



## **Gesture Zoom-Touchless Navigation at your Fingertips**

Valarmathi C<sup>1</sup>, Poornima N<sup>2</sup>, Nanditha M<sup>3</sup>, Yeshaswini R<sup>4</sup>, Vinay Kumar U<sup>5</sup>

<sup>1</sup> Assistant Professor, Department of Computer Science and Engineering, Sri Sairam College of Engineering, Anekal, Bangalore, vinmathi20@gmail.com

<sup>2,3,4,5</sup> UG students, Department of Computer Science and Engineering, Sri Sairam College of Engineering, Anekal, Bangalore.

### **ABSTRACT**

In this work, we introduce Gesture Zoom, a touch less navigation system that uses hand gestures to allow for intuitive visual manipulation. Users can zoom in and out of photos in real-time without touching a device thanks to Gesture Zoom, which uses Open CV and computer vision technologies to read hand gestures. Fingertip distance calculation, hand movement detection and tracking, and image position locking based on user gestures are all part of the process. Gesture Zoom is perfect for industries that need contact-free control, like healthcare, food service, and education, since it uses Python-based Open CV and the CV zone hand tracking module to offer responsive, fluid picture scaling and repositioning. The system's effectiveness in a range of environmental circumstances, responsiveness to scaling commands, and detection accuracy are all highlighted by experimental tests. According to our research, Gesture Zoom has potential for applications requiring cleanliness and precision, with further advancements targeted at reducing environmental sensitivity and improving gesture accuracy.

**Keywords:** Gestures, Open CV and touch less Navigation.

### **1 Introduction**

Nowadays, development of the human-computer interaction are very encouraging. Human computer are progressed in such a way that its interaction field has also opened a wide range opportunity for research and scholars to develop new technology. Microsoft Kinect, Play Station Move and Wii Remote are the tools have unfasten a new segment in this field. In present-day technology are towards interaction with machine using gestures.

Gesture Zoom uses advanced sensors and computer vision technology to detect even the smallest hand movements, it permits ultra- responsive and touch – free control. This technology is used in the several fields like hospitals, laboratories, teaching, and food service, it allows users to navigate without touching the surface. Research on Gesture will consume more time in the process, and it is a biggest task to researchers face. The gesture is based on the significant relationship with the operation. After the idea of touch-less interaction is introduced the Human body gestures are taken part. Specific tools are needed for each gesture to be received by the machine from the user. There are several ways that user can interact with the machines using gestures of voice, touch and hand movements. Gesture project is associated with the stimulation of the image using touch less navigation. It identified the movement of the fingers that is captured by the tools like Kinect or Camera and processed, the machine will determine the



gesture movement and performs the program over the image being displayed in the imager viewer. It depends on the motion of the fingers to zoom in and zoom out as the use needs.

## 2 Related Work

Gesture Zoom is a segment of the largest field of touch less and gesture-based navigation, which gained significant role in both industry and academia. Related work spans from hardware developments to software algorithm, and it integrates elements of computer vision, sensor technology, and artificial intelligence. The first widely available tool which is initially introduced for Xbox that is Microsoft Kinect. It uses the deepest sensors and vision of computer to track movement of the body and gestures, it enables users to control gaming without a controller. It especially design to tracking hands and body posture with in depth sensing technology. There are many other tools which are working on 3D space movements, The micro-geturrs, 3D visilization,.etc. Gesture Zoom it works using the Open CV (Open Source Computer Vision Library) in python. It is a software library that plays a significant part in gester based applications, including touch less navigation like Gesture Zoom. Open CV provides, features extraction, object recognition, and motion tracking, which are essential for interpreting hand gestures in real time. In this project we arranged the many images, we fixed the start and stop button respectively right and left to make users work easy, the users need click on the start button and select the image and user should use the his finger tips to zoom in and zoom out the image. The Open CV it capture the frame from a camera in real time and isolate the hand region. detect contours, convex hulls, and convexity defects to identify fingertips. Using the optical flow to track hand movement over time and implement the deep learning models to recognize gesture based on features. The control map recognized gestures to specific action on the devices (zoom in, zoom out). Open CV is various machine learning models make it an invaluable toolkit for developing gestures-based application like Gester-Zoom. Gesture Zoom is still facing challenges environmental sensitivity and need more standardized interfaces. It also includes future directions to improving gestures accuracy, minimizing sensor size and expanding into boader industries like healthcare, smart homes, logistics.

