and the camera. Background environment may be static or dynamic and cluttered.

The following figures 7.1 (a)-(f) shows the various gestures performed by the subject.



**(a)**Palm : Quit **(b)**Clap : Play/Pause



**(c)**Both hands down: Decrease Volume **(d)** Both hands up: Increase Volume

16



**(e)**Rotate-Clockwise: Play next video  **(f)**Rotate-Anticlockwise: Play previous video

**7.2 Hand Segmentation**

Hand segmentation was done using depth thresholding. A binary image was created in which only those pixels were marked as 255 whose value was less than the pre specified threshold and the remaining pixels were kept as zero. The thresholded depth image has been used to segment the hand area from the background. The input video was saved as a collection of 20 frames. Given below are the hand detection process images:

|  |
| --- |
| Screenshot from 2018-04-25 03-15-03.png |

(a) (b)

Fig. 7.2 Hand detection process (a) The RGB colour image captured by Kinect Sensor;

(b) The area segmented using depth information.

17

**7.3 Frame by Frame Feature Extraction**

Feature extraction is the process by which certain features of interest within an image are detected and represented for further processing. It is the transition from pictorial to non-pictorial data representation. We have used HoG (Histogram of Oriented Gradients) for feature extraction. This feature descriptor is used for the purpose of object detection. The HOG descriptor technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI). The HoG are extracted from each of the 20 captured frames and concatenated to form the feature vector. Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.
2. Discretize each cell into angular bins according to the gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.
4. Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
5. Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

The following figure demonstrates the algorithm implementation scheme:

|  |
| --- |
|  |

Fig. 7.3: HoG Implementation Representation

18

**7.4 Classification**

To recognize the hand gestures using HOG features, we train a multi-class SVM classifier which will classify hand gestures into six gestures:

* Clapping
* Both hands up
* Both hands down
* Clockwise Rotation
* Anti-Clockwise Rotation
* Palm

**7.5 VLC Interaction**

The classification phase gives us the predicted label for the gesture performed. The labels are 'BU' for both hands up, 'C' for clap, 'BD' for both hands down, 'P' for palm, 'RA' for anticlockwise rotation and 'RC' for clockwise rotation. After getting the predicted label we use vlc-ctrl commands for generating the corresponding action.

The following steps are to be followed for using our application: First, the subject(user) needs to sit in front of the kinect camera within the range of depth sensor of kinect which is 800mm to 4000mm. No obstacle should be present between the user and the camera and also the hand should be the nearest object to the camera. As soon as the program starts the user can perform gestures at any time. As the gesture is completed and the desired outcome is received the user can perform any other gesture according to their needs. To avoid wrong hand detection user must bring his hand in front of the camera only when he or she desires to perform a gesture otherwise wrong gesture interpretation will take place.

19

**8.RESULTS**

For batch testing, video frames of all 6 gestures of 8 people were collected. The average accuracy of each gesture is calculated and shown in Fig. 8.1. The confusion matrix in Fig. 8.3 shows the result of our experiment.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Gestures** | **Clap** | **Rotate**  **Anti** | **Rotate**  **Clock** | **Both**  **Up** | **Both Down** | **Palm** |
| **Accuracy(%)** | 87.5 | 100 | 100 | 87.5 | 100 | 100 |

