An Implementation of SEAL – mouSE virtuALized

Alita Anna Fernandes¹, Anisha Leema Noronha ², Ashrita Saldanha ³, Deepthi Nayak D⁴, Gayana M N⁵

Department of Computer Science and Engineering
St. Joseph Engineering College, Mangaluru

Abstract— In today’s world, technology is evolving day by day. One such promising concept is Human- Machine Interface and virtualization technology. The application SEAL is used to give the user a virtual experience on any opened application. In a wired mouse there is no provision to extend limit. In wireless mouse, one should have Bluetooth hardware installed in the computer and Bluetooth dongle attached. The proposed technology will have no such limitations and will instead depend on gesture recognition. Initially calibration is done for camera followed by clicking the ‘ON’ button. The system now monitors the hand movements and moves the cursor accordingly. When there is a flash off blue depending on the position with respect to red color the right click or left click is initiated. Therefore, the user can now use his/her hand to control the computer.

Keywords— Webcam, Color Rings, Euclidian filter, Color detection, Mapping.

I. INTRODUCTION

A computer mouse is a hand-held pointing device that is used to detect two-dimensional motion relative to a surface which is then translated into the motion of a pointer. There is no other more popular navigation/interaction device than the mouse. In order to satisfy increasing need to use interactive applications, SEAL provides a platform to allow users to use the hand gestures as mouse functionalities. As long as the main PC has a camera be it a webcam or usb camera, it can enable the user to use SEAL. SEAL efficiently tracks the user’s hand position and initiates click. It constantly monitors the movements of the finger and initiates the required click based on the color and its position with respect to red color. In this project the mouse is represented by the use of fingers. Camera detects the real time movement of the finger and the coordinates will be taken as the input of the mouse. It is an easy, user-friendly technology which supports human computer interaction in a hands free fashion.

II. COMPONENTS

The components used in this project can’t be specific, since this project is a prototype for all computers. As such, certain prerequisites are as follows:

A. Webcam:
Webcam is a necessary component for detecting the image. Sensitivity of mouse is directly proportional to resolution of camera. If the resolution of camera is good enough, an enhanced user experience is guaranteed. The webcam serves the purpose of taking real time images whenever the computer starts. On the basis of gestures and motion of fingers, system will decide the respective action.

B. Rings:
Rings as shown in figure 2 with a colour is a necessary component here. It is essentially the first step of the project. The rings will have different colours painted on them for different fingers. This clear distinction in colors will make the system faster and easier to process.

III. TOOL CHAIN FOR FIRMWARE

In software, a tool chain is a set of programming tools that are used to perform a complex software development task or to create a software product, which is typically another computer program or a set of related programs. This work makes use of such an equivalent platform each of them is listed below:

A. Microsoft Visual studio
Visual Studio is an Integrated Development Environment (IDE) used to develop computer programs. It is developed by Microsoft. Visual Studio uses Microsoft development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native and managed code. It includes a code editor supporting.

B. .NET Framework

International Conference on Smart Computing Communication & Control (ICSC3 2018), Organized by St Joseph Engineering College, Mangaluru, Karnataka, India.

Page 83
.NET Framework (pronounced dot net) is a software framework developed by Microsoft that runs primarily on Microsoft Windows. Framework Class Library (FCL) is a large class library that provides language interoperability (each language can use code written in other languages) across several programming languages. Programs written for .NET Framework execute in a Common Language Runtime (CLR), an application virtual machine that provides services like security, memory management, and exception handling. Computer code written using .NET Framework is referred to as "managed code". FCL and CLR are the integral parts of .NET Framework.

IV. SYSTEM DEVELOPMENT

There are four main modules in the development of SEAL application. They are:

- Camera Calibration
- Filtering
- Color detection
- Mapping

RGB colors are used to provide basic mouse functionalities. Red color is used to control cursor movement, Green color is used to perform double click at the current cursor position, and Blue is used to perform right and left click depending on its position with respect to red color.

