FPGA AND COMPUTER VISION BASED ON VIRTUAL LED MATRIX

Vedanti Gabhane[*1], Prathamesh Gadhave[*2], Prajwal Ramteke[*3], Ayan Sheikh[*4], Dr. P.B Pokle[*5]

[*1,2,3,4] *UG Student, Department Of Electronics & Telecommunication Engineering, Priyadarshini J.L. College Of Engineering Nagpur, Maharashtra, India.*

[*5] *Asst. Professor, Department Of Electronics & Telecommunication Engineering, Nagpur, Maharashtra, India.*

ABSTRACT

The main goal of computer vision is to identify and recognize different objects of various sizes, shapes, and positions. the major problems faced by computer vision are the illumination and the viewpoint of the object, concerning multiple studies on detecting and recognizing objects that showed a high level of accuracy and precision on these tasks. to facilitate object detection online the proposed work allows the user to track the movement of any cultured object of his/her choice. the user can even choose the colors of his choice to be displayed. by running the application, the camera is activated thus enabling the user to draw in the air just by waving the tracker object. the drawing is also simultaneously visible on the white window. The instructor can choose any color of his choice displayed above to draw and also can clear the screen when needed. we will be using the computer vision techniques of OpenCV and python to build this application.

Keywords*: FPGA, LED matrix, Hand Gesture, Camera, etc.*

1. INTRODUCTION:

Virtual led matrix is a computer vision fpga based embedded system project which consists of a 16x16 led matrix which can be controlled over usb-serial communication ttl interface it uses alteras cyclone ii ep2c5t144c8 fpga which drives all 16x16 leds in the matrix using multiplexing with the help of a standard usb-to-ttl converter fpga is able to communicate with the host computer application in order to update data on the led matrix the fpga is able to perform multiple tasks such as buffering data from usb-ttl checking for external interrupts updating led matrix driving indicator leds etc all at once with the help of fsm concurrent architecture implementation in vhdl apart from fpga hardware interface this system also consists of a host laptop which is used to perform all computer vision cv and digital image processing dip tasks in real-time the computer vision algorithm involves various sub algorithms and steps which need lots of processing power memory resources which is not possible to implement on small devices like a microcontroller or low-cost fpgacpld devices the host uses python as an application platform in which a computer vision algorithm is implemented this algorithm goes through various steps such as capturing images from the cameracapture devices pre-processing captured frames colour domain conversion colour masking thresholding contour detection contour approximation bounding rectangle tracking and mapping position of contour and sending data packets over usb-ttl device.

2. LITERATURE REVIEW:

1) Hand Gestures Recognition And Device Control: Author:- Jayashree Katkar, Omkar Kahane, Vivek Jadhav, Pratik Jadhav Department of Electronics and Telecommunication, Mumbai University K.C.College of Engineering & Management Studies & Research, Thane(E). Published:- International Journal Of Engineering Technology Science and Research, In April 2017. The hand gesture system is more popular these days since it facilitates human-machine connection. They utilized the hand for recognition in this work because it provides more significant information than other types of gestures. The principal component of the hand gesture recognition system includes the segmentation and tracking of hands from the background and then the feature is extracted from the segmented hand image using Finally, we'll recognize hand motions using multiple methods. Many applications use hand

gesture recognition, including HCI, robotics, sign language, and digit and alphanumeric value. They used the Kinect sensor in this study, which is extensively used in comparison to vision-based technology and glove-based methods. In comparison to static hand gestures, dynamic hand gesture recognition requires more computation.

2) Handwritten Text Recognition Using Machine Learning Techniques in Application of NLP: Author:- Polaih Bojja, Naga Sai Satya, Tej Velpuri Department of Electronics and Communication Engineering, Koneru Lakshmaiah Educational Foundation. In December 2019, the International Journal of Innovative Technology and Exploring Engineering was published. This paper shows Handwritten character recognition is a complex problem, which is not easily solvable. The necessity is around dataset and database. This model analyses the text we've produced and converts it to Computer Text and Voice forms. This application has a wide range of applications in the healthcare and consumer sectors. This type of paradigm, which is employed in health applications, may save people's viewpoints and digitally retain each and every record.