Fig. 8.1: Table showing accuracy for each gesture

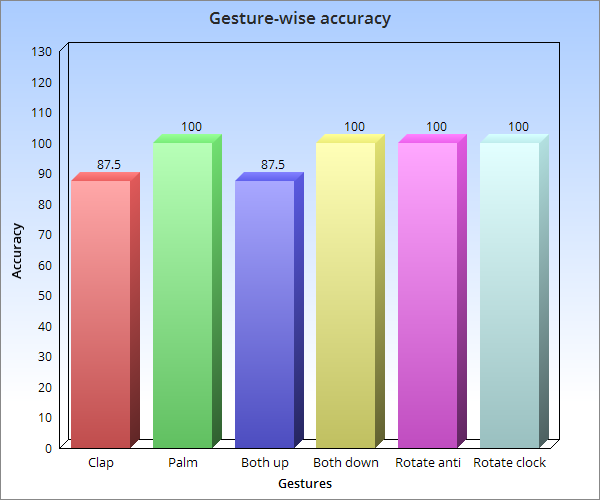


Fig. 8.2: Bar chart showing gesture wise accuracy

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Clap** | **Rotate**  **Anti** | **Rotate**  **Clock** | **Both**  **Up** | **Both Down** | **Palm** |
| **Clap** | 87.5 | 0 | 0 | 0 | 12.5 | 0 |
| **Rotate Anti** | 0 | 100 | 0 | 0 | 0 | 0 |
| **Rotate Clock** | 0 | 0 | 100 | 0 | 0 | 0 |
| **Both**  **Up** | 12.5 | 0 | 0 | 87.5 | 0 | 0 |
| **Both Down** | 0 | 0 | 0 | 0 | 100 | 0 |
| **Palm** | 0 | 0 | 0 | 0 | 0 | 100 |

Fig. 8.3: Confusion matrix of gesture recognition

20

Due to similarity between some frames of gestures 'Clap' and 'Both up', the confusion is higher for these gestures. Otherwise, high accuracy has been achieved for the remaining gestures.

The proposed methodology gives improved accuracy as compared to other works based on depth camera. In [16], skeleton tracking is used and gestures were recognized using HMM. The algorithm was tested on ISL 20 dynamic signs and achieved 89.25% average accuracy. In [17], contour tracking and dynamic time warping were used to recognize gestures. An average recognition rate of 92.4% was achieved over 55 static and dynamic gestures. Our methodology, which uses depth thresholding and SVM for gesture recognition achieved average accuracy of 95.83%. Fig. 8.4 is the corresponding bar chart for these results.

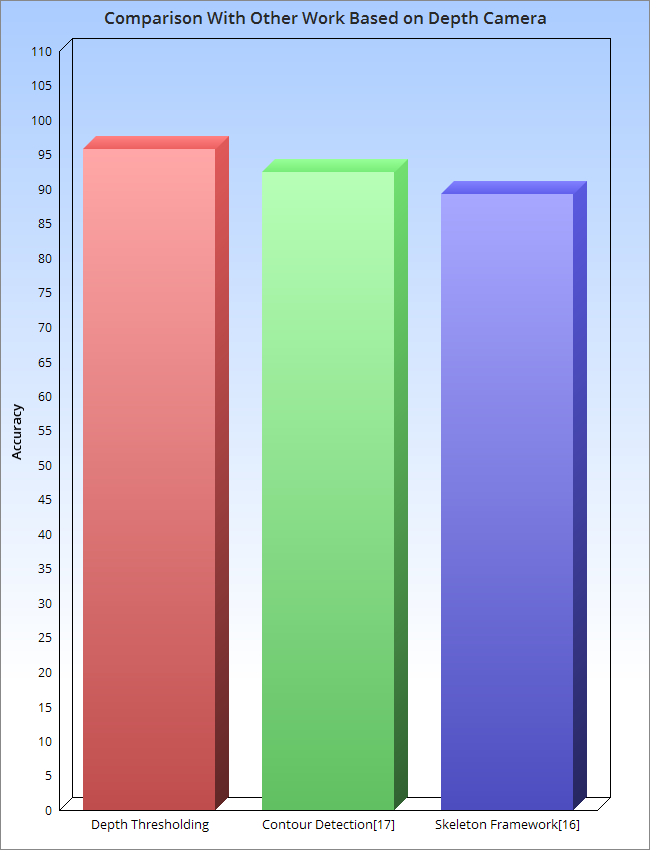


Fig. 8.4: Bar chart showing accuracy comparison with other works based on depth camera

Our proposed methodology also gives better gesture recognition rate than when only RGB information is used. [18] achieved average accuracy of 92.28% in plain background. However the accuracy dropped to 64.85% in case of non-plain background. In cases where the background was not plain, the objects in the background proved to be inconsistencies to the image capture process, resulting in faulty outputs. Our methodology overcomes this drawback by using the depth information provided by kinect. The fig. 8.5 shows the corresponding accuracy comparison.

