

A GUI FOR PROJECTOR-CAMERA USING TOUCH DETECTION

K.Vinod¹, P.Dileep Kumar²

¹PG Student, Department of ECE, Sri Venkateswara College of Engineering, Tirupati, Andhra Pradesh, India.
vinu.vinod59@gmail.com

²Assistant Professor, Department of ECE, Sri Venkateswara College of Engineering, Tirupati, Andhra Pradesh, India. dileepk38@gmail.com

ABSTRACT

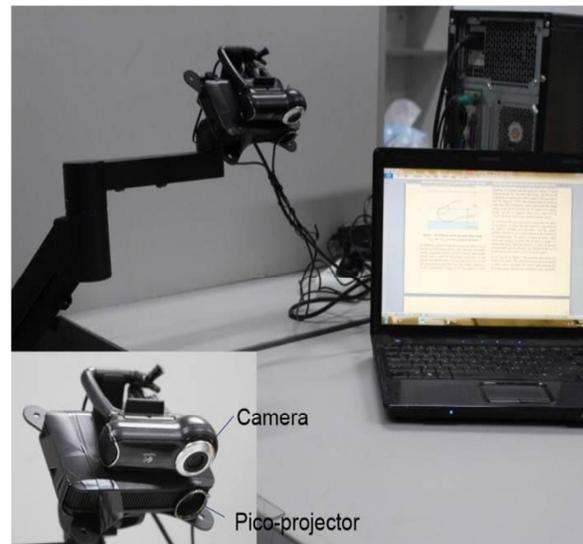
In this paper, we propose a novel interactive projection system (IPS), which enables bare-finger touch interaction on regular planar surfaces (e.g., walls, tables), with only one standard camera and one projector. The challenge of bare-finger touch detection is recovering the touching information just from the 2-D image captured by the camera. In our method, the graphical user interface (GUI) button is projected on the surface and is distorted by the finger when clicking it, and there is a significant positive correlation between the button's distortion and the finger's height to the surface. Therefore, we propose a novel, fast, and robust algorithm, which takes advantage of the button's distortion to detect the touch action. The existing keyboards used keys based keyboard for typing on the computer. These keyboards are working on the mechanical push principle. But for the small devices like mobile phones and tablets it is impossible to carry big keyboard with them. The touch screen based keyboards available in such devices are very inconvenient to write because the size of people finger is big and the size of the keys on the touch screen is small. So typing work on the small devices is not convenient and on computer our fingers get pain after doing long time typing work because of mechanical vibration of the keys.

Keywords—Edge detection, human-computer interaction, projector-camera system, touch detection, triangulation.

I. INTRODUCTION

Mobile devices (e.g., mobile phones, pads) with significant computational power and capabilities have been a part of our daily life. Benefiting from the small size of these devices, they are easy to carry.

However, the screen real estate of today's mobile devices is limited by their small sizes. This greatly diminishes their usability, functionality, and comfort. A pico-projector can be used to significantly increase the limited screen size of the mobile devices. With the development of the projection technology, we believe that embedded projectors in the mobile phones will be very common in the future, and people will enjoy a way of displaying digital contents on everyday surfaces.



Meanwhile, the interactions (e.g., touch, gesture) on the projected display are thought to be appealing. To achieve the touch interaction, the biggest challenge lies in how to determine whether the fingers touch the projected surface or not. Most of the researchers in this area use multicameras or a depth camera to obtain the relative position between the fingertip and the projected surface.

II. PROPOSED SYSTEM

In the proposed method, we propose an interactive projection system (IPS), which enables bare-finger touch interaction on regular planar surfaces (e.g., walls, tables), with only one standard camera and one projector. The challenge of bare-finger touch detection is recovering the touching information just from the 2-D image captured by the camera. In our method, the graphical user interface (GUI) button is projected on the surface and is distorted by the finger when clicking it, and there is a significant positive correlation between the button's distortion and the finger's height to the surface. Therefore, we propose a novel, fast, and robust algorithm, which takes advantage of the button's distortion to detect the touch action. We design a hardware system on interactive projection system.

