INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT

(Volume3, Issue2)

Available online at www.ijarnd.com

Real-Time Hybrid Face and Gesture Detection System Using Open CV

Sagar Srivastava[1], Shivam Sharma[2], Paras Jain[3], Rajeev Singh[4]

[1234]*Student, IMS Engineering College, Ghaziabad, U.P*

ABSTRACT

Gesture-based interaction systems are becoming more and more popular both at workplace and home. The projects intend to develop a system that can recognize hand gestures which can be used as an input command to interact with the PC which can be applied to picture gallery browsing. One of the key areas that need to be looked at while developing such systems is the image processing stage. Since it would be very hard to produce an algorithm that can recognize gestures in the time allocated, within my project, I plan to design and implement a system that can perform general image processing of the user image captured in real time. Most of my work will be based on image processing techniques. My expected outcome will be an algorithm that can detect skin regions of the image user image captured and detect contours around the detected skin regions. In order to manage my project, I will be using Waterfall model as the system development methodology. Microsoft Visual C++ will be used for the implementation of the code developed in combination with the OpenCV library. I feel that if I successfully meet my targets then I will have contributed towards the future of natural gesture-based interfaces, if only in a minimal way.

Keywords: *Face Detection, Open CV, Image Processing, Microsoft Visual C++.*

1. INTRODUCTION

Human-computer interaction (HCI) is the study, planning, and design of the interaction between users (people) and computers. As an alternative to the traditional human-computer interaction interface (HCI) such as the keyboard and the mouse, the use of human movements, involving the face, the whole body and specially hand gestures has attracted more and more people in recent years. A computer that can recognize and respond to the user's gestures could provide a natural interface. The diverse logical and physical capabilities of users (e.g. elderly, children or people with disabilities) also require human-computer interfaces that are easily learnable and usable, instead of traditional interaction techniques such as the mouse and keyboard, which require a certain kind of skills, and restrict the user in a certain kind of physical mode. One of the latest developments made in the field of gaming using such interaction technique is the Microsoft's Kinect for the Xbox 360 console. It enables users to control and interact with the Xbox 360 without the need to touch a game controller like remote, such gesture-based have become extremely popular among today's community. Hands are one of the most multipurpose tools in our human body to accomplish different tasks. They are one of the important features in our user interfaces and interaction applications. Interest in the field of computer vision based hand gesture recognition has increased in recent times due to its potential application in the field of Human-Computer Interaction. The most important feature of this technique is that the system uses hand gesture recognition as an input, through which the users can control the system or devices without having to touch any external interaction devices such as mouse or keyboard. This serves as a motivating factor to carry out this project. This project aims to develop a system, which captures certain hand gestures as an input from the users using a web camera and performs the task associated with the gesture recognized. The recognized will be applied in gallery browsing of the PC being used. A program based on

An open source library, which looks into image processing, and hand gesture recognition developed to accomplish the aim of this project. Research aim and objectives Aim The aim of this project is to develop and implement a set of an algorithm that utilizes gesture recognition to browse through the picture gallery of a PC using a web camera for the purpose of making the PC more usable and improve the interactions between the users and the computers. *Redefined Aim Due to the unexpected complexity of the problem, this project will look into developing and implementing a set of algorithms that enables skin and contour detection of the user's hand in real time. Objectives In order to fulfill the aims set for this project, the following

objectives must be followed: 1. Begin literature review on areas like gesturing recognition, image processing, and open source libraries. 2. Determine the appropriate methodology to be used to design and implement the proposed system based on a detailed analysis of each methodology. 3. Formulate requirements based on techniques chosen from the literature found during research. 4. Use analyzed requirements to begin designing the steps involved in image processing. 5. Implement the processes of skin detection and contour detection using a suitable programming language and an appropriate open source library. 6. Assess the developed code, by applying various testing techniques to ensure that the test cases developed for the software conform to the requirements specification. 7. Evaluate the implemented code and justify whether the derived results have achieved the project aims and objectives. After gaining some knowledge about the topic, the appropriate software development methodology will be chosen to plan the progress of the project. The information through literature review will be analyzed to gather the requirements of the proposed system based on which the design of the system will commence. This system will then be implemented and tested to ensure there are no faults. The project will then be evaluated against the aims and objectives of the project stated.

2. LITERATURE REVIEW

This chapter focuses on research, covering relevant literature relating to this project on touch less hand gesture based human-computer interaction. Appropriate journals, books, Internet sites will be used to gather the relevant literature. This chapter describes the literature associated with gesture recognition and image processing. It discusses the different steps that comprise image processing and the techniques used to accomplish the stages.

