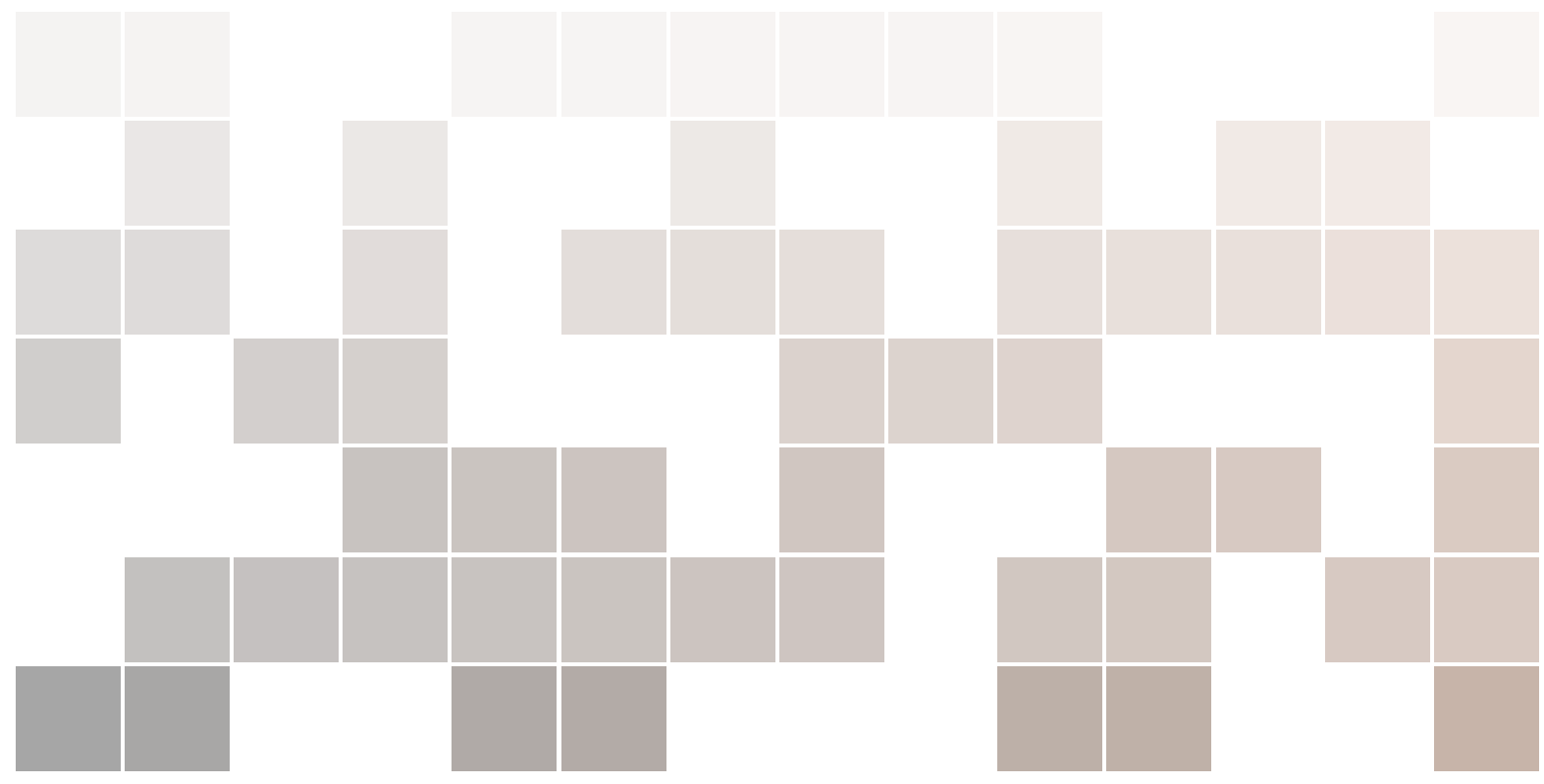


TBD

Names go here



First printing, July 2014

Contents

1	Introduction	5
1.1	Motivation	5
1.2	Related Work	6
1.2.1	Eye Tracking	6
1.2.2	Gesture Recognition	8

1. Introduction

Our project is based on two techniques, which are eye tracking and gestures recognition. Using these techniques, we can facilitate Human Computer Interaction. This will be very useful for people who have some disabilities. Handicapped who are unable to use their hands to touch the mouse and keyboard can interact with computers using their eye only. With eye tracking, we can detect the gaze point on the screen and determine where the person looks at in order to facilitate the handicapped communication.

The deaf people always use the sign language to interact with the things around them, Depending on the gesture recognition technique, It will be very useful for them to interact with the device using some gestures.

In general our application can make the interaction with the computer or any mobile device becomes easier, Even if the users can have no disabilities. It is very attractive if we use our eye to communicate with the screen or some signs with your hands to move through it, We will have no need for mouse or keyboard or any heavy devices.

1.1 Motivation

The Motivation of our system is that we find that we always need to use mouse and keyboard to interact with computer devices such as PC devices and Laptop Devices. With the development in the technology which appears around us every day, the sizes of such devices become smaller and smaller, so we found that there is a motivation to try to find a way by which we could interact with these small devices without need to use other tools to interact with the devices. Therefore we need to cope with the modern technology.

On other hand, we need to facilitate human computer interaction to make it easier to interact with the device. It will be attractive if we interact with computer devices with our senses such as eye or hands without using tools to translate what we want to do with the device. There is a pressing need to find a way to facilitate the communication language between the man and the machine. There is a motivation to discover some way to facilitate HCI in general.

We use the mouse to move through the screen which gives some commands to the computer device to translate and then process them. Also we use the keyboard to give the commands to the device. How about using our eye to interact with the screen? So the computer device will understand what we want from our eye movement which is extremely comfortable and usable for our needs. So we found that it is a strong motivation to use Eye Tracking Techniques based on Computer Vision. So the communication language between the user and the device is the eye movement which facilitates HCI. Depending on our hands, we can make some signs with our hands according to the action we want to do then, we detect the gesture which is made by hands and translate it to the specific action which facilitates HCI. So our application tries to use our senses to facilitate the way of interaction between the man and the device.

