

A Low Cost Digital Pen Based on Raspberry PI

Dr. M. Sudhakar¹, Dr. Vandana Khare² and Vijay Krishna Kanth³

¹CMR College of Engineering & Technology, Hyderabad, India.

¹Orcid Id: 0000-0002-2100-4196

²CMR College of Engineering & Technology, Hyderabad, India.

²Orcid Id: 0000-0003-4959-0657.

³PG Scholar, Department of ECE, CMR College of Engineering & Technology, Hyderabad, India.

³Orcid Id: 0000-0001-7696-5683

Abstract

The digital pen embedded system consists of a MEMS tri-axial accelerometer connected to raspberry pi through a comparator. The tri-axial accelerometer based digital pen measures the acceleration signals of user's hand motions and the corresponding trace is displayed on the monitor. As the display is 2-dimensional, only two accelerometers are used in this application. The comparator compares the outputs of the accelerometers to evaluate the direction of the cursor. Dynamic Time Warping based recognition algorithm for handwriting and gesture recognition is used in this application. The color of the display on the monitor can be varied in the program. The code for the application program in raspberry pi is written using Python programming language. The other software used in the application are Jessie Raspbian and OpenCV. The acceleration settings can be varied to get suitable sensitivity. This is a low cost application. The application is useful for presentations in conferences or classrooms.

Keywords: Component; Digital Pen, Tri-axial accelerometer, Raspberry Pi, OpenCV,

INTRODUCTION

With the rapid development of computer technology, contemporary Human-Computer Interaction (HCI) devices/techniques [1] [2] have become indispensable in our daily life. HCI devices/techniques such as computers, consumer electronics, mobile devices, etc., have also dramatically altered our living habits. The ease with which an HCI device or technique can be understood and operated by users has become one of the major considerations while selecting such a device. Therefore, it is necessary for researchers to develop advanced and user-friendly HCI

technologies which are able to effortlessly translate users' intentions into corresponding commands without requiring users to learn or accommodate to the device. Technologies are being developed which are able to intuitively express user's intentions, such as handwriting, gestures, and human body language, to naturally control HCI devices. These technologies have many applications in the fields of remote control, virtual reality and sign language.

Current trends in human machine interface have increased rapidly in respect of consumer devices such as multi-touch of iPhone, motion sensing devices of Wi-Fi, and etc. However, the hand tracking systems as user input have been limited due to the technical constraints and cost. In this paper, we introduce a hand motion capture system using a MEMS Accelerometer (Micro Electrical and Mechanical Sensor) that enable to track 2D hand pose [3] and interact with applications in real-time. Our goal is aimed to design the system for consumers with a low-cost approach while having to deal with the robustness, precision, and real-time constraints.

SYSTEM PROPOSED

The system proposed in the paper has following components:

Hardware:

- MEMS 3-Axis Accelerometer
- Comparator
- Raspberry Pi 3 Model B
- SD Card with 8 GB memory
- Monitor

Software

- Jessie Raspbian (Operating Systems)
- OpenCV (Application Software)
- Python (Programming Language)

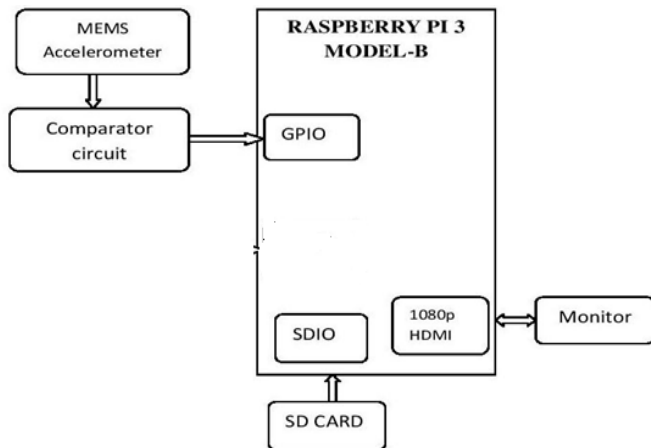


Figure 1: System Block Diagram

An Inertial-Sensor [1] [8]-based Digital Pen [9] with MEMS and its associated Dynamic Time Warping (DTW) [1]-based recognition algorithm has been used for handwriting and gesture recognition [10]. Users hold the inertial pen to write numerals, alphabets or drawings by making hand gestures with their preferred handheld style and speed. The inertial signals generated by hand motions [11] are processed in the comparator and transmitted to raspberry pi for further processing and display on the monitor. The proposed DTW-based recognition algorithm includes the procedures of inertial signal acquisition, signal pre-processing, motion detection [12], template selection, and recognition.