The literature also involves previous studies that aimed at the development of similar systems. The techniques used previously are analyzed and appropriate ones chose to devise a feasible solution.

2.1Human-Computer Interaction Human-computer interaction (HCI) is the study, planning, and design of the interaction between people and computers. The aim of HCI is to improve the interactions between computers and users by making computers more usable and more responsive to user's needs.

2.1.1Interaction methods: There are different interaction methods with which we can interact with the computer the most common being the use of a mouse. The mouse was developed at Stanford Research Laboratory (now SRI) in 1965 as part of the NLS project (funding from ARPA, NASA, and Rome ADC) [9] to be a cheap replacement for light pens, which had been used at least since 1954 [10, p. 68]. The mouse was then made famous as a practical input device by Xerox PARC in the 1970's. Another interaction method that has been increasingly popular in the recent years is using gestures for interacting with the computer. Instead of learning completely different new ways to interact, the users may prefer to adopt the natural ways of communication that they are familiar with in everyday life.

These demands have resulted in research in which the user interfaces take advantages of the natural ways that people interact with each other and the physical world, e.g. speech, gesture, eye-gaze, and physical tools (Grasso et al (1998); Oviatt (2001); Wang et al (2001)). Such systems accept gestures as an input form the user recognize the inputted gestures and perform the task associated with that gesture. This project will look into the gesture-based interaction in real time.

2.2 Gesture Recognition Gestures can be described as different types of human movements. These can be two dimensional or three-dimensional and can be specific to the hand, arm or body movements as well as facial expressions. (Hoffman et al (2004)). Gesture recognition enables humans to interface with the machine and interacts naturally with any external devices such as the keyboard. It is a method of assigning commands to the computer (machine) to perform specific tasks. This project will be focusing specifically on hand gesture, as they are easier to perform and recognize with less effort. Also, the users of the software are going to be a mixed crowd so it might be difficult for some people to perform elaborate gestures.

2.2.1 Hand gesture recognition A hand gestures us a sequence of hand postures connected by continuous hand or finger movements over a short period of time. Hand gestures provide a separate complementary modality to speech for expressing one's ideas. So, hand gestures recognizing system can be a natural way of communicating between the computer and humans. There are basically two approaches to hand gesture recognition; Vision based and Nonvision based approaches. Non- vision-based approach uses sensor devices like data glove as shown in figure 2.1 below. The extra sensors make it easy to collect hand location and movement.

2.3 Image processing "Images are stored as a collection of pixels." [4] Color Images consists of a red, green and blue value, which is combined to allow colors to be represented. Grayscale images are different, however; "as pixels are represented by a single number ranging from 0 to 255, where 0 is very black and 255 is very white." [4] Image processing in computing is used to extract useful information form images to perform some specific tasks. Image segmentation, which involves image conversion between different color spaces to minimize the complexity of the image. Skin detection, which gets rid of any unwanted background objects and noises associated with the image. Contour detection, to locate an object in the image. Each of these stages will be discussed in detail below.

2.3.1 Contour Detection The term contour can be defined as an outline or a boundary of an object. Therefore, contour detection deals with detecting various objects in an image.Use of contour detection in image processing is to

locate objects and their boundaries in Images. Also, the output of contour detection shows only the prominent region boundaries leaving behind unwanted edges in the image. Hence, detection of specific objects in the image is only possible through contours.