3) An Inertial Pen With Dynamic Time Warping Recognizer for Handwriting and Gesture Recognition: Author:- Yu-Liang Hsu, Cheng-Ling Chu, Yi-Ju Tsai National Chiao Tung University, Hsinchu, Taiwan. Published:- Institute of Electronics and Electrical Engineering, In January 2015. This paper has presented an inertial pen with a systematic time alignment algorithm framework for inertial-sensing-based handwriting and gesture recognition. The proposedDTW-based recognition algorithm consists of inertial signal acquisition, signal preprocessing, motion detection, template selection, and recognition. To obtain better movement signals, we have utilized a quaternion-based complementary filter to reduce orientation errors and the ZVC method so as to minimize the undesirable error accumulation of velocity signals. Subsequently, to improve the performance of the DTW recognizer, all movement signals are normalized via the Z-score method, and the class template is selected via the proposed Min-Max template selection method To test the effectiveness of the proposed inertial pen and algorithm, 2D handwritten digits, 3D handwritten digits, 2D handwritten English lowercase letters, 2D handwritten digits, and English letters, and 3D hand motions were collected during experimental validation. The above-mentioned tests had user-independent recognition rates of 97.9%, 87.3 percent, 92.0 percent, 92.1 percent, and 98.1 percent, respectively. Furthermore, the studies' user-dependent recognition rates were 99.4 percent, 94.6 percent, 94.3 percent, 93.0 percent, and 99.8 percent, respectively. We believe that the inertial pen and its accompanying DTW-based recognition algorithm can be deemed a novel and effective HCI gadget based on the above testing results.

3. RESEARCH METHODOLOGY:

The Virtual LED Matrix is built around Altera's Cyclone ii FPGA (EP2C5T144C8). As shown in the circuit diagram, this system consists of various passive and active electronic components such as resistors, electrolytic capacitors, ceramic capacitors, transistors, etc. These components are connected as per the circuit designed. The circuit is having one power supply unit which consists of a step-down transformer to convert 220v AC into lower 12v AC. This secondary voltage from transformer the is then fed into a full-bridge PN-junction diode-based rectifier (D1, D2, D3, D4) which converts 12v AC into 12v pulsating DC. This 12v pulsated DC is then converted into a smooth 12v DC with the help of a filter electrolytic capacitor of 1000uF (C1). After filtering this 12v DC (with small fluctuations)fed feed into a 5v voltage regulator IC LM7805 (U1) which converts higher DC input voltage into lower 5v DC output voltage. To further filter the output of regulator IC, a 470uF electrolytic capacitor is connected (C2). For power supply indication, a single red color LED is connected between 5v and GND via one 1.8k resistor (R33). This 5v is then used by low low-power-end components as such FPGA, transistors, LEDs, etc. The 16x16 LED matrix is connected to FPGA through the h driver circuit. This matrix is having 16 rows and 16 cols which are driven by FPGA and circuits. All 16 rows (anodes of the matrix connected to FPGA's standard output pins via 16x 100 Ohm current control resistors (R17-R32). To drive all columns of the LED matrix, 16 NPN type BJT (Bipolar Junction Transistor) BC547 (Q1-Q16)was accessed. These BJTs are configured in Common Emitter (CE) configuration which drives all cathodes of LEDs in cols. The base of the transitions is connected through 16x 1KOhm resistors which are connected & controlled through digital outputs of FPGA. For communication with a computer vision application running a host computer, a USB-to-TTL (UART) converter is used. This converter is connected a to computer/laptop throat ugh USB cable as a serial communication device. Another end of this USB-to-TTL converter is connected to FPGA via two digital pins ACTH acts as UART communication protocol pins (Rx, Tx). This system also consists of some tactile switches (push-to-on) for reset, and matrix operations (such as clear matrix, set matrix, etc.) purpose. The FPGA is programmed/configured to handle tasks such as buffering serial UART (Universal Asynchronous Receiver & Transmitter) data on Rx and Tx lines and also controlling 16x16 LED matrix parallelly using concurrent architecture design. The digital circuit design and synthesis are done in VHDL language on Quartus II Web Edition software tool for Altera’s FPGA. For debugging, analysis, and test purposes, a pulse view logic analyzer and serial monitor/terminal are used.

The computer vision algorithm in the host computer is designed to detect and track a particular colour object to draw things virtually. It is implemented on a python programming platform with various high-level APIs/libraries such as PYserial, OpenCV, etc. This application captures live frames from a hardware capture device such as an inbuilt camera of a laptop or any USB webcam to perform various Digital Image Processing (DIP) algorithms on the captured frame to detect and track colour objects within the frame. As shown in the Computer Vision Algorithm block diagram, it goes through various algorithm steps. The very first step is to capture a valid frame the from camera then it gets flipped with the help of the flipping function in OpenCV. After flipping the captured image, it gets converted from RGB (Red Green Blue) into HSV (Hue Saturation Value) colour domain which is having more accuracy to extract or filter particular colours from images. This converted HSV domain image is then masked with some lower and upper HSV ranges to filter out unwanted colour from an image and only the ep ROI (Region of Interest) colour. The ROI HSV domain colour image is then converted into a luminance (luma) / Gray-scale image by only copying V (value) component the from ROI image. This converted grayscale is then fed then feed into a single thresholding algorithm which converts this grey-scale image into a binary/threshold/monochrome image that only consists of complete black or complete white pixels. This binary image is then processed through a contour detection algorithm which locates and bound rectangles and lines around blobs (ROI object). After locating the contour within the image, it applies contour approximation to detecting particular objects, etc (square, rectangle, circle, etc.). This algorithm only processes contours with large areas (for better accuracy). Once the contour is detected and located, a bounding rectangle is drawn around the contour. It gives x, y coordinates (position) & width and height of contour concerning frame dimensions. This x location is then used to draw lines and points on the GUI (Graphical User Interface) screen of the laptop or computer. Once the drawing pixel data is collected the python uses the PY serial library which allows a program to communicate with devices over serial communication interfaces such as USB-to-TTL converter hardware devices. The algorithm sends data packets consisting of pixel information to FPGA through this USB-TTL hardware interface device. As it uses UART which is an asynchronous serial communication so this USB-TTL device and FPGA are configured to buffer data at the same speed/baud rate as there is not any kind of synchronization available.

