

Contactless Fingerprint detection

Shrey Kothavade

Master of Engineering in Information Technology SRH Hochschule Heidelberg

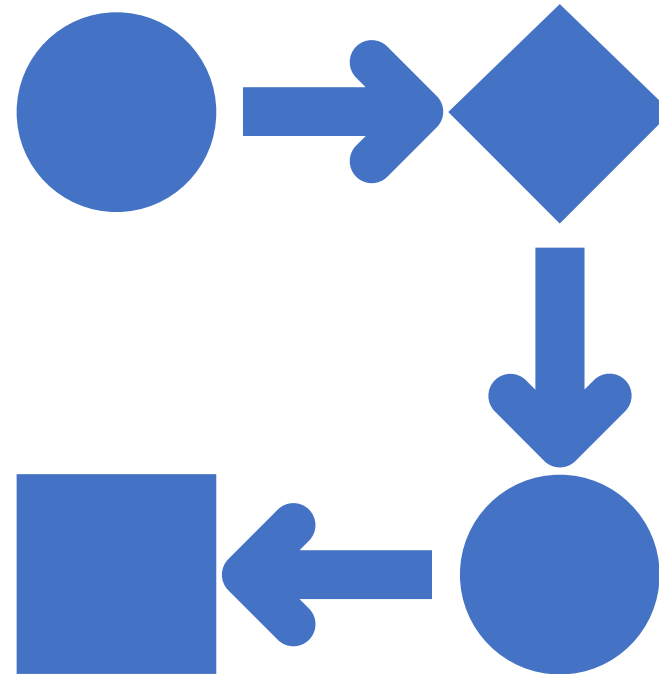
Matriculation Number: 11018044

Heidelberg, Germany

11018044@stud.hochschule-heidelberg.de

Flow of the presentation

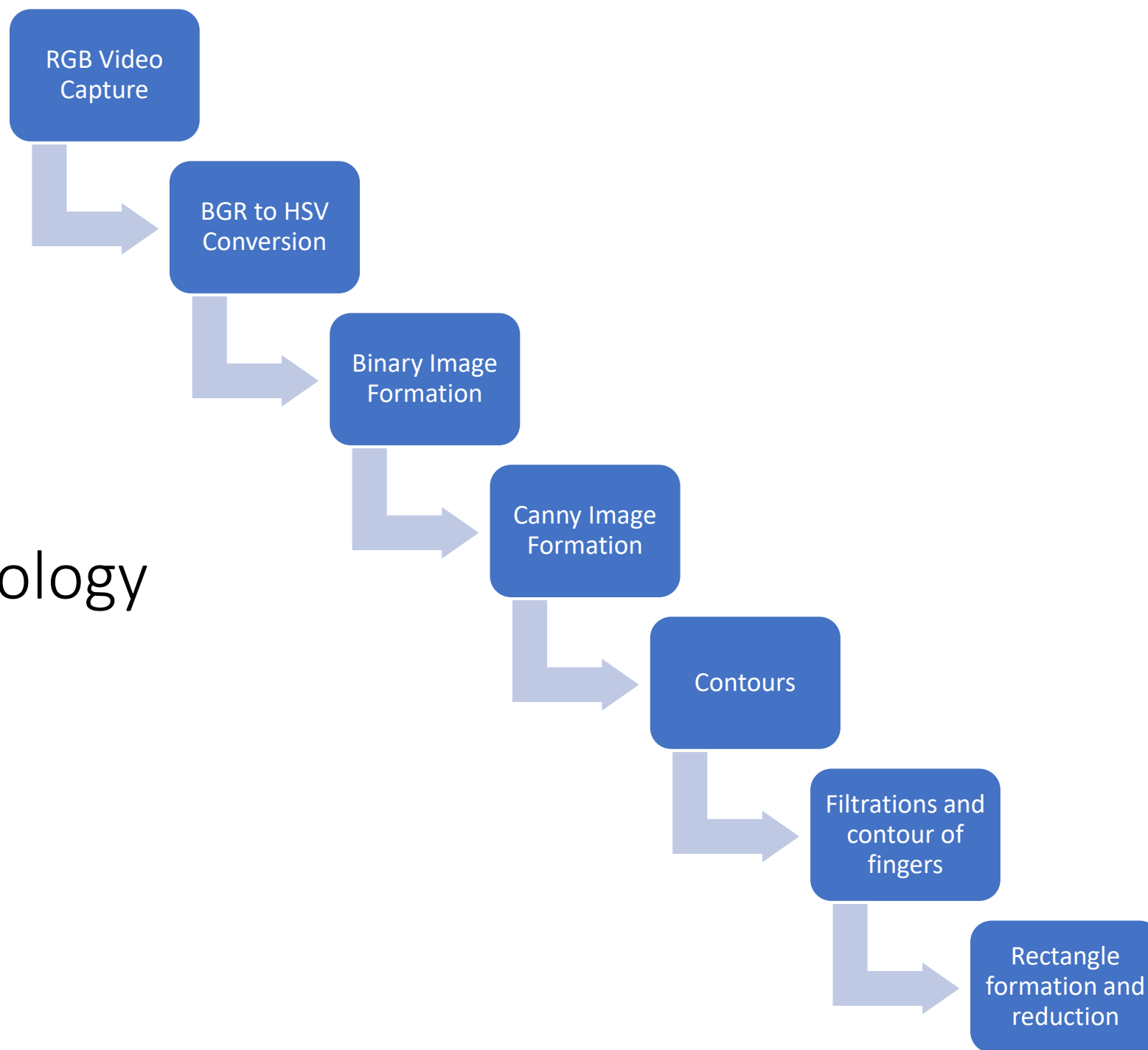
- Motivation
- Methodology
- Step by step algorithm
- Result