RASPBERRY PI CORE MODULE

The core module of the system is realized using a Raspberry Pi 3 board; it's a \$ 35 bare-bones computer designed and developed by the Raspberry Pi Foundation, the Pi 3 features a BCM 2837 System-on-Chip[13] which includes a Quad-Core 64-Bit ARM Cortex A7 CPU clocked at 1.2 GHz paired with 1 GB of RAM. It also has Video Core IV GPU for graphical processing applications, it also includes four USB ports for peripherals and 40 Pin General Purpose Input Output (GPIO) pins for interfacing the Pi with external electronic circuits, these GPIO pins are used to interface the Pi to the door lock module. The Raspberry Pi is designed to run various Linux based operating systems and has Raspbian as its official operating system and Python as its official programming language.

MEMS ACCELEROMETER

MEMS based Accelerometers [1] [10] are one of the simplest but also most applicable micro-electromechanical systems. They became indispensable in automobile industry, computer and audio-video technology. This seminar presents MEMS technology as a highly developing industry. An accelerometer is an electromechanical device that measures acceleration forces. These forces may be static, like the constant force of gravity pulling at our feet, or they could be dynamic - caused by moving or vibrating the accelerometer [8].

In the paper MEMS Accelerometer is used to draw the picture, alphabets and numerical on the terminal window of the monitor by controlling user hand motions. In this application, we have used only two accelerometers because the 2-D display. The MEMS Accelerometer generates analog voltage based on its orientation along X-axis and Y-axis. The output is fed to the comparator circuit.



Figure 2: MEMS Accelerometer

COMPARATOR

Comparator used here is LM358 I.C. This is connected to the MEMS accelerometer sensor. It compares MEMS X-direction and Y-direction analog voltages so as to decide the direction of the trace. We have used two Comparators with variable resistors. This operation is performed by assigning the GPIO pins to comparator output pin. The cursor draws by one pixel position along X-direction or Y-direction based on the output from the MEMS accelerometer. This activity of pixel increment or decrement is carried out by calling the openCV module in python programming. The detailed algorithm of the comparator is explained in Figure 4.



Figure 3: LM 358 Board

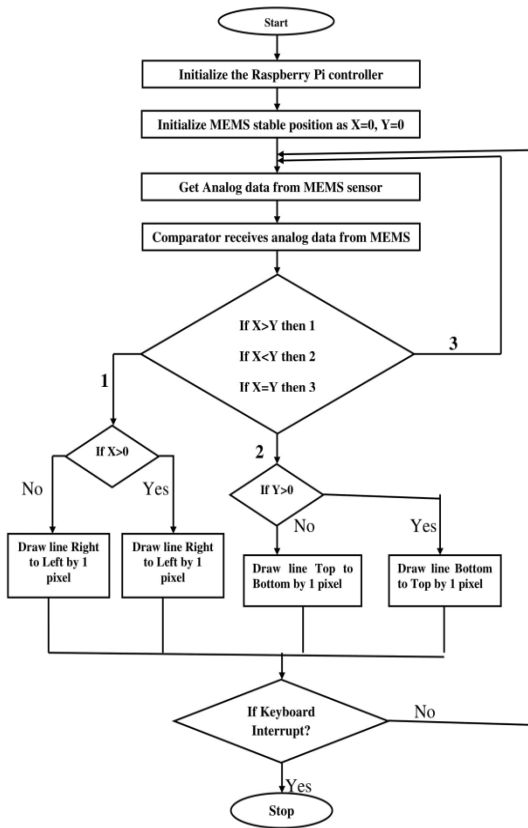


Figure 4: MEMS Accelerometer Comparator Algorithm

VISUAL TRACKING

A. Trackers using Color Information:

As a fundamental problem in vision, visual tracking has been drawing research attention for decades. A comprehensive review of the topic can be found. Since our focus is on integrating color information in tracking [2], we review only previous color trackers due to space limitation. Table- I below lists the abbreviations of trackers discussed in this paper. A notable early work on color tracking is the color particle filter introduced, which calculates the likelihood of each particle by comparing its color histogram from the HSV (Hue, Saturation, and Value) color space with the reference color model. In the target model and target candidates are represented by smoothed color histograms quantized from the RGB color space, and mean shift is used to minimize the distance between the discrete distributions of the target model and target candidates. In RGB color distribution was used to describe the target model and candidates, and the target object was located by minimizing the Kullback Leibler distance [4] between the color distributions of the target model and candidates with the help of a trust-region method. VTD integrates basic trackers derived from the combination of different basic observation and motion models, and four basic observation models, which use hue, saturation, intensity and edge templates as features respectively are adopted.

