**Virtual Keyboard**

Project report submitted for

**V Semester Minor Project**

By,

**Himanshu Khairajani (17100017)**

**Saurabh Pandey (17100049)**

**Saurabh Roy (17100050)**

Under Guidance of,

**Dr. Muneendra Ojha**

**Assistant Professor, Department of CSE**



**Department of CSE**

**Dr. Shyama Prasad Mukherjee**

**International Institute of Information Technology, Naya Raipur**

**(A Joint Initiative of Govt. of Chhattisgarh and NTPC)**

**Email:** [**iiitnr@iiitnr.ac.in**](mailto:iiitnr@iiitnr.ac.in)**, Tel: (0771) 2474040, Web:** [**www.iiitnr.ac.in**](http://www.iiitnr.ac.in)

**Certificate**

This is to certify that the project titled “**Virtual Keyboard**” by **Himanshu Khairajani, Saurabh Pandey, Saurabh Roy** has been carried out under my supervision.

**Dr. Muneendra Ojha**

**Assistant Professor**

**Department of CSE**

**Dr. SPM IIIT-NR**

**May, 2019**

**Declaration**

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Dated:** September 11,2019 **Himanshu Khairajani**

**17100017**

**Saurabh Pandey**

**17100049**

**Saurabh Roy**

**17100050**

**Acknowledgement**

We would like to express our deep sense of gratitude to our mentor **Dr. Muneendra Ojha** for his guidance, help and encouragement throughout our project work. We are grateful to him for patiently checking all our progress and proceedings time to time. We would also like to acknowledge the great internet facility provided by the college, without which it would not have been possible to complete this project. The help we received from our fellow batchmates is invaluable. We also acknowledge any other person that was directly or indirectly involved in successful completion of this project.

**Dated:** September 11,2019 **Himanshu Khairajani**

**17100017**

**Saurabh Pandey**

**17100049**

**Saurabh Roy**

**17100050**

**Approval Sheet**

This project report entitled “**Virtual Keyboard**” by **Himanshu Khairajani, Saurabh Pandey, Saurabh Roy** is approved for V Semester Minor Project.

**Date:** September 11,2019

**Venue**: Lecture hall 206, IIITNR **Dr. Muneendra Ojha**

**Dr. Mithilesh Chaube**

**Dr. R Cheruku**

**Abstract**

The basic idea of a virtual keyboard in an [virtual](https://en.wikipedia.org/wiki/Augmented_reality) environment is to give the user a text input possibility. A common approach is to render a flat keyboard into the augmented reality, e.g. using the [Unity](https://en.wikipedia.org/wiki/Unity_(game_engine)) Touch Keyboard.

***Keywords -*** *Posterior Probability, Feature Vector, Highest Posterior Probability, Array of Pre-stored Key, Personal digital assistant (PDA), Computer vision (CV)*

**Table of Contents**

1. **Abstract………………………………………………………….. vi**
2. **Table of Contents……………………………………………….. vii**
3. **Introduction……………………………………………………... 1**
4. **Motivation……………………………………………………….. 1**
5. **Aim……………………………………………………………….. 2**
6. **Objectives………………………………………………………… 2**
7. **Resources Used…………………………………………………... 3**
8. **Description of project…………………………………………… 4**
9. **Demo of the project …………………….……………………… 5**
10. **Conclusion………………………………………….…………… 6**
11. **References…………………………………………………****….…. 6**

**1. Introduction**

A **virtual keyboard** is a software component that allows the input of characters without the need for physical keyboard. The interaction with the virtual keyboard happens mostly via a touchscreen interface, but can also take place in a different form in virtual or augmented reality.

**2. Motivation**

On a desktop computer, a virtual keyboard might provide an alternative input mechanism for users with [disabilities](https://en.wikipedia.org/wiki/Disability) who cannot use a conventional keyboard, or for bi- or [multi-lingual](https://en.wikipedia.org/wiki/Multilingualism) users who switch frequently between different character sets or alphabets, which may be confusing over time.

**3. Aim**

Our aim is to use wearables to track the finger motion to replace a physical keyboard with virtual ones. we also tried to give an audio-visual feedback to the user, when a key got hit. The basic idea is to give the user a more natural way to enter text, based on what he is used to.

**4. Objectives**

The major objectives is to design a vision based Virtual Keyboard which detects interrupt as key recognition instead of mechanical transducer operations of key pressing. Mono-vision video of hand posture for pressing the keys is analysed. The analysed hand posture is taken into account under various transactions to estimate the key pressed. Mechanical transducers does two operations for key estimation (key press and key release concepts), while Virtual Keyboard requires only key press operation to estimate the key and not key release operation.