Motivation

- Currently, devices only record fingerprints through touch, but contactless fingerprint capture will increase security.
- Biometric and forensic verification and identification employ contactless palm/finger detection.
- It is simple to develop and has a robust nature.
- A basic webcam is enough to detect a palm/finger, which are very economic and easy to utilize.
- It is simpler for the developer to create his or her algorithm using OpenCV.

Methodology



RGB Video Capture

- We capture video in RGB color space.
- Any additive color space built on the RGB color model is referred to as an RGB color space. The red, green, and blue additive primaries' chromaticity coordinates establish an RGB color space, and a standard illuminant is often used as the white point.
- It is an analog component video standard for transferring video data.
- Command used to initiate a Video:-
 - `VideoCapture cap(0); if(!cap.isOpened())`

RGB to HSV converter

- We convert to HSV because the color of the image in BGR color space is difficult to distinguish in terms of luminance.
- we use the HSV color space to see the image's Intensity, Chroma, and Dominant Wavelength.
- Command/function used to convert RGB to HSV :-
 - `cvtColor(Org_Frame, Img_hsv, CV_BGR2HSV);`

Binary Image Formation

- Any digital image that just has the two hues of black and white is called a binary image.
- A binary picture comprises pixels with the two potential values of zero and one from the standpoint of image processing.
- A pixel's value of 0 symbolises the colour black, which is completely opaque. Pure white colour is represented by a pixel value of 1.

Canny Image Formation

- It is a multi-stage algorithm and its stages are as follows -
 - Noise Reduction
 - Finding Intensity Gradient of the Image
 - Non-maximum Suppression
 - Double Threshold
 - Hysteresis Edge Thresholding
- OpenCV puts all the above in single function, `cv.Canny()`.

Contours

- Contours are essentially lines that connect high-intensity points to generate a single large-area image.
- The contours will initially be produced for the entire palm region.
- So, we must filter the palm region, which is the largest region, and ensure that only contours for the fingertips are generated.

