VIRTUAL KEYBOARD USING IMAGE PROCESSING

Prof. R. G. Gupta[1],Raguru Shrikanth Reddy[2], Nakul Navnit Parmar[3]

[1]Assistant Prof., Department of Electronics, Pravara Rural Engineering College, Loni, India
[2]B.E, Department of electronics engineering, Pravara rural engineering college, Loni, India.
[2]B.E, Department of electronics engineering, Pravara rural engineering college, Loni, India.

Abstract -*People within the modern times are busy with serious workloads and that they don't have even one second to waste. they struggle to cut back their everyday work by using standard computing devices like Desktop Computers and Laptops, in addition more advanced devices like mobile devices and PCs. To improve the movability and quality individuals tend to cut back the dimensions of the devices. Thus most of the mobile devices and hand-held devices contain really small keypads. A number of the individuals who use such a tool find it tough to see the letters on the keyboard. With a mobile device, a full size physical keyboard isn't ideal. But alternatives are available in the form of written recognition, speech recognition etc. however all of them lack the accuracy and convenience of a full size keyboard.*

This paper presents an innovative virtual keyboard to beat the above mentioned issues, which might be a viable replacement for fold-up keyboards. The keyboard primarily based on the vision based human-computer interaction concept, image capturing and image processing technique that contains virtual keys adequate to the dimensions of traditional keys within the normal QWERTY keyboard. This keyboard consists of 3 main modules, namely, Image Capturing, Character Identification and device emulation. the end result of the virtual keyboard project was a cheap, user friendly and transportable virtual keyboard, that sends typed characters to any hand-held device, mobile device or laptop. This virtual keyboard offers the short touch-typing feedback to the user.

Key Words:*Interactive surface, virtual keyboard, raspberry pi, image processing.*

1. INTRODUCTION

Input data to pc devices is that the A B C in human computer interaction. Individuals use both the computing devices (desktops or laptops) and non-traditional computing devices (mobile devices) to complete their work quickly in homes and offices. Thus to deal with today's fast-paced business world they try to extend the movability and quality by reducing the dimensions of those devices. As a result of the information input unit of those devices get smaller and smaller they lack flexibility for a few users who find it difficult to visualize the letters written on every small key of the little keypads. However, after they carry an additional keyboard, kind of like the normal keyboard with a mobile unit, the quality and also the movability cannot be there. except for this, users who work with Personal Computers (PC) face issues like functional issues owing to technical faults, position of units, similarly because the usage of physical keyboards will rely upon the operating surroundings, as an example clean rooms wherever keyboards aren't allowed. to overcome these issues, one approach is to have a Virtual Keyboard, that dynamically vary the key arrangement for various users, situations, or perhaps from keystroke to keystroke [1, 2] like Visual Panel, FingeRing and touch Stream to name a few. However these have their own drawbacks, like high price, constant key location must be pressed repeatedly to enter information, slow typewriting speed, user must wear gloves or information processing units, risking damage to the keys, etc. However, the virtual input devices open the path for a unique scheme known as vision based human-computer interaction during which normal input and output devices are replaced with augmented reality displays [3], projection systems [8] and cameras. This idea gives birth to an innovative technology permitting the use of light made full size keyboards on mobile devices like PDAs, palmtops and different transportable computing devices. This new technology allows a keyboard to be projected onto a flat surface employing a beam of light, which can then be written on. The present projection keyboards have been discussed at length in [1,3], Some examples of such keyboards are Senseboard, VKB Projection, light key [3,10] and Vkey. However, except the Senseboard keyboard, all the others lack giving touch-typing feedback to the user, particularly for the touch typists. Moreover the users of Vkey, LightKey, and VKB projection cannot rest their hands on the keyboard image by keeping the fingers on them. All of them are tough to be employed by individuals with paralysis or muscular disorder, as the typewriting technique could be a bit kind of like the traditional typewriting technique. These are considered as foremost drawbacks of existing product, which then, can be considered as important of featuresof this keyboard. The approaches used for this projection keyboards like for the Senseboard are mentioned in [1,3,7]. Senseboard uses AI alongside laser technology. VKey uses laser technology with visual sensors (cameras) [1,3] for the detection of movement of all the fingers. LightKey uses laser technology with electronic perception technology [3,9,10] and VKB Projection uses laser sensors (infrared cameras) for the detection of the keystrokes of all the fingers [1,3]. The virtual keyboard proposed in this paper uses a unique approach [6] to spot the letters in line with the touching movements of the user. That is, Digital image capturing and image processing techniques [4,5] alongside the traditional projection techniques. A tiny slide projector projects an image of the complete size QWERTY keyboard. The choice of characters is done by a LED based electronic stylus. Once the user selects a character the virtual keyboard by pressing the light stylus against the required virtual character, the blue LED bulb within the electronic stylus is illuminated therefore giving the fast touch-typing feedback to the user. Image capturing module captures these touching events of the pen and also the virtual keyboard employing a web cam, and sends it to the character identification module. This module identifies the letters according to the touching events of the pen. Then the device emulation module sends a signal with the identified, written characters to the device, which will then be displayed on the screen. The subsequent sections describe the techniques used to design the modules so as to realize the keystroke identification