A. Camera Calibration and Image acquisition

Webcam or usb camera can be used to capture the images. In this project, we make use of webcam for this purpose. The camera needs to be calibrated before capturing the images. The camera can be calibrated to capture the image in either normal mode or stretch image mode. Normal mode capture only a part of the real world whereas the stretch image is of the calibration similar to a video call. Set camera to capture in stretch image mode by default. The images captured will be displayed in the picture box. ON and OFF buttons can be used to turn the camera on and off respectively. Timer is used to render the image frame in real time that makes a illusion of a video. The interval between the frames is set to milli seconds and the same can be modified in the properties of the timer so as to make the frame flow out smoothly as possible.

B. Filtering/ Pre processing of the frame

The frames each that are captured are filtered in next step and the filtering is done using Euclidian filter in this work. The filter takes a specified color as the centre color (say red) and a value for the radius or a given boundaries. It keeps pixels with colors inside/outside of the specified sphere and fills the rest with specified color. In this project a radius value of 100 bytes is used. Setting color to the filter is normally a trial and error method considering the background and other constraints. The filtered image is then displayed to cross verify with user this is only to make sure the proper shade of the color is filtered out based on the environment the user is in.

The Code snippet for red is given below:

```csharp
// create filter
EuclideanColorFiltering filter = new EuclideanColorFiltering();
// set center color and radius
filter.CenterColor = new RGB(179, 0, 0);
filter.Radius = 100;
// apply the filter
filter.ApplyInPlace(image);
```

If red is to be filtered:
If blue is to be filtered:

![Capture Apply]

**Fig 3. Filtering Blue**

If green is to be filtered:

![Capture Apply]

C. **Color detection or Tracking**

The filtered image is then processed to find the brightest shade of the color in the picture. Here we infer that the light falling on the color band also matters to give the brightest point. This aspect depicts the importance of light and its dependency to show a color to the camera lens. We use the track bar to find the exact value at which the brightest point can be obtained. One could hope for a work that gives a precise value so that the point can be found out independent of the background.

D. **Mapping – Cursor functions**

Now that the brightest point is obtained, the cursor of the system is mapped to the same. We have given different color codes to various functional operations of the cursor. Red for movement, Blue for right click or left click, and Green for double click. It is directly done using the system functions in .NET, taking x and y coordinates and corresponding action of the point on the screen is executed.

<table>
<thead>
<tr>
<th>No.</th>
<th>Observation</th>
<th>Accuracy</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><strong>Background-Black</strong></td>
<td>Yes</td>
<td>80</td>
</tr>
<tr>
<td>2.</td>
<td><strong>Background-Red, Green, Blue</strong></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td><strong>Frame Capturing</strong></td>
<td>Yes</td>
<td>100</td>
</tr>
<tr>
<td>4.</td>
<td><strong>Frame Filtering (For specified RGB)</strong></td>
<td>Yes</td>
<td>100</td>
</tr>
<tr>
<td>5.</td>
<td><strong>Mapping</strong></td>
<td>Yes</td>
<td>85</td>
</tr>
</tbody>
</table>

V. **OBSERVATIONS MADE**

VI. **CONCLUSION**

This project uses Euclidian filtering process in Visual Studio for object detection and control the mouse cursor via hand gestures. It completely eliminates the necessity of mouse. This project proved to be practically successful and the movement of mouse cursor is achieved with a good precision accuracy. This project doesn’t require expensive hardware and can be done using standardized operating system such as Windows. Conclusively, SEAL is an enhancement of existing mouse.
VII. FUTURE SCOPE

This project can be enhanced by working on the various background and its filtration. This work can be made precise by using different platforms and languages for better working of this prototype. This project can also be enhanced by using better algorithm for fast detection of hand gestures so that the system responds differently to different users based on their usage.

References
[7] https://en.m.wikipedia.org/wiki/Microsoft_Visual_Studio