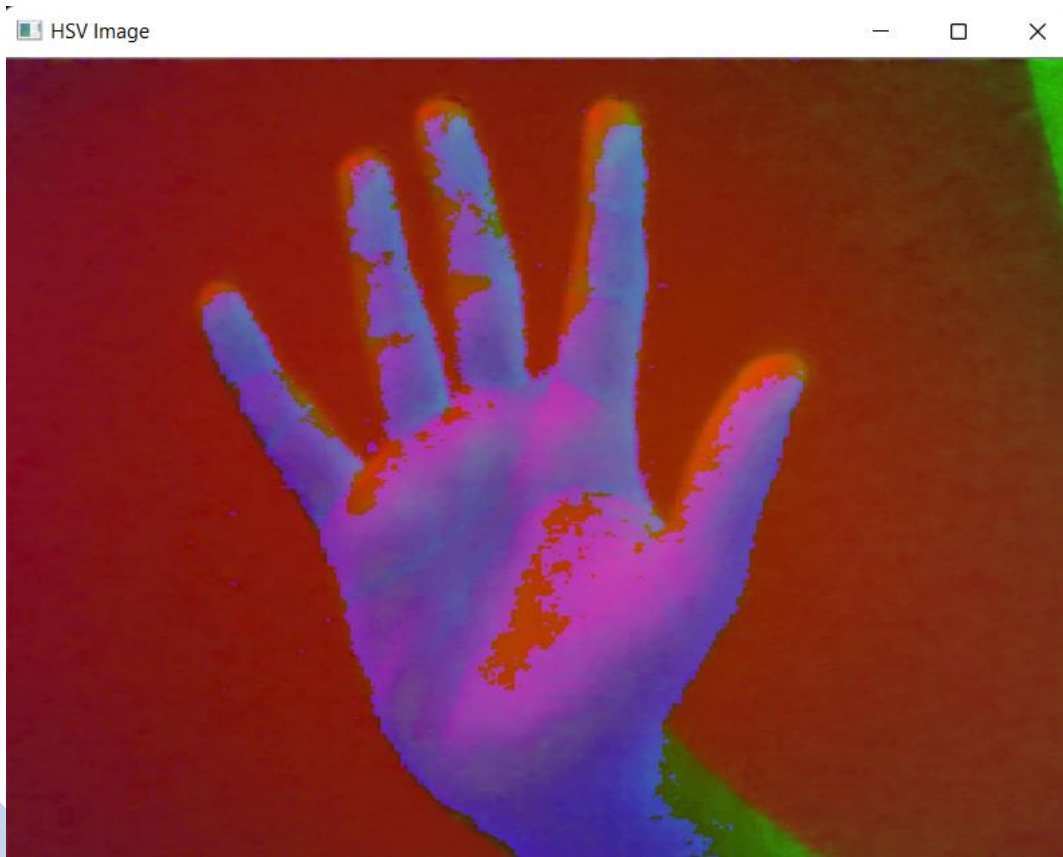
Rectangle formation and reduction

- After the fingertips have moulded their shapes, the next step in this procedure is to draw rectangles across the region of the fingertip.
- We can figure out the rectangle's dimensions and reduce it into smaller rectangles by drawing rectangles along the contour over the entire finger area at first.
- This can be done with the help of the built-in function **(minAreaRect ())**, which returns the angle, height, width, and center point of the rectangle.

