CS 585 HW 2

Ziye Chen
Feburary 2024

1 Problem Definition

Design and implement algorithms that recognize hand shapes (such as making a fist, thumbs up, thumbs down, pointing with an index finger etc.) or gestures (such as waving with one or both hands, swinging, drawing something in the air etc.) and create a graphical display that responds to the recognition of the hand shapes or gestures.

2 Method and Implementation

In this assignment, I used following methods:

- Skin Color Detection Function

  This function detects whether a pixel belongs to the skin based on RGB values. The destination grayscale image where skin pixels are colored white and the rest are colored black.[1][2]

- Template Matching with OpenCV

  Template Matching is a method for searching and finding the location of a template image in a larger image. OpenCV comes with a function cv.matchTemplate() for this purpose. It simply slides the template image over the input image (as in 2D convolution) and compares the template and patch of input image under the template image. Several comparison methods are implemented in OpenCV. (You can check docs for more details). It returns a grayscale image, where each pixel denotes how much does the neighbourhood of that pixel match with template.[3]

  If input image is of size (WxH) and template image is of size (wxh), output image will have a size of (W-w+1, H-h+1). Once you got the result, you can use cv.minMaxLoc() function to find where is the maximum/minimum value. Take it as the top-left corner of rectangle and take (w,h) as width and height of the rectangle. That rectangle is your region of template.[3]

3 Experiments

In my experiments, I focused on analyzing four distinct gestures, labeled as 'zero', 'one', 'two', and 'three', within a live video setting. Each gesture was subjected to a specific duration of observation: 'zero' was monitored for 5 seconds, while the gestures 'one', 'two', and 'three' were each observed for a duration of 3 seconds. To capture the nuances and variations of these gestures, I implemented a sampling rate of every 0.1 seconds, ensuring a detailed and comprehensive dataset for analysis.

4 Results

4.1 Trial Results

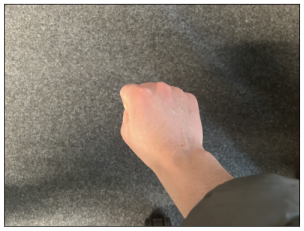
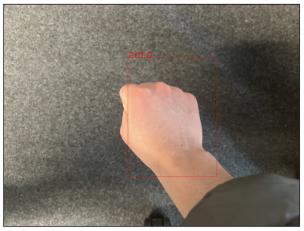
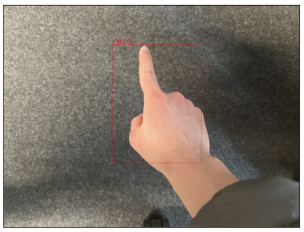
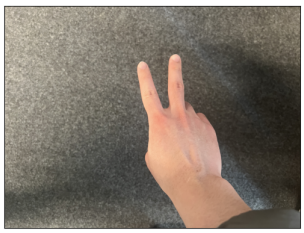
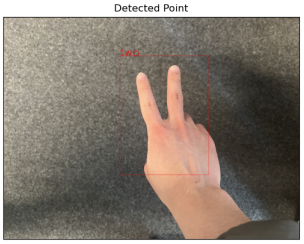
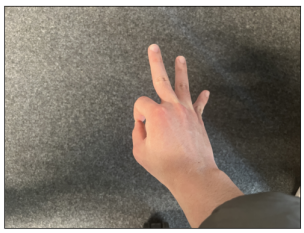
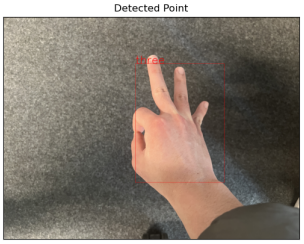
Trial	Source Image	Result Image
Trial 1		
Trial 2		
Trial 3		
Trial 4		