## 3 Methodology

The Implementing of hand tracking function in a python program using the Open CV and cv zone's Hand Tracking Module .In this project the users can start and stop hand tracking, select images using the curser, resize or reposition them based on gestures .The step-by-step breakdown of the methodology applied:

### 3.1 Setup and Initialization

Import libraries it import cv2 for Open CV and Hand Detection from CV zone for hand-tracking process. The Camera initialize the set up camera with a resolution of 1280x720. Using image loading two images are loaded and it is resized to 40 x 40 pixels. The Hand detector is used for hand detection it set detection Con=0.8 controls the detection confidence threshold it



only detect s registers hands if it's at least 80% confident. The find Hands ( ) method in cvzone which returns the list of detected hands and annotated frame. In each hand it consists of landmark (lmlist) which represent specific points (node) on the hand, such as the fingertips.

### 3.2 Button and Image Interaction

In button there is draw buttons() function which creates the "Start" and "Stop" buttons using the addition functions of Open CV and drawing rectangles. Buttons are placed at the bottom left and bottom right of the screen. The click event ( ) is responsible for simplify mouse clicks. The activation or deactivation to select the image is depends on the position of clicks. In top-right corner of the screen, the images are placed in predefined position, the user need to choose one image for manipulation.

### 3.3 Image Manipulation and Hand Tracking

When hand tracking is active, image is selected, it find out hands in the frame and calculate the distance between the index fingertips of hands.

The distance formula is obtained using detector. Find Distance ( )

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

- (X1, y1) and (x2, y2) are the coordinates of the index finger tips hand 1 and hand2, respectively.
- The image position is allowed to be fix at specific spot , when the fingers are close enough it locks the image position using the position locking formula is

$$\text{Img\_position} = (x1 - \frac{\text{img\_width}}{2}, y1 - \frac{\text{image\_height}}{2}) \quad (2)$$

- Img\_width and img\_height are the current dimention of the image and it centers the image around the index finger's position.

The detection of hand gesters represents the resized and position of the selected image. Scaling determined the proportion of the image size scales to the distance between the two hands. A minimum size is enforced to prevent excessive shrinking is d 40 x 40 pixels using the formula scaling formula.

$$\text{Scale\_factor} = \frac{\text{distance}}{200} \quad (3)$$

Using this scale factor:  $\text{new\_width} = \text{int}(40 \times \text{scale\_factor})$   
 $\text{new\_height} = \text{int}(40 \times \text{scale\_factor})$

- The width and height of the original image is set to default value 40.
- To prevent shrinking , the minimum limit is set to below the original size.

Length is calculated distance between two index fingers.

img1\_final\_position / img2\_final\_position , it is used to position the final images when it is locked.



$\text{img1\_resized} / \text{img2\_resized}$ , it is used to resized version of the images based on finger's distance.

For example the tracking formula these elements as follows:

- If distance < 50: The index finger's coordinates will lock the image position.
- If distance  $\geq 50$ : It calculates the scale factor from the distance, resize the image, and reposition it to the index finger's location.

### 3.4 Real-Time Display and Image Placement

The captured frame is then displayed in real-time, with the added features of elements such as buttons and the images. If the image is locked, its position remains fixed until it is deselected. The final position of each image is calculated and the image is cover on the frame at the location using array slicing. It helps to ensure that image manipulating happens without display error.

### 3.5 Error Handling and Clean Exit:

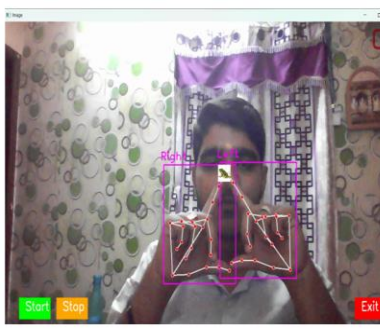
The code includes several checks, such as verifying if the images are loaded correctly. If the error occurs (e.g., missing images or camera issues), it automatically come back and close the program. The try-except-finally block is main loop which wrapped to ensure the camera feed to release an error arises.