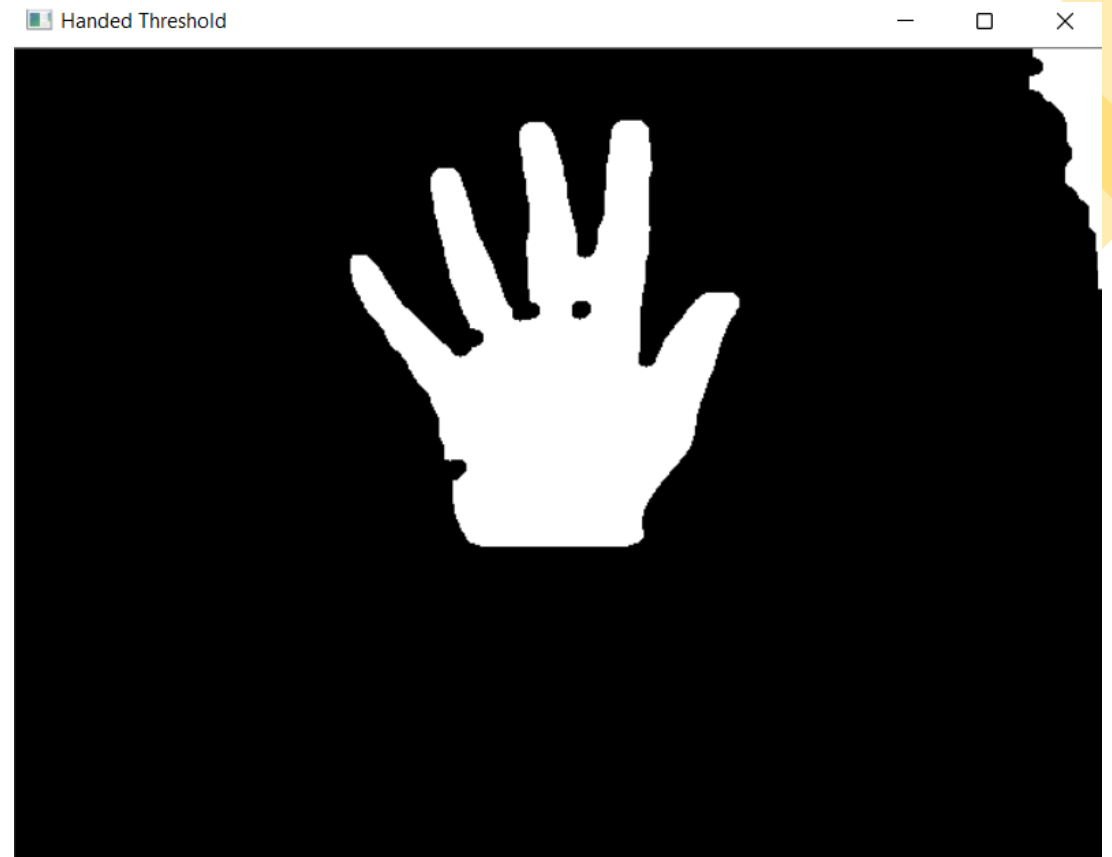
Process and Functions used

- `getskin`: to set the skin color threshold's
- `RGB2HSV`: to convert image from RGB to HSV color space
- `THRESH_BINARY`: to set threshold value
- `imshow`: to show image
- `Canny image`: to detect edges
- `Contours`: to make square around specific parts
- `minRect`: to minimise the rectangle to fingertip

