CS585 Assignment2: Video processing. Shape analysis.

Reported by Boyang Liu
Collaborated with Douglas Moy

February 2024

1 Problem Definition

In this assignment, we need to utilize algorithms to detect gestures and shapes of hands in videos. This is an object detection and video processing problem. Gesture recognition plays a crucial role in enhancing human-computer interaction, allowing for more intuitive and natural ways to communicate with devices. It eliminates the need for physical contact or traditional input methods, making technology accessible to a wider range of users, including those with disabilities. However, realizing effective gesture recognition systems poses significant challenges, including the accurate interpretation of varied and nuanced human gestures and ensuring robustness against diverse environmental conditions. These complexities necessitate valid approaches to improve recognition accuracy and adaptability.

2 Method and Implementation

To facilitate analysis and improve efficiency, firstly, we tested several algorithms for analyzing single-frame images. Our approach involves several key steps in image preprocessing to enhance the quality and accuracy of the analysis. The process starts with image resizing to maintain a consistent scale across all images. This is followed by converting images to grayscale to reduce complexity and focus on structural details. Gaussian blur is ap-

plied to smoothen the image, reducing noise and making edge detection more effective.

We experimented with setting thresholds directly on RGB images to identify skin and separate hands from the background. However, this method proved inadequate, with its effectiveness heavily dependent on skin tones and the backdrop. Switching to grayscale as binary images emerged as a superior strategy.

To further minimize the impact of noise on the image, we employed the erode and dilate techniques to refine the mask obtained from threshold filtering.

Edge detection is another choice, which is performed using the Canny algorithm, identifying boundaries of objects within the image. Noticing that edges are often unclosed, to accurately detect the position of the hand, we used morphological operations, specifically closing, to close gaps in the edges. This facilitated a more accurate contour detection and can detect hands precisely.

These methods involve finding contours in the processed image, with the assumption that the largest contour corresponds to the object of primary interest, often referred to as the "hand" in this context. A mask is created and filled based on the detected contours to isolate the object from the rest of the image, allowing for further analysis or manipulation.

To accurately delineate the bounding box around the detected hands from the mask, we explored two methods: the straightforward min-max approach and the horizontal and vertical projection technique. Both strategies proved effective in precisely outlining the bounding box.

Subsequently, we ventured into direct video processing, leveraging the correlation and variations between successive frames. We adopted the motion blobs technique, adept at accurately identifying moving segments. By integrating these moving parts, we were able to pinpoint hand locations and discern gestures with high precision.

Having implemented and compared different algorithms, we further designed a hand gesture detecting system. Our gesture recognition system differentiates four distinct hand shapes - fist, L shape, peace sign, and 5 fingers spread - using a combination of skin color detection, contour analysis, and convexity defects within a Tkinter GUI integrated with OpenCV for real-time video processing. The algorithm isolates hand regions through skin filtering, then applies grayscale conversion, blurring, and binary thresholding to facilitate contour detection and hand shape identification. By examining the

number, angles of convexity defects, and hand solidity, it accurately recognizes gestures, employing the cosine theorem for precise angle measurements.

3 Experiments

The most challenging aspect of this assignment was fine-tuning the thresholds. After completing the video read and write coding, I continuously adjusted the parameters and assessed the performance. Typically, I began by testing the first frame or a few frames in the video. Once I achieved satisfactory results, I applied those parameters to process the entire video.

4 Results

Table 1 displays the performance of our system.
Image 1, 2, 3, 4 are the performance of algorithms.
Please find the demo of our system from the link.
I also submitted some videos displaying the performance of our algorithms.

4.1 Results Table

	L shape	**Fist**	**Five Fingers**	**Peace Sign**	**No Gesture**
L shape	100.0	0.0	0.0	0.0	0.0
Fist	0.0	100.0	0.0	0.0	0.0
Five Fingers	0.0	0.0	100.0	0.0	0.0
Peace Sign	0.0	0.0	0.0	100.0	0.0
No Gesture	0.0	0.0	0.0	0.0	100.0

Table 1: The confusion matrix presented above corresponds to the experiment described earlier. The program demonstrated high accuracy in recognizing hand shapes, successfully avoiding any misclassification of static hand gestures. You can view a demonstration of the system in action at the following link.