Table 1: Abbreviation of Trackers

Tracker	Brief description
CSK	Circulant Structure of tracking-by-detection with Kernel
LOT	Locally Orderless Tracking
MEEM	Multi-Expert restoration using Entropy Minimization
VTD	Visual Tracking Decomposition
VTS	Tracking by sampling trackers

LOT measures the similarity between a candidate and the target using locally orderless matching, and HSV color space is used to describe the appearance of each pixel. MEEM[5] uses features extracted in the LAB color space. In the most recent work, CSK [6] is extended with color names, and to speed up, the dimension of the original color names is reduced with an adaptive dimensionality reduction technique. There are also trackers that take color input, but do not explicitly exploit the use of color information. Despite previous arts, there is a lack of a systematic study and understanding of how color information can be used to improve visual tracking. Our work aims to fill the gap by thoroughly investigating the behavior of numerous state-of -the-art visual trackers with various color representations.

In the image shown below, each small box represents a pixel of the image. In real images, these pixels are so small that human eye cannot differentiate. Usually, one can think that BGR color space is more suitable for color based segmentation. But [HSV color space](#) is the most suitable color space for color based image segmentation.

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	255	255	0	0	0
0	0	255	255	255	255	0	0
0	0	255	255	255	255	0	0
0	0	0	255	255	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

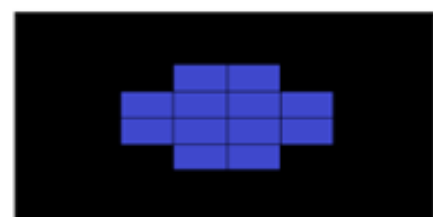


Figure 5: Blue Matrix

255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255
255	255	255	0	0	255	255	255
255	255	0	0	0	0	255	255
255	255	0	0	0	0	255	255
255	255	255	0	0	255	255	255
255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255



Figure 6: Green Matrix

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	255	255	0	0	0
0	0	255	255	255	255	0	0
0	0	255	255	255	255	0	0
0	0	0	255	255	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0



Figure 7: Red Matrix



Figure 8: Resultant Image

HSV color space is also consists of 3 matrices, HUE, SATURATION and VALUE. In OpenCV, value range for HUE, SATURATION and VALUE are respectively 0-179, 0-255 and 0-255. HUE represents color, SATURATION represents the amount to which that respective color is mixed with white and VALUE represents the amount to which that respective color is mixed with black.

In the above application, we have considered that the red object has HUE, SATURATION and VALUE in between 170-180, 160-255 and 60-255 respectively. Here the HUE is unique for that specific color distribution of that object.

SATURATION and VALUE may vary according to the lighting condition of the environment.

Hue values of basic colors:

- Orange 0-22
- Yellow 22- 38
- Green 38-75
- Blue 75-130
- Violet 130-160
- Red 160-179

After thresholding the image, you'll see small white isolated objects here and there. It may be because of noises in the image or the actual small objects which have the same color as our main object. These unnecessary small white patches can be eliminated by applying 'morphological opening'. 'Morphological opening' can be achieved by erosion, followed by the dilation with the same structuring element.

Threshold image may also have small white holes in the main objects here and there. It may be because of noises in the image. These unnecessary small holes in the main object can be eliminated by morphological closing. Morphological closing can be achieved by dilation, followed by the erosion with the same structuring element.

B. Color Information in Other Vision Tasks:

Not surprisingly, the discriminative power of color information has been systematically investigated for various vision topics [7], such as object recognition, human action recognition, object detection, etc. While being highly motivated by these pioneering works and borrowing some ideas from them, our work however focuses on visual tracking. To the best of our knowledge, this is the first comprehensive study on encoding color information for visual tracking. In fact, as shown in our experiments, many modern grayscale trackers, when augmented with color information, outperform previously proposed color trackers.