Results

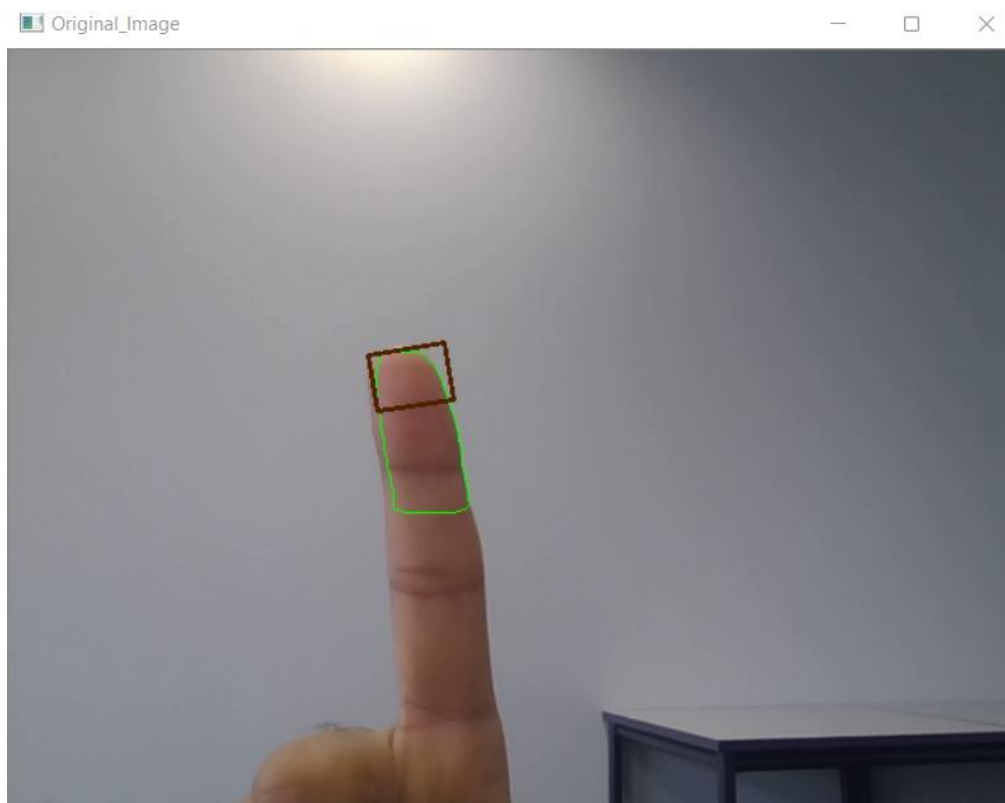


BGR image

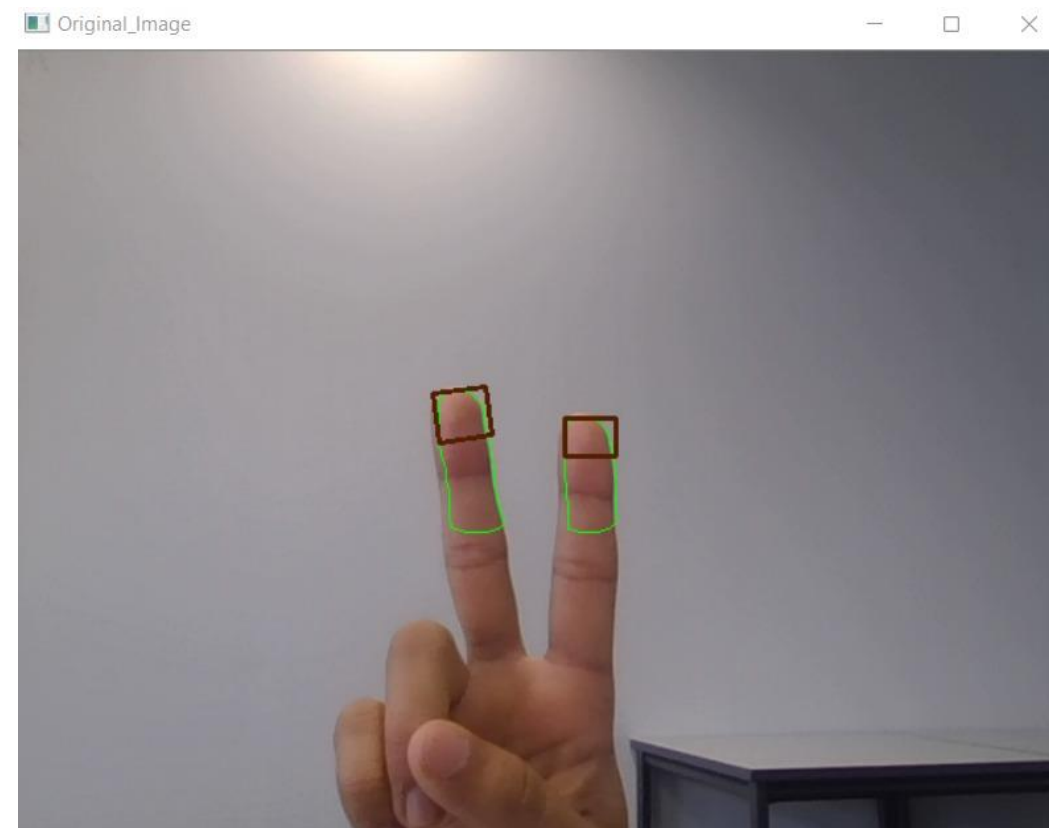


Binary Image

Results

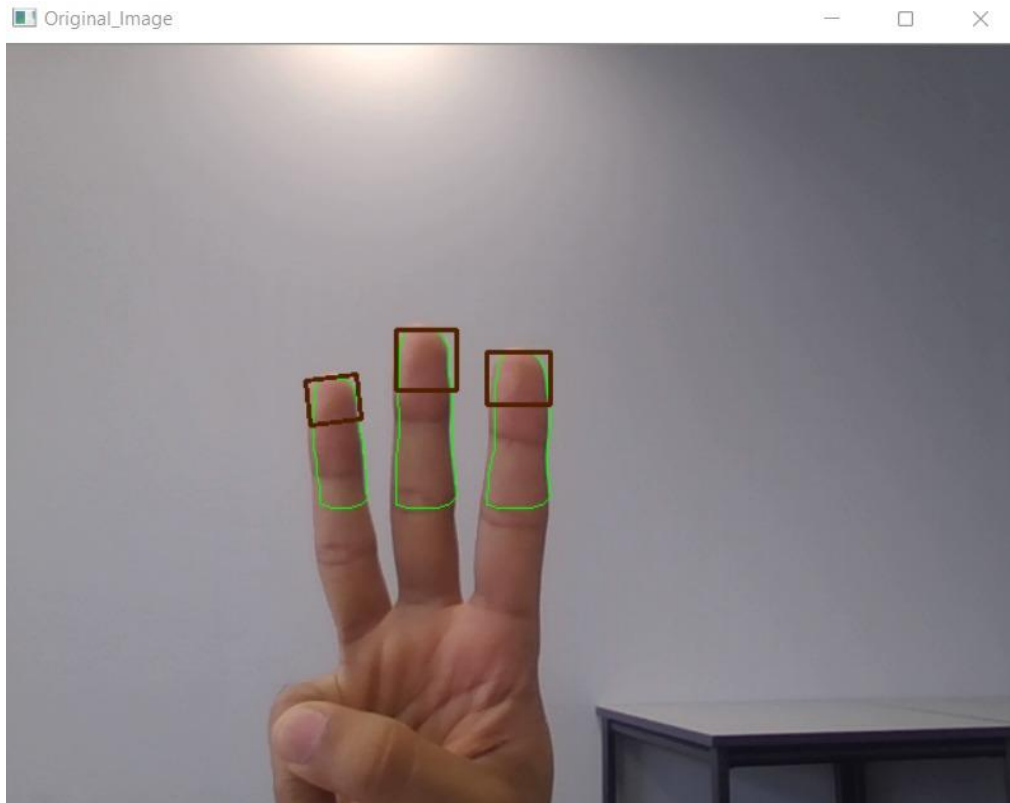


Fingerprint detection for 1 finger