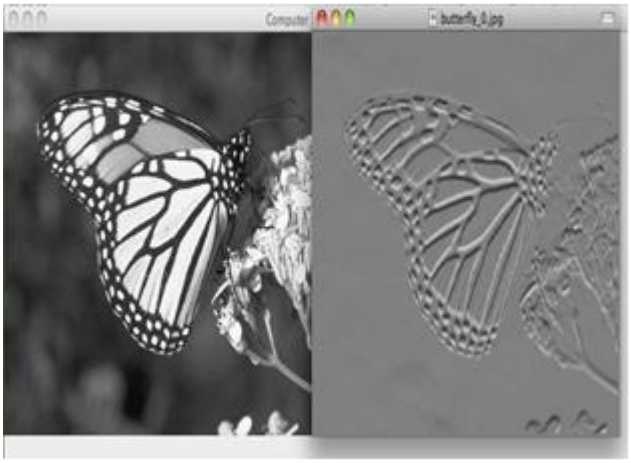

Figure1: Sender's View of Steganography Techniques

3. PROPOSED DETECTION TECHNIQUE

3.1 OpenCV Functions Used

cvGetSize(): used to call the CvSize structure.This gives us the size of the existing structure image.
IplImage() : IplImage is an OpenCV construct.OpenCV uses this structure to handle all kinds of images.
cvCreateImage(): create a new image to hold the changes made to the original image.
cvCvtColor(): converts from one color space to another expecting the data type to be the same. CV_BGR2HSV : conversion code to convert BGR image to HSV(Hue Saturation Value) color space.
cvNamedWindow(): opens a window on the screen that can contain and display an image.
CvScalar (): this function takes one, two,three or four arguments and assigns those arguments to the corresponding elements of val[].It represents the RGB values.
cvInRangeS (): this function is used to check if the pixels in an image fall within a particular specified range.
cvSmooth() : used to reduce noise or camera artifacts.
cvFindContours(): computes contours from binary images.
cvDrawContours() : drawing a contour on the screen. 5.3 Initialization Since this project deals with real time image captured from the web camera, we used the high GUI portion of the OpenCV library that deals with input/output routines and functions for storing and loading videos and images. We used the OpenCV function called cvCreateCameraCapture(). It initializes capturing video from camera.

```
#include "highgui.h"
int main()

  {

  CvCapture* capture = cvCaptureFromCAM(0);


  if(!cvQueryFrame(capture)){
  cout<<"Video capture failed, please
  check the
  camera."<<endl;}else{cout<<"Video
  camera capture status: OK"<<endl;};
  }

  cvReleaseCapture( &capture);

  }
```

Image Processing As seen in section 2.3 literature review, the image processing phase involves sub stages; image conversion, skin detection, and contour detection. Below, we will look at how each stage was implemented:

```
int main()

 {

    int c = 0;

   CvSize sz = cvGetSize(cvQueryFrame( capture));

   IplImage* src = cvCreateImage( size, 8, 3 );

   IplImage* hsv = cvCreateImage( size, 8, 3);

while( c != 27)

  {

   src = cvQueryFrame( capture);
   cvCvtColor(src, hsv,
   CV_BGR2HSV);
   cvNamedWindow( "src",1);
   cvNamedWindow( "img",1);
   cvShowImage( "img", hsv);
   cvShowImage( "src", src);

  }}
```

.The first argument gets the size for the image captured by the camera, the second argument indicates the number of channels , which in our case is 3 and the last argument indicates the bits available in each channel, in our case its 8 bits per channel.

Similarly, another image called "hsv" is created to hold the image after its converted into HSV color space with the exact same arguments.

Next, the function cvCvtColor converts the original RGB image stored in the variable "src" into HSV color space and hold the output HSV image in the variable "hsv" using the conversion code CV_BGR2HSV.

Lastly, it will create a window named "src" and "hsv" using the function.

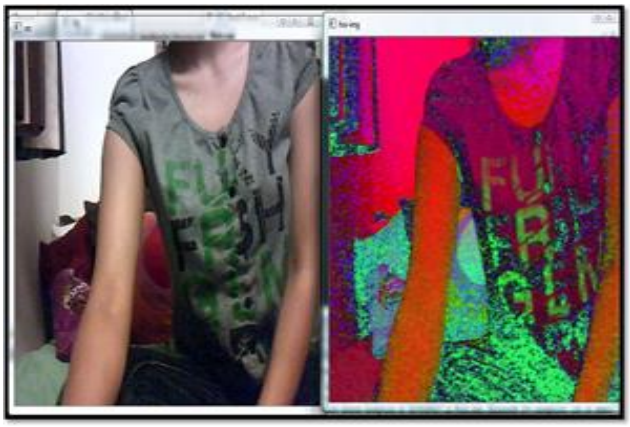

Figure 2: Shows the RGB Image (left) which is converted into the HSV Color Space (right)

4. CONCLUSION

This chapter provides a recap of the key points of the dissertation as well as a brief overview of the research contributions made, the scope for improvements in future research and development of the system evolved from this study and personal reflections based on the overall project to emphasize its achievements, limitations, and scope. 7.1 Summary of the dissertation

Chapter1: Introduction This chapter provides an introduction to the project topic that is gesture recognition and image processing and stating the aim of the project to develop software that performs basic image processing and hand gesture recognition. Also, discussed the research approaches and the project scope of the project. I slightly decided to change the project scope and concentrate solely on image processing.