Also, there is a serious motivation for our project which is how to make the people who have some disabilities to interact with the computer device without find any difficulties. If this happens, the person with the disability will find it is so easy to use the computer device and interact with it like any person which can develop his abilities and skills. So the most important motivation is to facilitate human computer interaction for people with some disabilities.

Handicapped who are unable to use their hands to touch the mouse and keyboard can interact with computers using their eye only. With eye tracking, we can detect the gaze point on the screen and determine where the person looks at in order to facilitate the handicapped communication.

Deaf people always use the sign language to interact with the things around them, Depending on the gesture recognition technique, It will be very useful for them to interact with the device using some gestures. They may not find difficulty to use mouse or keyboard but, when they find that they can interact with the computer device with their familiar language, this will facilitate their interaction and it could be very attractive for them.

1.2 Related Work

1.2.1 Eye Tracking

Eye tracking is to collect data using either a remote or head-mounted 'eye tracker' connected to a computer. Most generally eye trackers include two common components: a light source and a camera. The light source(usually infra-red) is directed towards the eye. The camera tracks the reflection of the light source along with visible ocular features such as pupil. This data is usually used in estimating gaze of point.

There are many different methods for exploring eye tracking data. The most common is to analyze the visual path of one or more participants across an interface such as a computer screen. So, the presence or absence of eye data points in different screen areas can be examined. This type of analysis is used to determine which features are seen, when a particular feature captures attention, how quickly the eye moves, what content is overlooked and virtually any other gaze-related question.

Think of all the ways that you will use your eyes today. You might drive a car. You might read a magazine. You might surf the internet or play a video game or complete a training exercise or watch a movie or look at pictures on your mobile device. These are the applications of eye tracking. With very few exceptions, anything with a visual component can be eye tracked. We use our eyes almost constantly, and understanding how we use them has become an extremely important consideration in research and design.