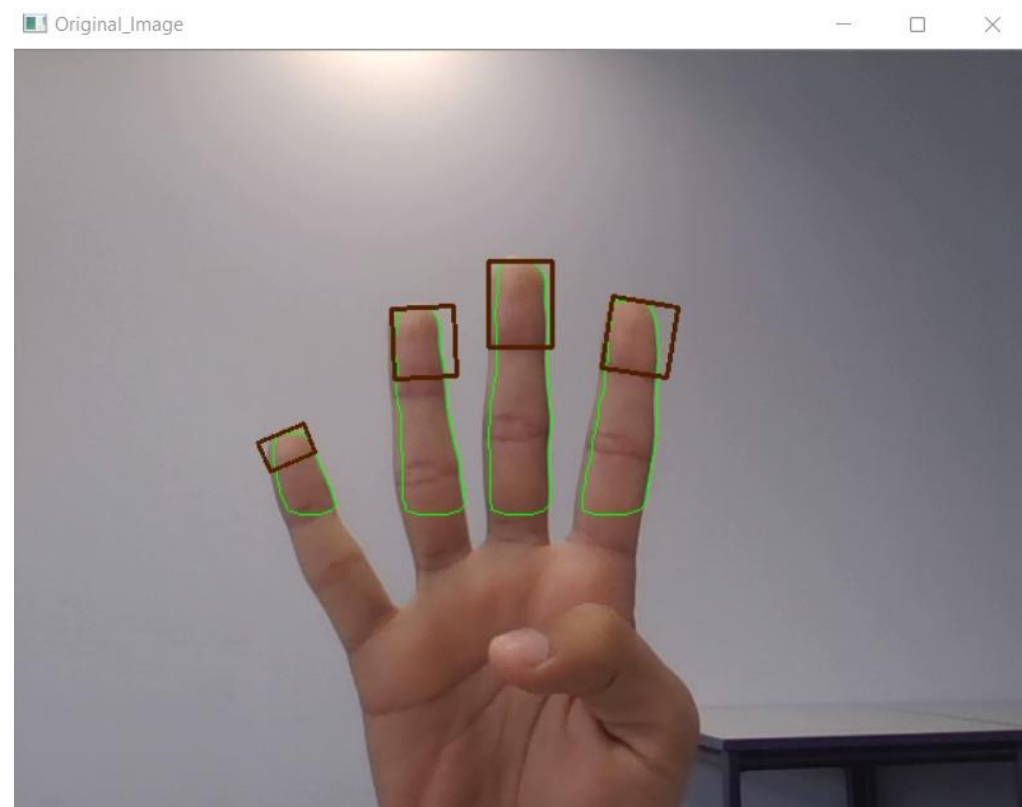


Fingerprint detection for 2 fingers

Results



Fingerprint detection for 3 fingers



Fingerprint detection for 4 fingers

Thank You

Shrey Kothavade

MEng – IT(Summer 2022)

Matriculation Number: 11018044