**5. Resources Used**

**Hardware requirements**-

* Web camera
* Stylus

**Software requirements**-

* OpenCV
* Python 3.2.1
* Spyder IDE
* Wx python
* Playsound module

1. **Description**

**Object Extraction**

Feature extraction for the purpose of object recognition consists of segmentation of image components that contribute to the formation of inputs. Only specific colored object (pen, colored gloves, etc.) form the basis of feature extraction.

**Threshold**

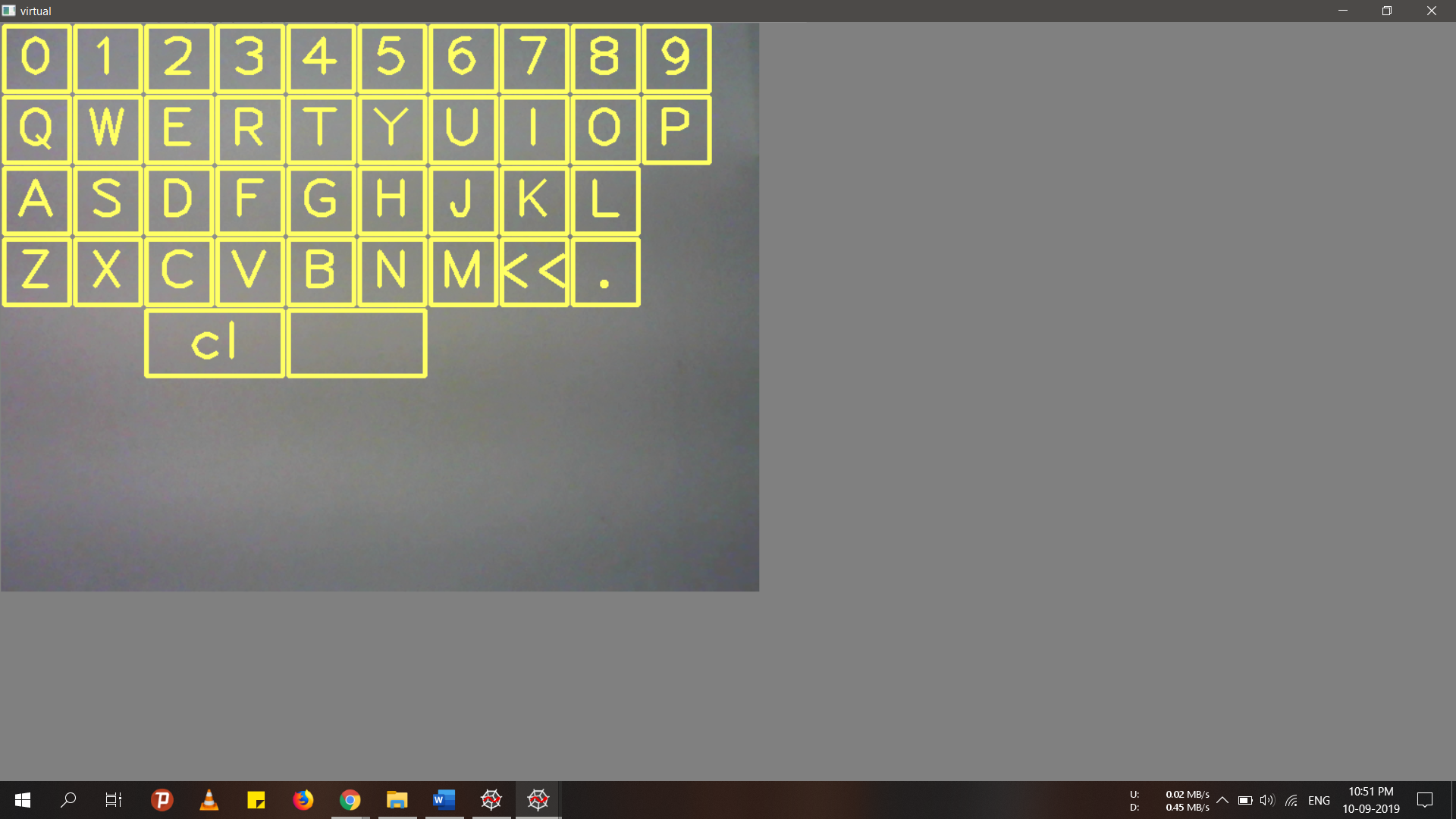
Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images. During the thresholding process, individual pixels in an image are marked as “object” pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as “background” pixels otherwise. This convention is known as threshold above. Variants include threshold below, which is opposite of threshold above; threshold inside, where a pixel is labeled "object" if its value is between two thresholds; and threshold outside, which is the opposite of threshold inside.