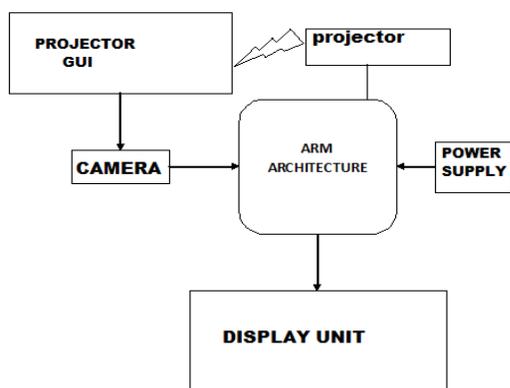


Fig1: System Framework

Our system is designed by using ARM 32-bit micro controller which supports different features and algorithms for the development of automotive systems. Here the camera and projector are connected to ARM controller. We are projecting a GUI on surface by projector and camera for capturing GUI, The camera will capture the places where user put his finger and the movement of the finger. The camera capture images are analyzed by the algorithms and programs present in the ARM microcontroller and then detect exactly which key pressed by user. The only requirement is that the face of the camera should be in the direction of projection.

III. HARDWARE IMPLEMENTATION

A. RASPBERRY PI BOARD



The **Raspberry Pi** is a credit-card-sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools. The Raspberry Pi is manufactured in two board configurations through licensed manufacturing deals with Newark element14 (Premier Farnell), RS Components and Egoman. These companies sell the Raspberry Pi online. Egoman produces a version for distribution solely in China and Taiwan, which can be distinguished from other Pis by their red coloring and lack of FCC/CE marks. The hardware is the same across all manufacturers. The Raspberry Pi has a Broadcom BCM2835 system on a chip (SoC), which includes an ARM1176JZF-S 700 MHz processor, Video Core IV GPU, and was originally shipped with 256 megabytes of RAM, later upgraded to 512 MB. It does not include a built-in hard disk or solid-state drive, but uses an SD card for booting and persistent storage.

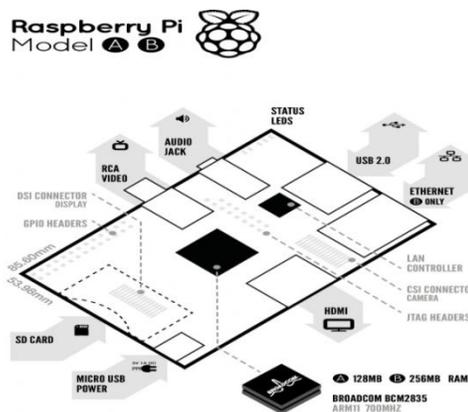


Fig2. Board features

The Foundation provides Debian and Arch Linux ARM distributions for download. Tools are available for Python as the main programming language, with support for BBC BASIC (via the RISC OS image or the Brandy Basic clone for Linux), C, Java and Perl.

B. TFT display unit

TFT stands for Thin Film Transistor, and is a type of technology used to improve the image quality of an LCD. Each pixel on a TFT-LCD has its own transistor on the glass itself, which offers more control over the images and colors that it renders.

While TFT-LCDs can deliver sharp images, they also tend to offer relatively poor viewing angles, meaning they look best when viewed head-on. If you view a TFT-LCD from the side, it can be difficult to see. TFT-LCDs also consume more power than other types of cell phone displays.

C. UVC Driver Camera:



Fig6. UVC Driver Camera

A UVC (or Universal Video Class) driver is a USB-category driver. A driver enables a device, such as your webcam, to communicate with your computer's operating system. And USB (or Universal Serial Bus) is a common type of connection that allows for high-speed data transfer. Most current operating systems support UVC. Although UVC is a relatively new format, it is quickly becoming common.

There are two kinds of webcam drivers:

1. The one included with the installation disc that came with your product. For your webcam to work properly, this driver requires some time to install. It is specifically tuned for your webcam, designed by your webcam manufacturer and optimized for webcam performance.

2. A UVC driver:-You can only use one driver at a time, but either one will allow you to use your webcam with various applications.

The following Logitech webcams support UVC:
 Logitech® QuickCam® Pro 9000 for Business, Logitech® QuickCam® Pro for Notebooks Business, Logitech® QuickCam® Communicate MP for Business, Logitech® QuickCam® Deluxe for Notebooks Business, Logitech® QuickCam® 3000 for Business.