2.DESIGN OF MODULE

Virtual keyboard is projected by a small projector on to a flat surface. The surface should be a sleek, non-glossy, static one. That background should be free of any objects. During developing the system by using digital image processing, having this kind of background is must. The touching movement is achieved by employing a light pen. The LED bulb of the light pen is blue in colour. The touching movement on the projected keyboard is captured in real time by the webcam and sent to the server for additional processing to extract necessary data. The image capturing module of the system grabs the image one by one and stores it in a buffer in first in first out (FIFO) basis, until the image processing module processes the image.The image processing module initially analyses the image that is captured first to spot the key locations of the keyboard on real time.Subsequently the second image is processed according to the user selected device at the user interface, to spot the character of the particular location of the key that has the touching event of the pen.This character is sent to the respective device as a signal returning from the server through communication media (Infra-red, BlueTooth,etc).

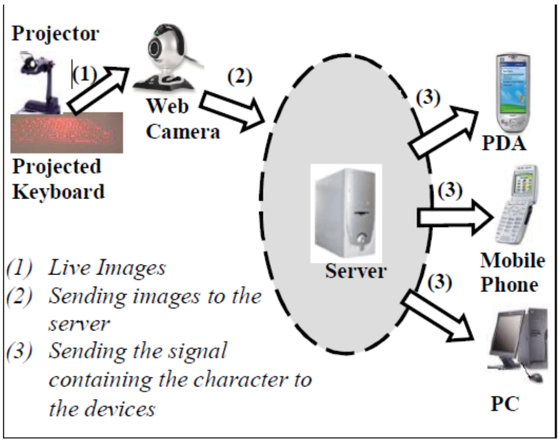


Fig -1: Conceptual design of system

2.1 Dividing into subsystems

The following diagram shows the way of dividing the full system into subsystem.

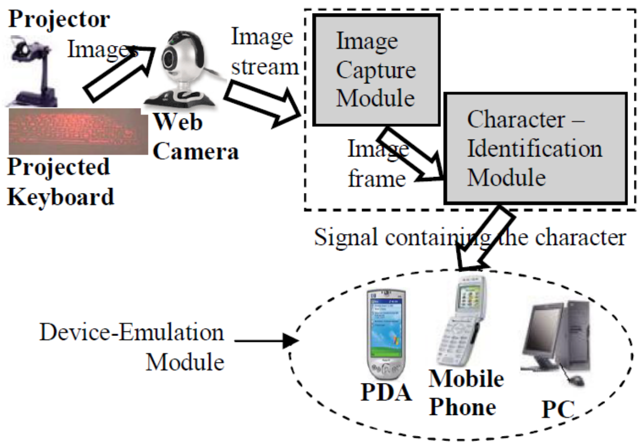


Fig-2: Division of system into subsystems

2.2 Image capturing module

This module consists of sequentially capturing images of the touching movements and storing those images in buffer acting as a queue. In this module comes into play once the user has selected the particular device that he/she desires to input information. Using the virtual keyboard at the user interface. Images are popped one by one for the character identification module. Before giving the initial captured frame of the buffer on to the character identification module for the aim of analysing the keyboard, this keeps backup copy of it. This backup copy along with the secondary returning images is going to the character identification module for the aim of identifying the pen location and for identifying the character.