21

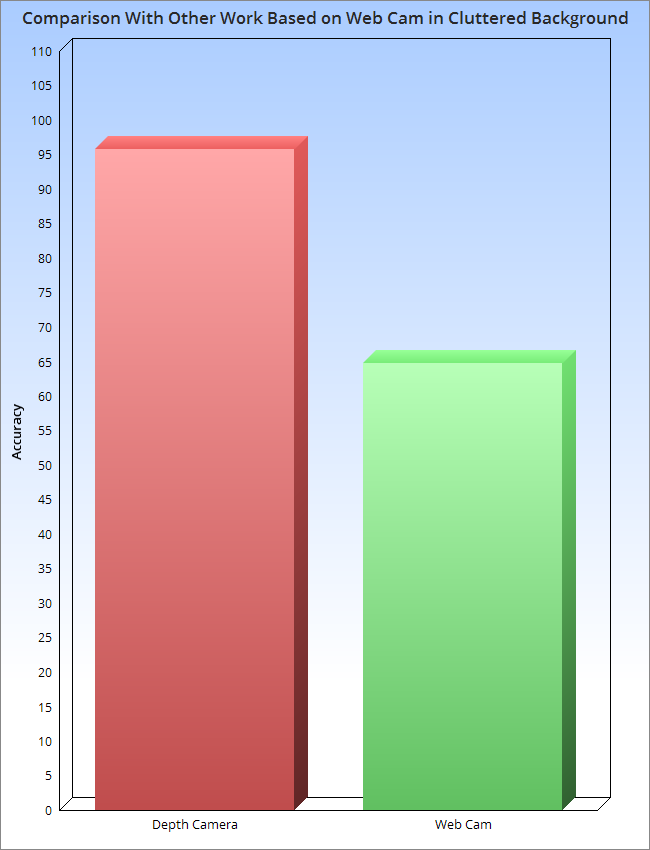
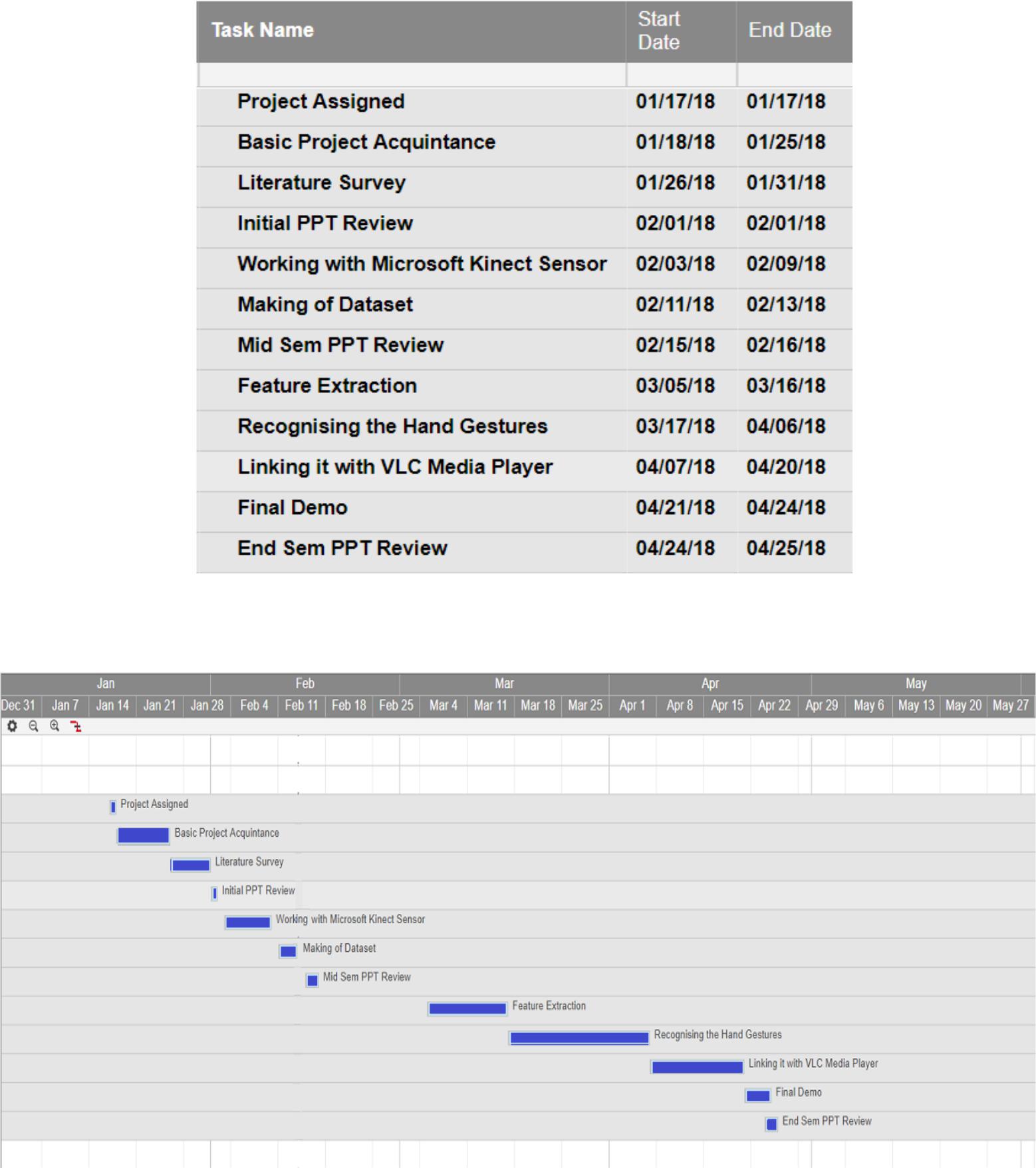