RESULTS & EVALUATION

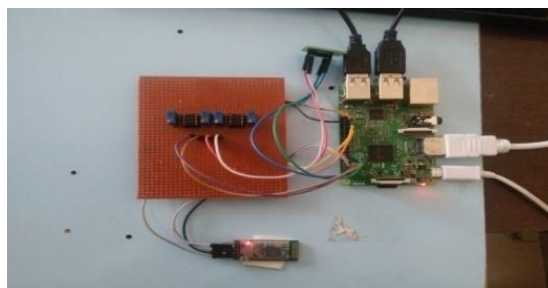


Figure 9: Embedded System

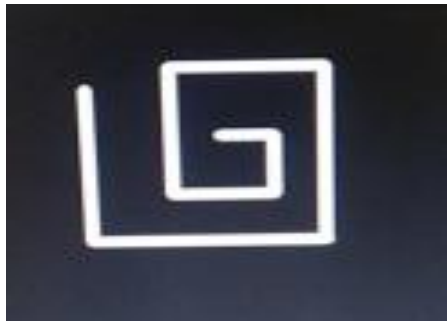
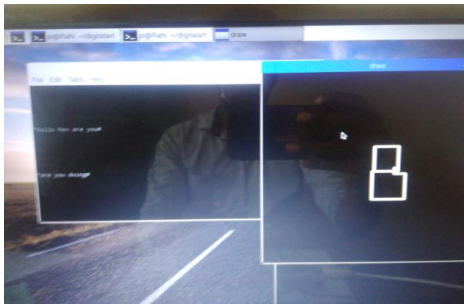


Figure 10: Digital Pen Hand Motion Display

CONCLUSION

A low cost Digital Pen application has been successfully implemented using Raspberry Pi. The MEMS based Accelerometer is integrated to the Raspberry Pi controller through a comparator. The hand motions have been successfully converted in the 2-dimensional trace and displayed on the monitor connected to the Raspberry Pi. The programming is done in python. OpenCV software has been used for image processing for display purpose.

REFERENCES

- [1] K. Altun, B. Barshan, and O. Tunçel, "Comparative study on classifying human activities with miniature inertial and magnetic sensors," *Pattern Recognit.*, vol. 43, no. 10, pp. 3605–3620, 2010.
- [2] Card, Stuart K.; Thomas P. Moran; Allen Newell (July 1980). "The keystroke-level model for user performance time with interactive systems". *Communications of the ACM*. 23 (7): 396–410.
- [3] M. H. Ko, G. West, S. Venkatesh, and M. Kumar, "Using dynamic time warping for online temporal fusion in multisensor systems," *Inform. Fusion*, vol. 9, no. 3, pp. 370–388, 2010.
- [4] Kullback, S.; Leibler, R.A. (1951). "On information and sufficiency". *Annals of Mathematical Statistics*. 22 (1): 79–86.
- [5] J. Zhang, S. Ma, and S. Sclaroff, "MEEM: robust tracking via multiple experts using entropy minimization," *Proc. European Conf. Computer*

Vision (2014), 2014.

- [6] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," *Proc. European Conf. Computer Vision*, pp. 702–715, 2012.
- [7] M. J. Swain and D. H. Ballard, "Color indexing", *International Journal of Computer Vision*, vol. 7:1, pp. 11-32, 1991.
- [8] W. C. Bang, W. Chang, K. H. Kang, E. S. Choi, A. Potanin, and D. Y. Kim, "Self-contained spatial input device for wearable computers," in *Proc. IEEE Int. Conf. Wearable Comput.*, Oct. 2003, pp. 26–34.
- [9] A. Akl, C. Feng, and S. Valaee, "A novel accelerometer-based gesture recognition system," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6197–6205, Dec. 2011.
- [10] R. Xu, S. Zhou, and W. J. Li, "MEMS accelerometer based non specific user-hand gesture recognition," *IEEE Sensors J.*, vol. 12, no. 5, pp. 1166–1173, May 2012.
- [11] C. M. N. Brigante, N. Abbate, A. Basile, A. C. Faulisi, and S. Sessa, "Towards miniaturization of a MEMS-based wearable motion capture system," *IEEE Trans. Ind. Electron.*, vol. 58, no. 8, pp. 3234–3241, Aug. 2011.
- [12] Y. S. Kim, B. S. Soh, and S. G. Lee, "A new wearable input device: SCURRY," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1490–1499, Dec. 2005.
- [13] <http://www.raspberrypi.org/archives/3195>