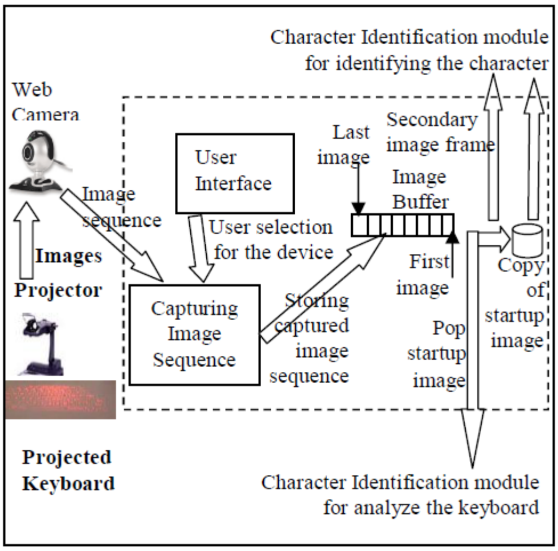


Fig-3: Image capturing subsystems

2.3Character identification module

Image- capturing module sends images to the module. Then one image goes through many image processing techniques to extract relevant data in every level. Some of the most vital data, which are identified by applying image processing techniques to every images are:

I. Analysing the keyboard to spot the key locations
II. Segmenting the pen from the background
III. Generating the letters

After detection of the edges of the keyboard from the primary start-up image. It calculates the dimensions of the keyboard. Thereafter it calculates the height of one column and width of one raw keyboard and calculates the key coordinates of every key and is stored in array with virtual key codes. After finding the blue pixel coordinates of an image there coordinates are compared with every raw of the array to find the keyblock area to spot the virtual key code. The virtual key codes are used to generate the letters.

2.4Device emulation module

The server receives the virtual code from the character identification module. Server 0/S accepts the signal from the buffer storage and forwards the signal to the keyboard driver. The appropriate window which has the input focus receives the signal from the keyboard device driver. There needs to be an interface to send signal containing the character from an external application to the emulators. However the manufacturers have already created and embedded this interface for commercially available emulators. Thus this prototype simulates the keystroke by sending the signal from character identification module to the buffer of the server

3. SOFTWARE IMPLEMENTATION

The main operation of the software implementation is:
1. Read red values of the keyboards by camera through I2C protocol
2. Process there image values within the microcomputer to get the pressed key
3. Convert the pressed key into the computer and obtain the key

3.1Camera operation and image processing

Initially, we will set the frame rate to six frames per second, enough time to distinguish two pressed keys. Then operate the getREDIndex function to receive values of the boundaries of the camera, with that we will be able to determine the areas of the keys. As we only receive the red values of the key, and the threshold is set to 19200, UBRRL=51, if the camera detects the red value higher than the threshold it'll send the position value of the key to the microcomputer. If else no information shall be sent. Third, if the queue is empty the getREDIndex function shall be operated to scan from camera; finally the microcomputer will scan the information if the line is idle. The uv and y line will receive the new frame information. On deciding the pressed keys, we use an array of eighty eight values on which we obtain the information on the information on the uv line every two PCLKS, which implies the pressed keys in two PCLKS will considered as same key. The camera will run the same set of pixels for consecutive rows and 2 lines processed to the equivalent pixel. As we do not have enough space to store the whole values the processing is done after each vertical line. HREF will be negative for about zero. 8ms and the camera information will be invalid after simply after every vertical line of valid data, which provides us enough time to process the information. If the pixel goes higher than the red threshold, we shall check if the pixel was a part of a contiguous line of red pixels, which will help determine the pressed key. We bounce the key by examining the contiguous four presses. If it's valid, we then add the key to the queue of keys and send them to the pc

3.2I2C communication

The I2C communication acts as a bridge between the camera and microcomputer. There are about ninety two registers available for is to set up the camera. A 2-wire communication scheme activated by a 10kOhm-pill u resistor is used by the protocol. The SCL line provides clock signal to the camera, and the frequency is set by (16 + 2 x (TWBR) (4TWPS)). Constants have been defined for TWCR register, codes for TWI master status and a few alternative functions. The bit rate register (TWBR) is set to seventy two and the status register (TWSR) is set to zero. Rest of the values are set in accordance to the standard protocol and with the above setting we will get the expected results

3.3Determining the edges

The determination of the edges of the camera vision is the main difficulty. After several experiments OV6630 that has a broad vision is selected as it could capture about 248 x 160 pixels, however the accuracy near the edges is a little low. Since we need to capture the image of entire A 3 sized keyboard we have to set the camera at 176 x 144. By setting the above resolution we can set the output format for capturing 16bit UV/Y data [2] where Y had GGGG data and UV had BRBR data. Y data was completely neglected.