IV. SOFTWARE REQUIREMENTS

A. Linux Operating System:

Linux or GNU/Linux is a free and open source software operating system for computers. The operating system is a collection of the basic instructions that tell the electronic parts of the computer what to do and how to work. Free and open source software (FOSS) means that everyone has the freedom to use it, see how it works, and changes it. There is a lot of software for Linux, and since Linux is free software it means that none of the software will put any license restrictions on users. This is one of the reasons why many people like to use Linux.

A Linux-based system is a modular Unix-like operating system. It derives much of its basic design from principles established in UNIX during the 1970s and 1980s. Such a system uses a monolithic kernel, the Linux kernel, which handles process control, networking, and peripheral and file system access. Device drivers are either integrated directly with the kernel or added as modules loaded while the system is running.

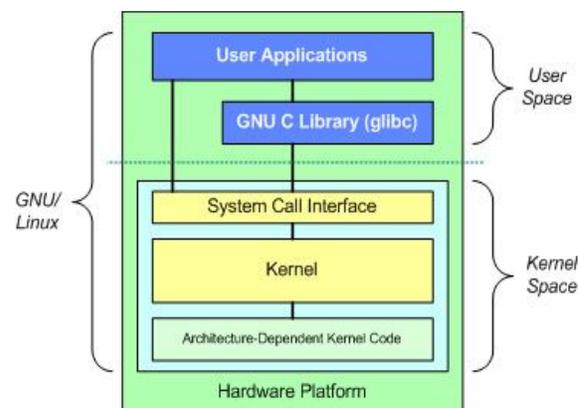


Fig7. Architecture of Linux Operating System

B. Qt for Embedded linux:

Qt is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases Qt is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers. Qt uses standard C++ but makes extensive use of a special code generator (called the Meta Object Compiler, or moc) together with several macros to enrich the language. Qt can also be used in several other programming languages via language bindings. It runs on the major desktop platforms and some of the mobile platforms. Non-GUI features include SQL database access, XML parsing, thread management, network support, and a unified cross-platform application programming interface for file handling. It has extensive internationalization support.

C. Open CV:

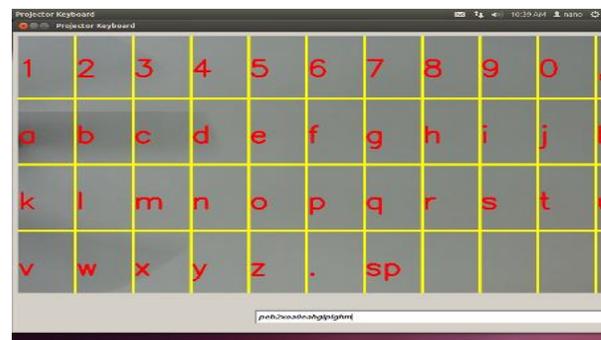
Open CV (Open Source Computer Vision) is a library of programming functions for real time computer vision. It is developed by Willow Garage, which is also the organization behind the famous Robot Operating System (ROS). Now you'd say MATLAB also can do Image Processing, then why open CV? Stated below are some differences between both. Once you go through them, you can decide for yourself. Advantages of OpenCV over MATLAB (Collected from various blogs/forums):-

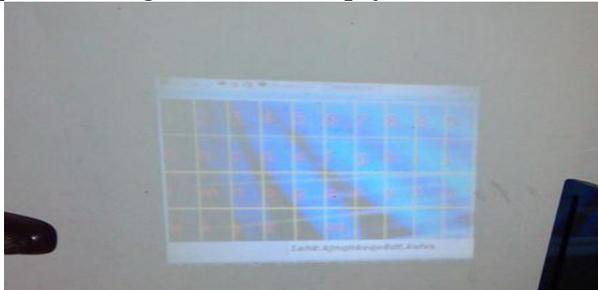
- **Speed:** Matlab is built on Java, and Java is built upon C. So when you run a Matlab program, your computer is busy trying to interpret all that Matlab code. Then it turns it into Java, and then finally executes the code. Open CV on the other hand, is basically a library of functions written in C/C++. You are closer to directly provide machine language code to the computer to get executed. So ultimately you get more image processing done for your computers processing cycles, and not more interpreting. As a result of this, programs written in Open CV run much faster than similar programs written in Matlab. So, conclusion? Open CV is damn fast when it comes to speed of execution. For example, we might write a small program to detect people's smiles in a sequence of video

frames. In Matlab, we would typically get 3-4 frames analysed per second. In Open CV, we would get at least 30 frames per second, resulting in real-time detection.