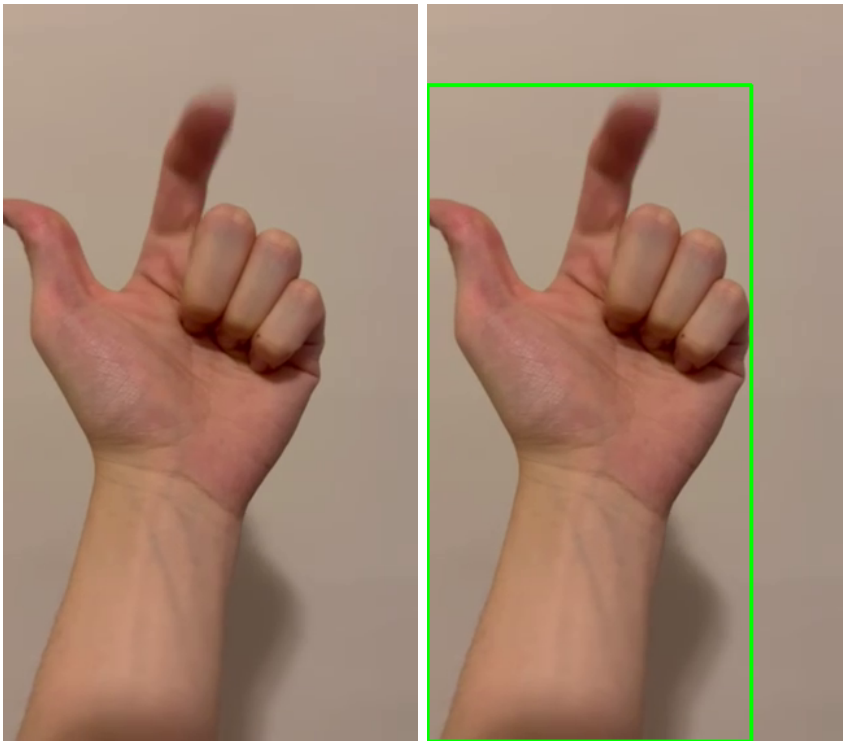

Figure 1: Fingers and its corresponding Canny edge detection result

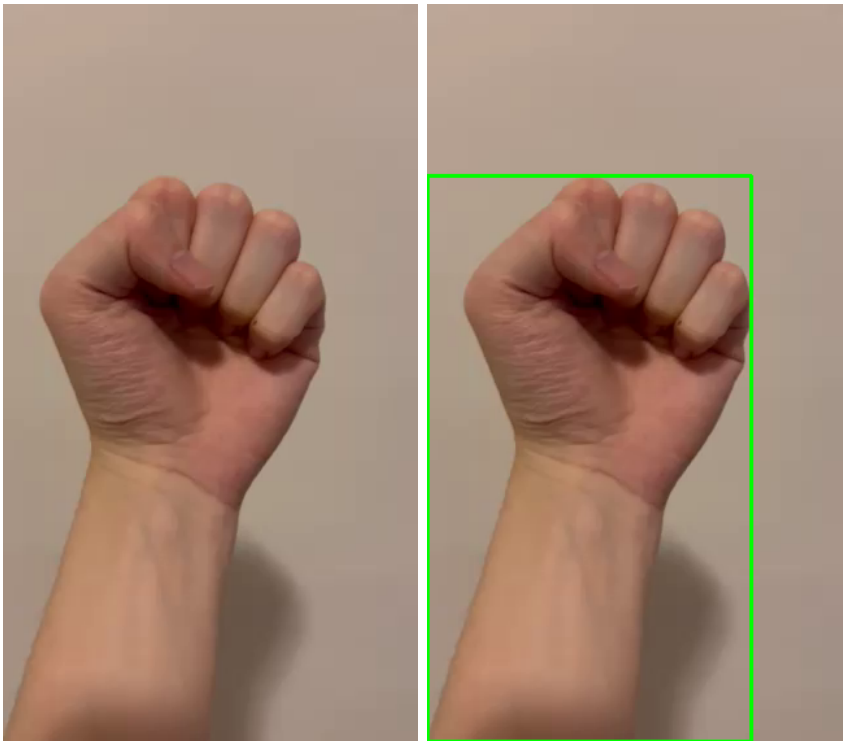

Figure 2: Fist and its corresponding Canny edge detection result

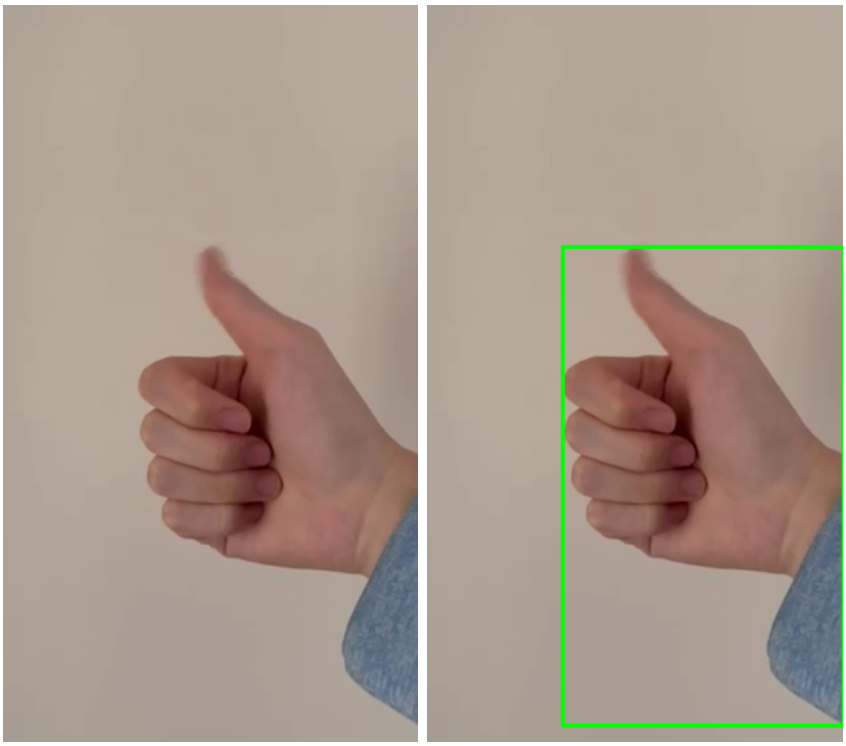

Figure 3: ThumbUp and its corresponding Canny edge detection result

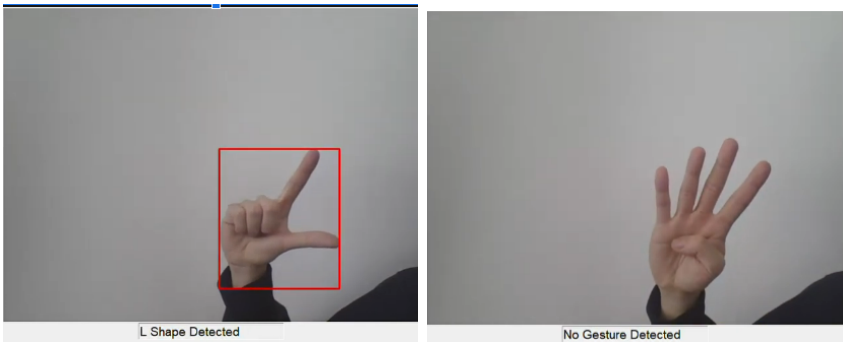

Figure 4: detected gestures and no gestures in the system

5 Discussion and Conclusions

Our system demonstrates excellent performance in accurately detecting correct hand shapes in a controlled environment. However, there may be limitations when dealing with complex backgrounds. While the program meets the requirements of the assignment effectively, there is potential for improvement, especially for commercial applications requiring operation in diverse environments. The approach used for hand detection and classification worked well within the scope of the assignment, but its applicability to more complex hand shapes and backgrounds may require alternative methods, possibly including neural networks.

6 Credits and Bibliography

I received all the assistance I need from GPT. All of them are implementations and parameter explanations of the opencv functions related to the relevant requirements