Now days, the automotive and medical industries have applied eye tracking technology to make us safer. The fields of advertising, entertainment, packaging and web design have all benefited significantly from studying the visual behaviour of the consumer. Research with special populations has generated exciting breakthroughs in psychology. Every day, as eye tracking is used in creative new ways, the list of applications grows.

Today's Eye-tracking algorithms use two main approaches: feature-based and model-based approaches.

Feature-based approaches detect and localize image features related to position of the eye. There is an important property among all feature-based algorithm which is criteria (e.g. a threshold) needed to decide when a certain feature is present or not. Usually the determination of the criteria or the threshold left to the user. For example on the feature-based algorithm is Starburst because it depends on the corneal reflection. [starburst]

On the other hand, model-based approaches don't explicitly detect features but rather find the best fitting model that is consistent with the image. For example Pupil Detection algorithm is model-based approach. However, model-based approaches require searching a complex parameter space that can be fraught with local minima and thus cannot be used without a good initial guess of the model parameters. The gain in accuracy of a model-based approach is obtained at a significant cost in terms of computational speed and flexibility but it is so promise for real-time performance. [starburst]

Starburst

Starburst is a robust algorithm for video based eye tracking. We started with implementing Starburst Algorithm. This algorithm is more accurate than pure feature-based approaches yet is significantly less time consuming than pure model-based approaches. A validation study shows that the technique can reliably estimate eye position with an accuracy of approximately one degree of visual angle even in the presence of significant image noise.

The use of eye tracking has significant potential to enhance the quality of everyday human-computer interfaces. Two types of human-computer interfaces utilize eye-movement measures active and passive interfaces. Active interfaces allow users to explicitly control the interface through the use of eye movements. For example, eye-typing applications allow the user to look at keys on a virtual keyboard to type instead of manually pressing keys as with a traditional keyboard. Similarly, systems have been

designed that allow users to control the mouse pointer with their eyes in a way that can support, for example, the drawing of pictures. Active interfaces that allow users with movement disabilities to interact with computers may also be helpful for healthy users by speeding icon selection in graphical user interfaces or object selection in virtual reality. Passive interfaces, on the other hand, monitor the user's eye movements and use this information to adapt some aspect of the display. For example, in video transmission and virtual-reality applications, gaze contingent variable-resolution display techniques present a high level of detail at the point of gaze while sacrificing level of detail in the periphery where the absence of detail is not distracting. While eye tracking has been deployed in a number of research systems and to some smaller degree consumer products, eye tracking has not reached its full potential.

Robust Realtime Pupil Tracking

In this paper, they present a real-time dark-pupil tracking algorithm designed for low-cost head-mounted active-IR hardware. Their algorithm is robust to highly eccentric pupil ellipses and partial obstructions from eyelashes, making it suitable for use with cameras mounted close to the eye. It first computes a fast initial approximation of the pupil position, and performs a novel RANSAC based ellipse fitting to robustly refine this approximation.

Pupil

Pupil is an open source platform for pervasive eye tracking and mobile gazed interaction. In this paper, they argue that affordability does not necessarily align with accessibility. They define accessible eye tracking platforms to have the following qualities: open source components, modular hardware and software design, comprehensive documentation, user support, affordable price, and flexibility for future changes.

They have developed Pupil, a mobile eye tracking headset and an open source software framework, as an accessible, affordable, and extensible tool for pervasive eye tracking research. They explain the design motivation of the system, provide an in depth technical description of both hardware and software, and provide an analysis of accuracy and performance of the system.