4. APPLICATION:

1) This system can be used in schools, colleges, offices, etc. for easy interaction without any need for physical contact with a drawing board or display.
2) It can also be used to track the location or draw the trajectory of any object.
3) In industry also this system can be used on an assembly line to detect and locate objects by color in order to control a robotic arm or manipulator without any human interaction.
4) In automobiles and robotics this system is very useful for applications such as autopilot, trajectory tracking, navigation system, traffic control, etc.

5. CONCLUSION:

We consider our project a worldwide success! With air, we have built a hands-free drawing program that uses computer vision to detect the user's object at the fingertip. The colour line can be drawn wherever the user wants. It's really like drawing in the air.

Of course, machine vision has many flaws that could be interesting areas of research in the future. The first is the issue of frame rate. Imager processing slowed down the camera. It would be the best multicore functionality, which we attempted in this project. If timing issues with queuing data between processes can be handled so that frame information is transmitted in order, we can upgrade our "virtual LED matrix and FPGA based computer vision" to work authentically in real-time.

6. REFERENCES:

[1] Hand Gesture Recognition And Device Control "Jayashree Katkar, Omkar Kahane", IJETSR (Volume 4), April 2017.

[2] Survey On Handwritten Character Recognition (HCR) Techniques For English Alphabets "Manoj Sonkusare and Narendra Sahu", Advance in Vision Computing: An International Journal (AVC) Vol.3, No.1, March 2016.

[3] A Gesture-Based Digital Art with Colour Coherence Vector Algorithm "S. Ravikumar, N. Harshini, D. Iswarya", IRJET, Volume 04, Issue 03, March 2017

[4] Review Of Classification Algorithms Used For Gesture Recognition Using Accelerometer "Sudiksha Khaduja, Sparsh Samir", IJPAM, Volume 118, No. 22, 2018.

[5] Hand Gesture Recognition: A Literature Review "Rafiqul Zaman Khan, Noor Adnan Ibraheem", IJAIA, Volume 3, No. 4, July 2012.

[6] Handwritten Text Recognition Using Machine Learning Techniques in Application of NLP "Polaih Bojja, Naga Sai Satya, Tej Velpuri", IJITEE, Volume 9, Issue 2, December 2019.

[7] An Inertial Pen With Dynamic Time Warping Recognizer for Handwriting and Gesture Recognition "Yu-Liang Hsu, Cheng-Ling Chu", IEEE Sensor Journal, Volume 15, No. 1, January 2015.

[8] Volume 5, Issue 1, January 2015 ISSN: 2277 128X, International Journal of Advanced Research in Computer Science and Software Engineering: Research Paper -- Gesture Controlled Computer.

[9] Volume 43, Issue 1, June 2012 ISSN: 2277 128X, International Journal of Advanced Research in Computer Science and Software Engineering: Research Paper -- Hand Data Glove: A Wearable Real-Time Device for Human-Computer Interaction

[10] International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.4, July International Journal of Scientific Research in Science and Technology (www.ijsrst.com) | Volume 5 | Issue 8 © 2020 IJSRST | Print ISSN: 2395-6011 | Online ISSN: 2395-602X | Published: 18/12/2020 - 189-194 194 2012, DOI: 10.5121/ijaia.2012.3412 161- HAND GESTURE RECOGNITION: A LITERATURE REVIEW

[11] OpenCV for Computer Vision Applications, Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15), March 20, 2015

[12] Real-Time Object Detection and Tracking Using Deep Learning and OpenCV Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA 2018) IEEE Xplore Compliant Part Number: CFP18N67-ART; ISBN:978-1-5386-2456-2

[13] Study on Object Detection using Open CV – Python, International Journal of Computer Applications (0975 – 8887) Volume 162 – No 8, March 2017

[14] Nidhi, "Image Processing and Object Detection", Dept. of Computer Applications, NIT, Kurukshetra, Haryana, 1(9): 396-399, 2015.

Comparison of OpenCV's Feature Detectors and Feature Matchers Frazer K. Noble Centre for Additive Manufacturing School of Engineering and Advanced Technology Massey University New Zealand.