Table 1: Trial Results

4.2 Confusion Matrix

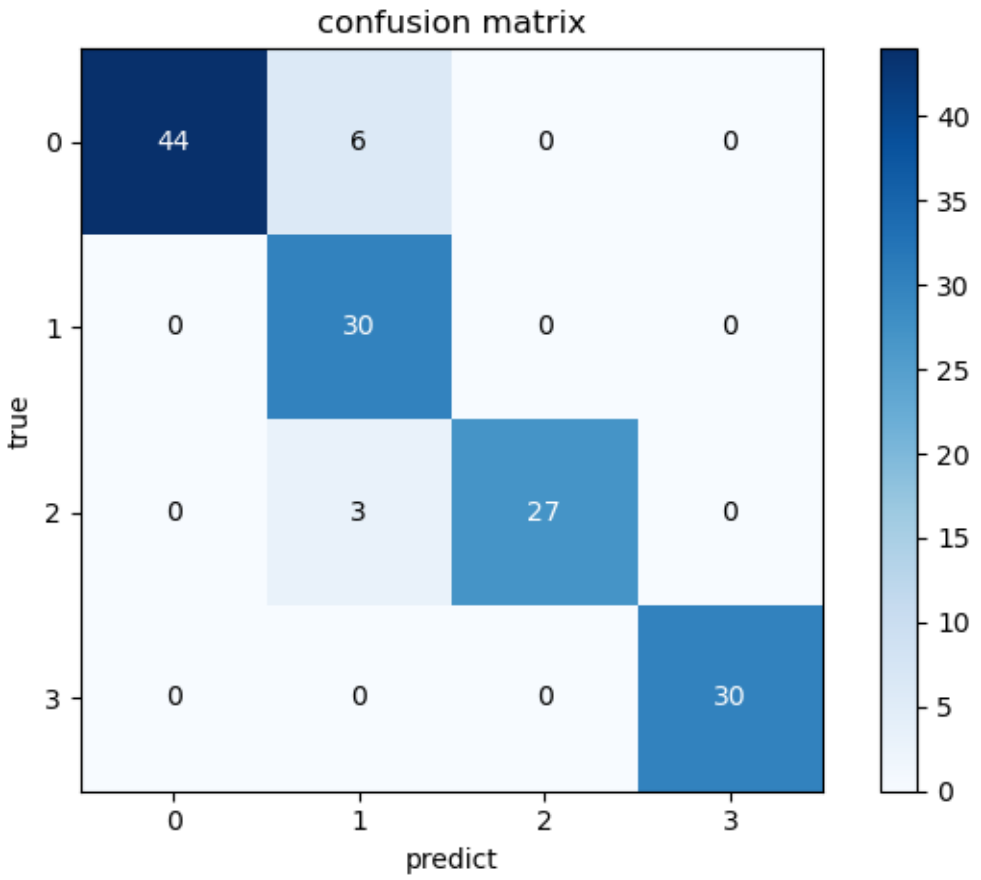


Figure 1: An angry man opened his mouth dramatically

5 Discussion

My method combines skin color detection and template matching with OpenCV. Skin color detection efficiently isolates relevant regions, reducing computational demands, while template matching provides a straightforward mechanism for gesture identification. Despite these advantages, limitations arise from environmental variations affecting skin detection accuracy and template matching's sensitivity to gesture scale and orientation changes.

Experimentally, I observed promising results for recognizing specific gestures in live video, validating the method's effectiveness in controlled conditions. However, challenges were noted with variations in gesture presentation and environmental factors, slightly diverging from initial expectations of near-flawless recognition.

6 Conclusions

My assignment shows the effectiveness of combining skin color detection and template matching for hand gesture recognition in controlled settings. While promising, the method has limitations from environmental variability and gesture diversity. Future work should focus on enhancing adaptability and accuracy through advanced algorithms and addressing recognition challenges in dynamic environments.

References

[1] Praveen Kakumanu, Sokratis Makrogiannis, and Nikolaos Bourbakis. A survey of skin-color modeling and detection methods. *Pattern recognition*, 40(3):1106–1122, 2007.

[2] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreeva. A survey on pixel-based skin color detection techniques. In *Proc. Graphicon*, volume 3, pages 85–92. Moscow, Russia, 2003.

[3] OpenCV.