1.2.2 Gesture Recognition

Traditional human-computer interaction devices such as the keyboard and mouse became ineffective for an effective interaction with the virtual environment applications because the 3D applications need a new interaction device. An efficient human interaction with the modern virtual environments requires more natural devices. Among them the “Hand Gesture” human-computer interaction modality has recently become of major interest. The main objective of gesture recognition research is to build a system which can recognize human gestures and utilize them to control an application. Hand gestures are a collection of movements of the hand and arm that vary from the static posture of pointing at something to the dynamic ones used to communicate with others. Recognizing these hand movements needs modeling them in the spatial and temporal domains. Hand posture is the static structure of the hand while its dynamic movement is called hand gesture and both are particularly crucial for human-computer interaction. The methods

used for understanding these structures and movements are among the most classifying researches that still in progress.

Methods for vision based hand gesture recognition fall into two categories: 3D model based methods and appearance model based methods. 3D model may exactly describe hand movement and its shape, but most of them are computational expensive to use. Recently there are some methods to obtain 3D model with 2D appearance model such as ISOSOM and PCA-ICA in [relatedsg1] and [relatedsg2].

Static Gesture Recognition

There have been a number of research efforts on appearance based method in recent years. Freeman and Weissman recognized gestures for television control using normalized correlation [relatedsg3]. This technique is efficient but may be sensitive to different users, deformations of the pose and changes in scale, and background. Lars and Lindberg used scale-space color features to recognize hand gestures [relatedsg4]. In their method, gesture recognition method is based on feature detection and user independent while the authors showed real-time application only under uniform backgrounds.

Yikai Fang, Kongqiao Wang proposed "A REAL-TIME HAND GESTURE RECOGNITION METHOD" [relatedsg5] which based their work on Lars's work. They combine fast hand tracking, hand segmentation and multi-scale feature extraction to develop an accurate and robust hand gesture recognition method. It takes advantage of color and motion cues acquired during tracking to implement adaptive hand segmentation.

Our initial work was inspired by a combination of Yikai, Wang's work and a method described in[survey]. We made an initial approach for hand detection and static gesture recognition based on the following procedure:

- Detect hand from image based on:
 - Skin color detection.
 - Hand is assumed to be the largest contour.
- Extract Number of fingers, center of hand.
- Identify gesture based on outputs from previous module using blocks of condition statements based on:
 - Number of fingers.
 - Their angles with respect to x, y axis.

This approach has many problems that rise during the implementation:

- There is a threshold to determine which points on the contour represent a finger point. That threshold is not reliable and sensitive to:
 - Orientation of the hand.
 - Size of the hand.
- Then we modified the threshold to be relative to hand size by using an equation that holds the hand width, height, and center point as variables. That modification solves the reliability to hand size problem, but doesn't solve the orientation problem.

- Unreliable if for example we want to distinguish between up gesture made by index finger, or thumb finger.

To solve the problems with the initial work, we directed our thoughts to machine learning. Where, we extract features from hand and use classification as described in paper [relatedsg6].

So we defined Three Approaches for static gestures. They differ on features used to represent the gesture:

1. Binary Representation Gridding as described in [].
2. Predefined Feature Extraction as described in [paper2sT].
3. SIFT And Bag of Features Approach as described in [dardas].

Dynamic Gestures

During our search for already existing approaches for hand gestures, we came across a method for detecting continuous gestures by hand trajectories. Hidden Markov Model based [hoda] continuous gesture recognition system is Statistical Model capable of modeling spatio-temporal time series where the same gesture can differ in shape and duration. This method defines a gesture as Spatio-Temporal Pattern which may be Static, Dynamic or Both. The objective of this approach according to the paper explained it, was to Recognize gestures for drawing any number from (0-9) one or two digits.

This approach assumes that the hand has been detected and its articulated motion is tracked. Although they have delivered promising results, the robustness of these approaches is dependent on the prior success of (frequently challenging) hand detection and motion tracking. In addition, it is both data intensive and computationally difficult to train these models before they can be applied in recognition. For these reasons, we didn't pick this method in our system. We need a relatively fast processing. Meanwhile, we need our method not to have an overhead on the processor. We deal with relatively small processors with limited capabilities and so, our work has to comply with this limitation.