Chapter 2: Literature Review This chapter looked into the literature related to gesture recognition and image processing in order to gain an in-depth knowledge of the topic. The literature was then analyzed and various past projects reviewed in order to identify the key techniques to be used in the project. The requirements needed in order to design the system was gathered after analyzing the different techniques.

Chapter 3: Methodology This chapter discussed the Software Development Life Cycle (SDLC) methods that could be considered for this project; namely the Waterfall model and the Spiral model. Based on analysis of each, the Waterfall model was chosen, as it was most suitable for this project. The research methodology involved analyzing the literature and the techniques used by researched and any past projects that looked into the similar problem. Finally, Visual C++ was chosen as the platform to implement the software that also incorporated OpenCV, an open source library containing functions that specialize in image processing and gesture.

5. FUTURE ASPECTS

The popularity of gesture-based systems have grown over the years and I believe that it will continue to expand and become an essential part of every day like as the technology advances. This project has expanded my knowledge about the different techniques used in the development of such systems and its useful application in different fields.

My system is now capable of performing basic image processing like conversion between two different color spaces, differentiates between skin pixels and non-skin pixels in the image, performs background subtraction and cancels out major noises associated with the image.

However, there are limitations to software functions and may not always produce expected outputs. One of the major limitations is that I have only tested my system in a few environments therefore low or varying lighting conditions may affect the outputs produced. Also, a more complex image filtering should be applied to the image as it still contains some amount of noise associated with the final image. In regards to further development,

I feel that the project can be expanded further if the minor environmental and filter issues can be resolved. In order to improve the project further, research and further development of the filters used should be critically examined.

I feel that once these issues have been further implemented, the algorithms developed can be used as a basis for future development. Functions for feature extraction, hand tracking, and gesture recognition can be added to make the system usable. Right now the project just looks into image processing but after adding further functions to the current algorithm, the system can be used to assist disabled and elderly to interact with the computer with ease. The project, therefore, has immense scope in terms of the functionality that can be changed to gain better results. The gesture-based system can also be used in the gaming field to provide the gamers with an ultimate touch less gaming experience. Image processing can be used in robotics that can be used to further develop an aid to help elderly autonomously commute in a building/house, which will help immensely.

6. REFERENCES

[1]. Ambler,W.S (2003-2010) UML 2 Activity Diagrams [WWW] Agile Modelling.Available from: http://www.agilemodeling.com/artifacts/activityDiagram.htm [Accessed: 10/03/2011].

[2]. Baudel, T. and Beaudoin-Lafon, M. (1993). Charade: remote control of objects using free-hand gestures. Communications of the ACM, 36(7). p. 28-35.

[3]. Bolt, R. (1980). Put-that-there: Voice and gesture at the graphics interface. ACM SIGGRAPH Computer Graphics, 14(3). p. 262-270.

[4]. Boehm, B. "A Spiral Model of Software Development and Enhancement", ACM SIGSOFT Software Engineering Notes", "ACM", 11(4): 14-24, August 1986.

[5]. Bradski, G. and Kaehler, A. (2008) Learning OpenCV.1st ed.United States.O'Relly Media.

[6]. Creek Photo (2011) Introduction to Color Spaces [WWW]. Available from: http://www.drycreekphoto.com/Learn/color_spaces.htm [Accessed: 26/01/2011]

[7]. English, W.K., Engelbart, D.C., and Berman, M.L., "Display Selection Techniques for Text Manipulation." IEEE Transactions on Human Factors in Electronics, 1967. HFE-8(1)

[8]. Feris, R.,Turk, M.,Raskar, R., Tan, K. and Ohashi, G. "
Recognition of isolated Fingerspelling Gestures Using Depth Edges, "in Real-Time Vision For HumanComputer Interaction. Kisacanin.V. Pavlovic. And T.S. Huang. Eds: Springer, p.43-56, 2005. FYP Task 3 Project Report Student ID: 0822577 47

[9]. Grasso, M., Ebert, D., and Finin, T. (1998) The integrality of speech in multimodal interfaces. ACM CH 1998 Conference on Human Factors in Computing Systems. p. 303-325

[10]. Goldberg, A., ed. A History of Personal Workstations. 1988, Addison-Wesley Publishing Company: New York, NY. 537.

[11]. Hashem, H.F, "Adaptive technique for human face detection using HSV color space and neural.networks," Radio Science Conference, 2009. NRSC 2009. National, vol., no., pp.1-7, 2009.