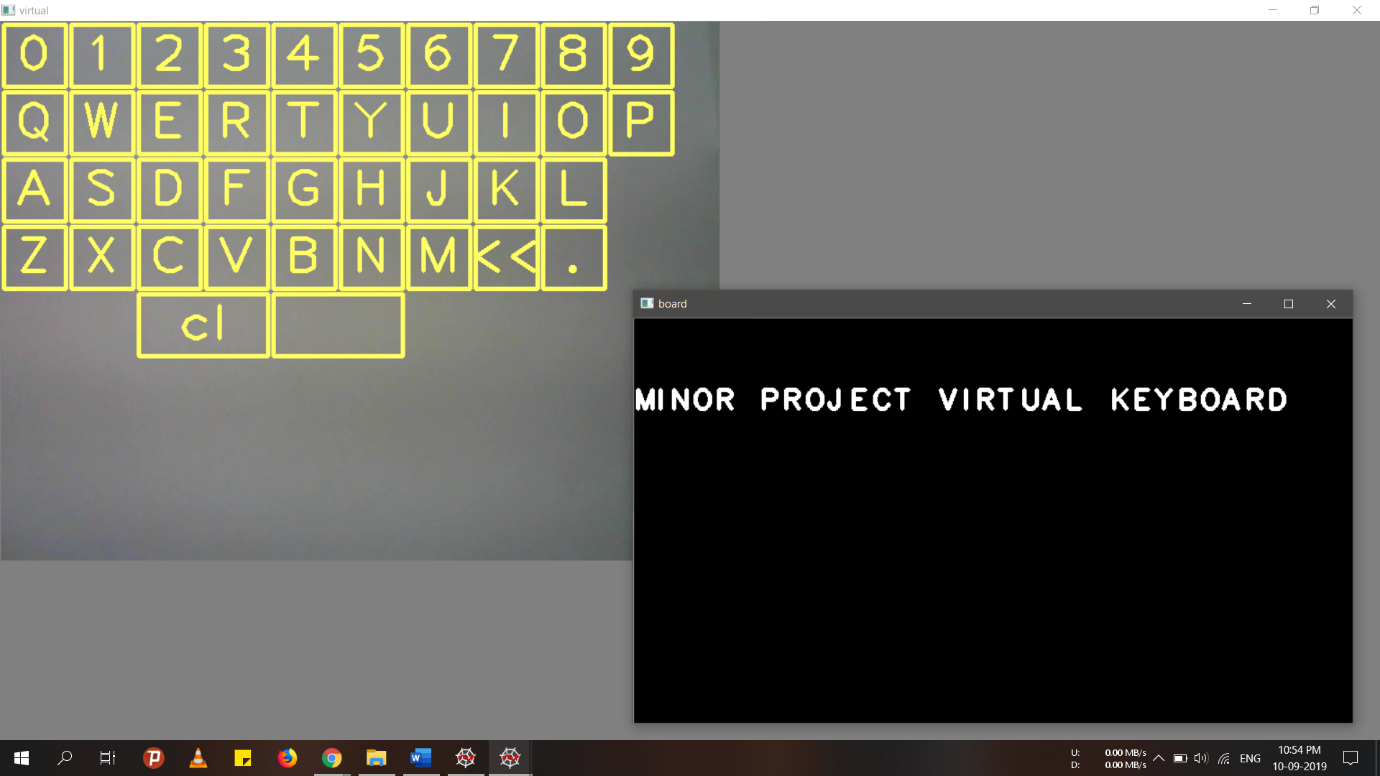
**Edge Detection Mechanism**

Edge detection is a fundamental tool in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally which has discontinuities. The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to:

* discontinuities in depth,
* discontinuities in surface orientation,
* changes in material properties and
* Variations in scene illumination.

1. **Demo of Project**

****

****

1. **Conclusion**

Results showed a very reliable and practical system. The proposed system is less cost due to its software centric mechanism rather a hardware centric mechanism. Performance of the system had been tested over Personal Computer. The data set involved in the development of the system can be easily altered as user requests. Standard data set style is implemented. Response time for Key Extraction is less quite compared to system like Finger-Joint Gesture Wearable Keypad, Thumb code, etc.

1. **References**

**URL:**

1. Youtube.com (video related to OpenCV for object detection using image segmentation.)
2. <https://pysource.com/> (learn how to make keyboard layout.)
3. [https://opencv.org](https://opencv.org/) (learn contour extraction of specified object.)