Fig. 8.5: Bar chart showing accuracy comparison with work based on webcam

22

**9.ACTIVITY TIME CHART AND GANTT CHART**



**10.CONCLUSION AND FUTURE SCOPE**

The aim of this project was to develop a real time system for controlling VLC media player using dynamic hand gestures. For this purpose we have made use of Kinect(XBOX-360-Version-1) Camera. We have presented an application by which the user can control the functions of VLC media player from a distance without using mouse and keyboard. The major motivation behind the work was to decrease the use of a physical controller like mouse, keyboard for man-machine interaction which become a barrier between user and the computer in daily life. The software is invariant to lighting conditions and background cluttering ,it doesn't depend on skin color and size of hands.

The main hardware requirement Kinect(XBOX-360-Version-1)Camera is an expensive equipment, not available to the common people. At present our application works for VLC media player but in the future it can be generalized to work for any media player. Furthermore the accuracy obtained by the implemented method can be enhanced and the system tends to be more robust to variations in hand gestures, performing styles and speed. The work has the potential to be applied in other HCI applications that adopt hand gestures as the interface.

24

|  |
| --- |
|  |

**11.REFERENCES**

[1] Zhou Ren, Junsong Yuan, and Zhengyou Zhang. 2011. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In Proceedings of the 19th ACM international conference on Multimedia (MM '11). ACM, New York, NY, USA, 1093-1096.

[2]R.Min, J. Choi, G. Medioni and J. L. Dugelay, "Real-time 3D face identification from a depth camera," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Tsukuba, 2012, pp. 1739-1742.