[12]. Hardenberg, C. V, and Berard, F. (2001) Bare-Hand Human-Computer Interaction Proceedings of the ACM Workshop on Perceptive User Interfaces

[13]. Hofmann, F., Heyer, P. and Hommel, G. Velocity Profile Based Recognition of Dynamic Gestures with Discrete Hidden Markov Models. Proc. of the International Gesture Workshop on Gesture and Sign Language in HumanComputer Interaction, Springer London (2004), 81–95.

[14]. Image processing (2007) [WWW] Available from: http://pyrorobotics.org/?page=Introduction_20to_20Computer_20Vision [Accessed 28/02/2011]

[15]. Kovac,P., Peer and Solina,F. (2003). Human Skin Color Clustering for Face Detection. EUROCON2003

[16]. Lahari.T and Aneesha.V. Contour Detection in Computer Vision [WWW] V.Siddhartha EngineeringCollege. Available from: http://www.scribd.com/Contour-Detection-My-Ppt-1/d/41564107 [Accessed: 01/03/2011] FYP Task 3 Project Report Student ID:0822577 48

[17]. Linda , G., Shapiro, and George, C. Stockman(2001): "Computer Vision", pp 279-325, New Jersey, Prentice-Hall

[18]. Mysliwiec, A.T. 1994. FingerMouse: A Freehand Computer Pointing Interface, Vision Interfaces and Systems Laboratory Technical Report, VISlab-94-001, Electrical Engineering and Computer Science Department, The University of Illinois at Chicago.

[19]. Nadgeri, Sulochana , M., Sawarkar, S.D., Gawande, A.D., "Hand Gesture Recognition Using CAMSHIFT Algorithm," Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on, vol., no., pp.37- 41, 19-21 Nov. 2010

[20]. 20) Nallaperumal, K., Ravi, S., Babu, C. N. K., Selvakumar, R. K., Fred, A. L. C., Seldev, and Vinsley. S. S. (2007). Skin Detection using Color Pixel Classification with Application to Face Detection: A Comparative Study. International Conference on Computational Intelligence and Multimedia Applications 2007.

[21]. Oavlovic,V.I., Sharma,R. and Huang,T.S. "Visual Interpretation of Hand Gestures for Human-Computer Interaction: AReview. "IEEE Transactions on Pattern Analysis and Machine Intelligence, vol . 19, pp.677-695, 1997.

[22]. Oviatt,S. (2001) Designing Robust Multimodal Systems for Diverse Users and Environments. EC/NS Workshop on Universal Accessibility of Ubiquitous Computing

[23]. Parekh, N, (2011) The Waterfall Model Explained .Availablefrom:http://www.buzzle.com/editorials/1-5-2005-63768.asp [Accessed: 03/03/2011]

[24]. Pressman, R.S (2010). Software Engineering: A Practitioner's Approach, (7th Edition) New York:McGraw – Hill. FYP Task 3 Project Report Student ID: 0822577 49

[25]. Quek,K.H.F., 1994. Non-Verbal Vision-Based Interfaces, Vision Interfaces and Systems Laboratory Technical Report, Electrical Engineering and Computer Science Department,The University of Illinois at Chicago.

[26]. [Figure 2.2] rickyc (2004) Github Social Coding [online image].Available from: https://github.com/rickyc/contour-detection [Accessed: 21/02/2011].

[27]. Redstone Software Inc (2008) Black-box vs. White-box Testing: Choosing the Right Approach to Deliver Quality Applications [WWW]. Available from: http://www.testplant.com/download_files/BB_vs_WB_Testing.pdf [Accessed : 05/03/2011]

[28]. Rodrigo.R, (2009) Image Processing and Computer Vision [WWW] University of MoratuwaAvailable from: http://www.ent.mrt.ac.lk/~ranga/files/mathematics_society_talk.pdf [Accessed 28/02/2011]

[29]. Segen,J. and Kumar,S. (1998). Gesture VR: Vision-based 3D hand interface for spatial interaction. ACM International Conference on Multimedia. p. 455-464.

[30]. Search Software Quality (2001) systems development life cycle (SDLC) [WWW]. Available from: http://searchsoftwarequality.techtarget.com/definition/systems-development-lifecycle [Accessed: 06/03/2011]

Short Bio Data for the Author

	**Sagar Srivastava** is a student of Information Technology at IMS Engineering College, Ghaziabad (U.P). He is in 4th year and has been member of several academic and internship/ training based events. During his academics he has coordinated/ participated in several technical fests, Workshops, Industrial visits and Seminars at Institute and University Level.).