- **Resources needed:** Due to the high level nature of Matlab, it uses a lot of your systems resources. And I mean A LOT! Matlab code requires over a gig of RAM to run through video. In comparison, typical Open CV programs only require ~70mb of RAM to run in real-time. The difference as you can easily see is HUGE!
- **Cost:** List price for the base (no toolboxes) MATLAB (commercial, single user License) is around USD 2150. Open CV (BSD license) is free!
- **Portability:** MATLAB and Open CV run equally well on Windows, Linux and Mac OS. However, when it comes to Open CV, any device that can run C, can, in all probability, run Open CV.

V. RESULTS





VI. CONCLUSION

The project “A GUI for Projector-Camera using Touch Detection” has been successfully designed and tested. It has been developed by integrating features of all the hardware components and software used. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit. Secondly, using highly advanced ARM board and with the help of growing technology the project has been successfully implemented.

VII. REFERENCES

- [1] M. Khalilbeigi, R. Lissermann, M. M'uhlh"ausen, and J. Steimle, “Xpaaand: **Interaction techniques for rollable displays**,” in *Proc. ACM CHI*, 2011, pp. 2729–2732.
- [2] L. G. Cowan and K. A. Li, “**ShadowPuppets: Supporting collocated interaction with mobile projector phones using hand shadows**,” in *Proc. ACM CHI*, 2011, pp. 2707–2716.
- [3] Z. Mo, J. P. Lewis, and U. Neumann, “**SmartCanvas: A gesturedriven intelligent drawing desk system**,” in *Proc. ACM IUI*, 2005, pp. 239–243.
- [4] H. Benko and A. Wilson, “**DepthTouch: Using depth-sensing camera to enable freehand**

interactions on and above the interactive surface,” in *Proc. IEEE Workshop ITS*, vol. 8, 2009.

- [5] C. Harrison, H. Benko, and A. D. Wilson, “**OmniTouch: Wearable multitouch interaction everywhere**,” in *Proc. ACM UIST*, 2011, pp. 441–450.
- [6] D. Scharstein and R. Szeliski, “**High-accuracy stereo depth maps using structured light**,” in *Proc. IEEE CVPR*, 2003, vol. 1, pp. 195–201.
- [7] C. R. Wren, Y. Ivanov, P. Beardsley, B. Kaneva, and S. Tanaka, “**Pokey: Interaction through covert structured light**,” in *Proc. IEEE Workshop TABLETOP*, 2008, pp. 185–188.
- [8] C. Harrison, D. Tan, and D. Morris, “**Skinput: Appropriating the body as an input surface**,” in *Proc. ACM CHI*, 2010, pp. 453–462.
- [9] S. K. Kane, D. Avrahami, J. O. Wobbrock, B. Harrison, A. D. Rea, M. Philipose, and A. LaMarca, “**Bonfire: A nomadic system for hybrid laptop-tabletop interaction**,” in *Proc. ACM UIST*, 2009, pp. 129–138.
- [10] A. D. Wilson, “**PlayAnywhere: A compact interactive table top projection-vision system**,” in *Proc. ACM UIST*, 2005, pp. 83–92.
- [11] L.-W. Chan, H.-T. Wu, H.-S. Kao, J.-C. Ko, H.-R. Lin, M. Y. Chen, J. Hsu, and Y.-P. Hung, “**Enabling beyond-surface interactions for interactive surface with an invisible projection**,” in *Proc. ACM UIST*, 2010, pp. 263–272.
- [12] A. D. Wilson, “**Using a depth camera as a touch sensor**,” in *Proc. ACM ITS*, 2010, pp. 69–72.
- [13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA, USA: Addison-Wesley, 1992.
- [14] Optoma Ltd., [Online]. Available: <http://www.optomasa.com/products/search/pk301>
- [15] Logitech Ltd., [Online]. Available: <http://www.logitech.com/enus/search?q=pro+9000&isSupport=1>