3.4 Environmental Requirements

We are using red pixel as the threshold to identify the pressed key to the rest. In order to make sure that the red pixel of the background bellows the threshold, the surrounding conditions are expected to be a little dark. The chances of error increases if the strong light is offered in background. The desired lighting values of the background would be near about 10-70 Lm. After conducting the experiment and reporting the keys accuracy the results listed was as below.It is evident that the time required to type 1 character in virtual keyboard is significant as its need about eight times the time needed for physical keyboard. The accuracy of 88.61 % is considered good for a new systemThe only problem found in the experiment was the time needed for 1 character in virtual keyboard i.e. 2.92 s/character.It should be noted that this not the processing time in each sub-process that even combined amount is 22.64ms.

TESTED VALUE	PHYSICAL KEYBOARD	VIRTUAL KEYBOARD
Failure	0	18
Time Taken	51.8	461.3
Right character	140	140
Total character	140	158
Time per character	0.37	2.92
Accuracy	100	88.61

Table-1: Experimentation result

The big difference in time is only due to the time taken by webcam to take a picture, the time to display the info and the matching sys.(i.e 2 frames of the same position needed to trigger the input)

4. HARDWARE IMPLEMENTATION

4.1Raspberry pi2 model B

Developed in England by the Raspberry Pi Foundation, Raspberry Pi is single-board card-sized computer. The foundation tends to promote the importance of computer science in several schools and many developing countries. The original Raspberry Pi1 and Pi2 were fabricated in may board configurations via agreements with NEWARK ELEMENT 14 (Premier Famell), R S COMPONENTS and EGOMAN. The hardware is similar over entire manufactures Many Raspberry Pi generations are released till date. The 1st one was (Pi1) which was launched in Feb 2012. It was launched in basic model (A) & Higher specification model (B). In 2013 after a year A+ & B+ models were launched. In Feb 2015 Raspberry (Pi2) model (B) was launched and in Feb 2016 (Pi3) model (B) was launched. The prices of the boards range between 20 – 35 US$. Although Pi Zero with limited IO capabilities and smaller footprint was launched in Nov 2015 for 5 US$ and a cut down compute model was launched in Apr 2014.

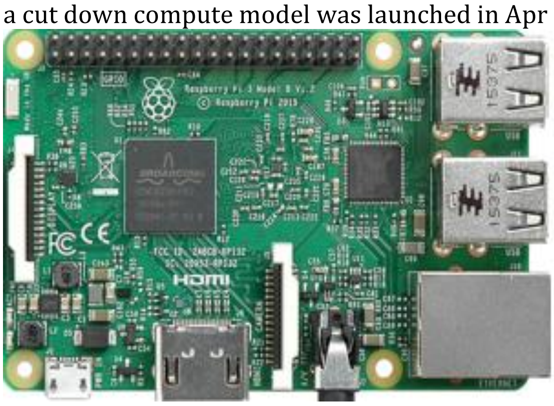
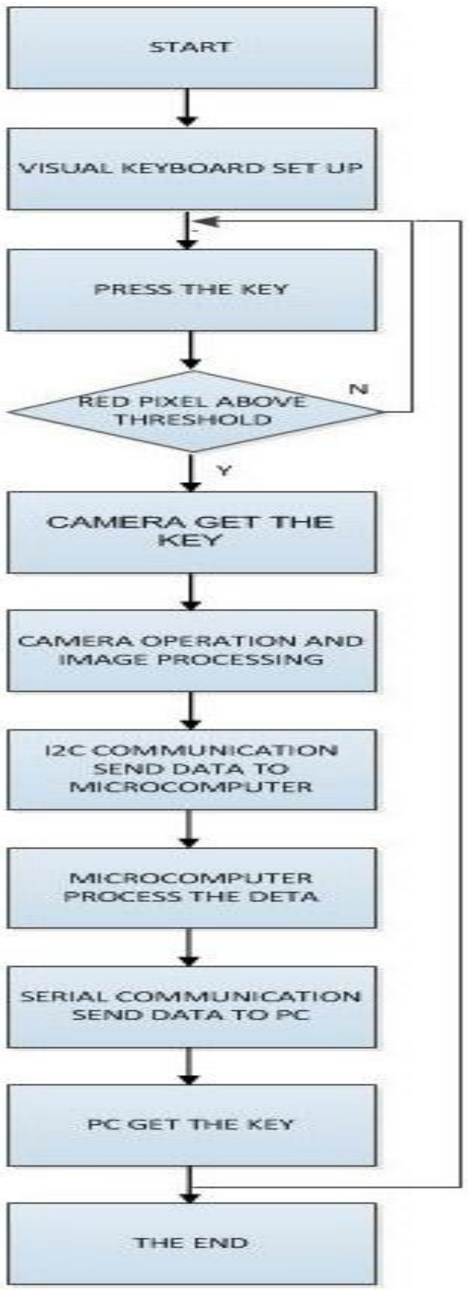