[3]R.Aggarwal, S. Swetha, A. M. Namboodiri, J. Sivaswamy and C. V. Jawahar, "Online handwriting recognition using depth sensors," *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, Tunis, 2015, pp. 1061-1065.

[4] Yi Li, "Hand gesture recognition using Kinect," *2012 IEEE International Conference on Computer Science and Automation Engineering*, Beijing, 2012, pp. 196-199.

[5] T. Kapuscinski, M. Oszust and M. Wysocki, "Recognition of signed dynamic expressions observed by ToF camera," *2013 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Poznan, 2013, pp. 291-296.

[6] Z. Ren, J. Meng and J. Yuan, "Depth camera based hand gesture recognition and its applications in Human-Computer-Interaction," *2011 8th International Conference on Information, Communications & Signal Processing*, Singapore, 2011, pp. 1-5.

[7]Y. Chen, B. Luo, Y. L. Chen, G. Liang and X. Wu, "A real-time dynamic hand gesture recognition system using kinect sensor," *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Zhuhai, 2015, pp. 2026-2030.

[8]Jiaqing Liu, Ryoma Fujii, Tomoko Tateyama, Yutaro Iwamoto, Yenwei Chen. Kinect-Based Gesture Recognition for Touchless Visualization of Medical Images. International Journal of Computer Electrical Engineering, Volume 9, Number 2.

[9]Cao Dong, M. C. Leu and Z. Yin, "American Sign Language alphabet recognition using Microsoft Kinect," *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Boston, MA, 2015, pp. 44-52.

[10]K. K. Biswas and S. K. Basu, "Gesture recognition using Microsoft Kinect®," *The 5th International Conference on Automation, Robotics and Applications*, Wellington, 2011, pp. 100-103.

[11]R. Saini, P. Kumar, P. P. Roy and D. P. Dogra, "An efficient approach for trajectory classification using FCM and SVM," *2017 IEEE Region 10 Symposium (TENSYMP)*, Cochin, 2017, pp. 1-4.

25

[12] K. Liu, C. Chen, R. Jafari and N. Kehtarnavaz, "Fusion of Inertial and Depth Sensor Data for Robust Hand Gesture Recognition," in *IEEE Sensors Journal*, vol. 14, no. 6, pp. 1898-1903, June 2014.

[13]Pratik H. Suvagiya, Chintan M. Bhatt and Ritesh P. Patel. Indian Sign Language Translator Using Kinect. Springer Science + Business Media Singapore.

[14] Costa Filho, Cicero Ferreira Fernandes, Souza, Robson Silva de, Santos, Jonilson Roque dos, Santos, Bárbara Lobato dos, & Costa, Marly Guimarães Fernandes. (2017). A fully automatic method for recognizing hand configurations of Brazilian sign language. *Research on Biomedical Engineering*, *33*(1), 78-89.

[15] L. Rioux-Maldague and P. Giguère, "Sign Language Fingerspelling Classification from Depth and Color Images Using a Deep Belief Network," *2014 Canadian Conference on Computer and Robot Vision*, Montreal, QC, 2014, pp. 92-97.

[16]Ghotkar, Archana & Vidap, Pujashree & Deo, Kshitish. (2016). Dynamic Hand Gesture Recognition using Hidden Markov Model by Microsoft Kinect Sensor. International Journal of Computer Applications. 150. 5-9. 10.5120/ijca2016911498.

[17] G. Plouffe and A. M. Cretu, "Static and Dynamic Hand Gesture Recognition in Depth Data Using Dynamic Time Warping," in IEEE Transactions on Instrumentation and Measurement, vol. 65, no. 2, pp. 305-316, Feb. 2016

[18]Aashni Haria, Archanasri Subramanian, Nivedhitha Asokkumar, Shristi Poddar, Jyothi S Nayak, "Hand Gesture Recognition for Human Computer Interaction", *Procedia Computer Science, Volume 115,2017,* *Pages 367-374,ISSN 1877-0509.*

26

**COMMENTS AND SUGGESTIONS**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**