## 4 Experimental Results

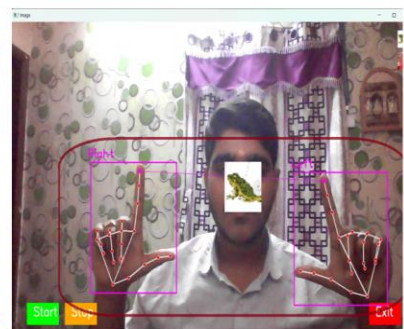
With the use of the Open CV API in Python, this work will begin to recognize hand detection, as seen in Fig. 1.a. The distance is computed using Eq.1 and used to zoom in and out of the chosen figures. This is useful to the students in demonstration sessions where the instructor is introducing his/her subjects to the class. The Fig.1.b shows how one demo image is started to enlarge its size. The Fig.1.c shows how one demo image is started to enlarge its size.



**Figure. 1.a** hand Detection in Open CV



**Figure. 1.b** Zooming out by calculating the distances between fingers



**Figure.1.c** Zooming in as per fingers distance

### 4.1 Hand Detection Accuracy

Observation: The detection of hand accuracy will be vary based on three condition, are followed that is lighting condition, hand positioning, and background noise. With a confidence



threshold of 0.8(detection on=0.8), the system is successfully detecting hands in most cases under the good lighting. In low-light scenarios it may increase the false positives but lowering the threshold might improve detection in low-light. To improve the detection reliability, we can adjust the detection on parameter.

#### **4.2 Distance Calculation Stability**

Observation: The measurement distance between index fingers should be within close distances (e.g., under 50 pixels) however, at longer distances, the movement of hand slight may cause fluctuating measurements. When the distance threshold for locking the position is set at 50 pixels, it effectively allows precise placement of image without drifting. To improve the detection reliability we can adjust the detection on parameter.

#### **4.3 Image Scaling Responsiveness**

Observation: The distance between hands is directly proportional image resizing, it allows dynamic resizing. An users can easily zoom in and zoom out the image by moving the hands. A scaling factor of distance/200 offers moderate control over image size. when tested an image scale it is defaultly fixed to base size 40 x40 pixels. Testing different adding constraints to limit maximum image size could improve user control.

#### **4.4 Latency and Frame Rate**

Observation: Depending on the hardware and camera resolution (1280 x720) in this case it is minor latency. With average webcams and comports, the range of the frame rate from 15 to 30 FPS. Smoother performance can achieve higher-end-setups. Performance detection every second frame it can help to reduce latency and improve real-time responsiveness.

### **5 Conclusions**

Gesture Zoom improves real-time user engagement with digital pictures by showcasing a useful and efficient method of touch less navigation using hand gestures. Although dependable hand tracking and gesture identification are achieved, the system's performance is affected by background noise, hand placement, and illumination. In high-end configurations, our tests provide encouraging outcomes for gesture-based zoom responsiveness and hand detection accuracy with low latency. Future research may concentrate on improving gesture accuracy, shrinking sensor dimensions, and extending the applicability of this technology to further sectors. Gesture Zoom exemplifies a significant breakthrough in the field of human-computer interaction, laying the groundwork for future developments in touch less, intuitive navigation systems.

### **References**

1. Vito Gentile; Daniele Fundarò; Salvatore Sorce, Year: 2019, "Elicitation and evaluation of zoom gestures for touchless interaction with desktop displays", Proceedings of the 8th ACM International Symposium on Pervasive Displays, pp. 1-7.



2. Shun Sekiguchi; Takashi Komuro, Year: 2017, “Vol Grab,” Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems, ACM, pp. 2064–2069.
3. Tadaaki Ishikawa; Y. Horry; T. Hoshino, Year: 2005, “Touch less input device and gesture commands,” 2005 Digest of Technical Papers, International Conference on Consumer Electronics, pp. 205–206.
4. Y. Takeoka; T. Miyaki; J. Rekimoto, Year: 2010, “Z-touch,” ACM International Conference on Interactive Table tops and Surfaces, ACM, pp. 91–94.
5. S. Cronin; G. Doherty, Year: 2018, “Touch less computer interfaces in hospitals: A review,” Health Informatics Journal, Vol: 25, no: 4, SAGE Publications, pp. 1325–1342.
6. C. Valarmathi; S. Sharanya, Year: 2024, “Cyberbullying Detection in Instagram Comments”, Interantional Journal of Scientific Research in Engineering and Management, Vol: 08, no: 008. Indo space Publications, pp. 1–5.
7. C. Valarmathi; S. Saranya, Year: 2024, “Scam Call Detection Using NLP and Naïve Bayes Classifier”, Interantional Journal of Scientific Research In Engineering And Management, Vol: 08, no: 07, pp. 1–6.