Fig-4: Raspberry pi2 model B

Every model exhibit a Broadcom system within a chip (SOC) which has an ARM compatible CPU along with an on chip processing unit GPU (Video core 4). The speed of the CPU ranges from 700MHz – 1.2 GHz for Pi 3 whereas the on board memory ranges from 256 MB to 1 GB RAM. Operating system and program memory are stored on secured digital SD cards in either SDHC or MicroSDHC sizes. 1-4 USB slots are available on most of the boards along with HDMI, Composite video o/p and a 3.5mm phono jack for audio. Number of GPIO pins are used to provide a lower level output which supports common protocols like I2C. Few models do have an RJ 45 ethernet port and Pi 3 contains on board Bluetooth and WiFi 802.11n. Debian and Arch Linux ARM distributions are provided by the foundation for download ant it promotes python as main programming language with support for C, C++, Ruby, Pearl, Squeak Smalltalk, BBC BASIC (via the RISC OS image or the Brandy Basiv Clone Linux) and more are available.



5. CONCLUSIONS

Compared with previous basic keyboard, Virtual Keyboard has rather more advantages for personalization, convenience, and applicable to special environments. For individuals with disabilities and uses where several rapid keystrokes are required Virtual keyboard can be of great use. This keyboard is one of the few which uses microcomputer to implement the functions unlike many virtual keyboards available in market. Hardware and Software implementations are explained in brief in this paper. The precision and accuracy of keys has been tested thoroughly. Based on this we confidently reach to a conclusion that this implementation is easily attainable and practical. The paper should serve as the instigator to describe the application of IR cameras and ARM11 in virtual keyboard. The flow chart shown in Figure 5 will help the fast developing computer industry

REFERENCES

[1]. Kolsch, M. and Turk, M."Keyboards without Keyboards: A survey of Virtual Keyboards," n. pag.University of California, Nov. 05, 2003 (http://www.create.ucsb.edu/sims/PDFs/Koelsch_and_Turk_SIMS.pdf).

[2]. Lawrence Erlbaum Associates Inc, "Human Computer Interaction," pp 89-129,2002(http://www.almaden.ibm.com/u/zhai/papers/ZhaiHunterSmithHCIGalley.pdf).

[3]. Alpern, M., "Projection Keyboards," Oct. 28, 2003 (http://www.alpern.org/weblog/stories/2003/01/09/projectionKeyboards.html).

[4]. Baxes, G. A. "Digital Image Processing – Principles and Applications," America, 1994.

[5]. Gonzalez, R. and Woods, R.,"Digital Image Processing,," India, 2002.

[6]. Wijewantha N. S.," VistaKey: A Keyboard Without A Keyboard – A Type Of Virtual Keyboard," **Final***year project thesis*, Informatics Institute of Technology, Wellawatta, Sri Lanka, April 2004.

[7]. Hirsh, L., "Virtual Keyboard Replicates Real Typing," Oct. 23, 2003 (http://www.wirelessnewsfactor.com/perl/story/14762.html).

[8]. Wrolstad, J., "Laser Device Projects Full-Size Keyboard for Handhelds," Oct. 28, 2003 (http://www.wirelessnewsfactor.com/perl/story/17756.html).

[9]. Khar, A., "Virtual Typing," Oct. 28, 2003 (http://www.pcquest.com/content/technology/103030402.asp).

[1] [10].Aid Inc, "LightKey Visualize A Virtual Keyboard. One With No Moving Parts," Nov. 12, 2003 (http://www.advancedinput.com/AIDpdfDownloads/AIDLightKey.pdf)..