Optical Touch Detection by Tracking Fingertip Color

Alex Wang
University of California, Los Angeles
alexwang23@ucla.edu

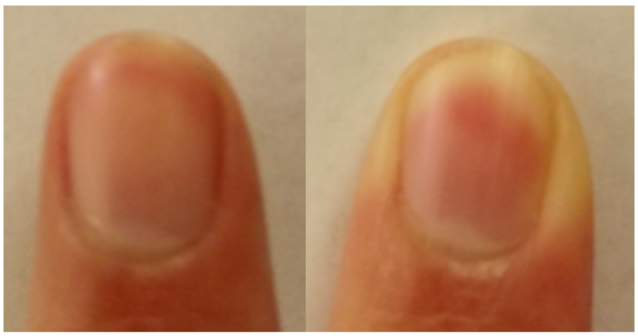

Figure 1. A fingertip applying pressure (right) is lighter in color compared to when it is at rest (left).

Abstract

A single camera can be used to detect touch input on an ordinary surface. Current optical imaging methods require well-defined shadows, infrared illumination, or a reflective surface. This project introduces a method to register touch by tracking the change in color of the user's fingertips when the user applies pressure on a surface. The user's fingertip is detected by a neural net, and the location data obtained via tracking is transformed into the input data using a homography. The proposed technique can convert a desk or a wall into a tactile sensor at low cost and with high portability.

1. Introduction

Touch-enabled input devices are ubiquitous due to the flexible and intuitive controls provided to the user. Touchpads and touchscreens have paved the way for highly portable electronic devices, but they require specialized hardware sensors. Current consumer-grade touch sensors are cost restrictive when scaled to larger areas, and they are typically limited to specific surfaces. With optical imaging methods, ordinary surfaces can be converted into touch-sensitive interfaces.

This project explores a novel technique for fingertip touch detection based on the human finger's response to normal force when making contact with a surface. Pressure on the skin temporarily restricts blood flow to the contact area, causing the skin to appear pale or blanched (Fig. 1). A heuristic based on the user's biological response can provide a more robust user interface with fewer false touch detections compared to other optical imaging methods.

2. Related Work

Previously developed methods have used multiple mounted cameras [1] or a stereo camera [2]; however, a multiple-camera setup is not a portable solution, and stereoscopic cameras are not yet common in mobile devices. Among the single camera methods that have been proposed, their additional requirements still prove to be limiting factors.

Infrared optical techniques can detect touch with a single camera, as achieved in [3] by illuminating the user's fingertip with a laser line projector aimed parallel to the surface. While effective, these techniques require an infrared light source.

Reflective surfaces are useful as shown in [4], where a fingertip meeting its reflected image signifies a touch. While this technique is inherently limited to reflective surfaces, it is useful for converting normal displays into touchscreens. The issue with illuminated displays is that the finger's reflection is not easily visible when the display is on. The camera must be mounted at an angle nearly parallel to the surface in order to capture enough intensity from the reflection. This results in poor spatial resolution when detecting touch location along the camera's depth direction.

In [5] a lighting setup causes the user's finger to cast a dark shadow against a light background. Contact is assumed when the camera detects that the fingertip is sufficiently close to the tip of the shadow. Proper lighting and a textureless surface is required for the finger to cast a well-defined shadow with enough contrast. In an indoor office setting, there may be multiple diffuse light sources which would interfere with the edge detection algorithms employed by this method. Even when the appropriate conditions are met, certain viewing angles relative to finger ori-

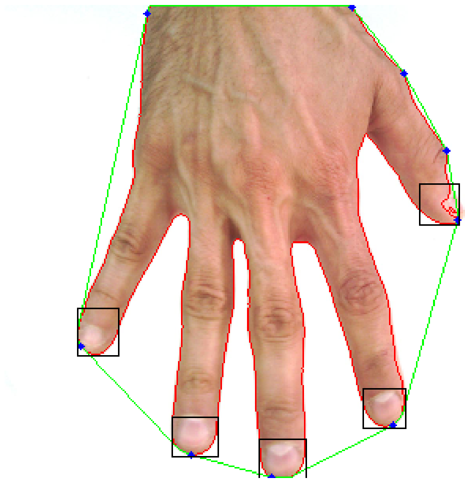

Figure 2. Convex hulls are used to label fingertips.

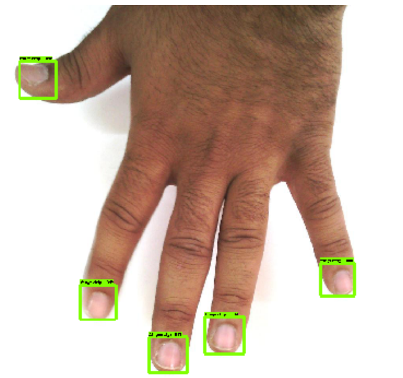

Figure 3. Inference results from the neural network.

entation result in false touch detections.

3. Theoretical Model

The two main aspects of touch input are the touch location and the touch signal itself. The touch location is represented by two-dimensional points on a plane, and the touch signal simply has a binary state of on or off.

The primary contribution of this project is the novel method of detecting the touch signal. According to [6], a blanch response is significantly detectable in the visible spectrum for both light-skinned and dark-skinned individuals. The color of the blanch response will vary depending on skin pigmentation, so a manual calibration method is used. When the user generates the blanch resposne in his or her fingertip, the range of colors that appear can be recorded to identify when touch occurs. A basic heuristic is to filter the RGB values of the fingertip image for the color of the blanch response. Counting the number of pixels within the calibrated bounds will give an amplitude, which can be thresholded for the digital touch signal.

The touch location can be determined by applying a homography on the spatial data obtained from a tracking algorithm. In order to initialize the tracker, I elected to use a neural network. The 11K Hands dataset [7] provides an excellent source of hand images, but there aren't any labels for fingertips. To process the images for training, I wrote a script that automatically generates bounding boxes for each fingertip. First the script finds the contour of the hand by thresholding away the white background. Then the convex hull of the contour is found, and the vertices of the convex hull yield approximate locations for the fingertips (Fig. 2).

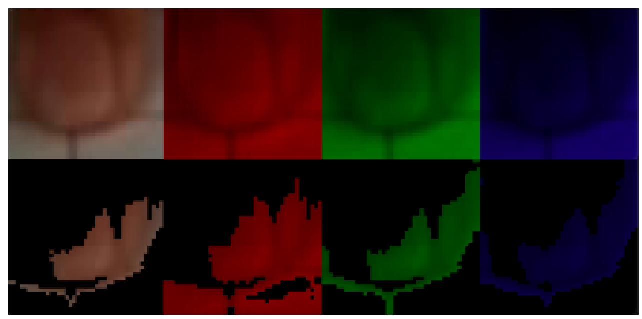

Figure 4. The top row shows an image of a fingertip applying pressure and its resepective RGB channels. The bottom row shows the results of masking away pixels that are outside of the calibrated color range.

4. Experiments

A Raspberry Pi 3 is used for the camera setup. A compact neural network is needed due to the Raspberry Pi's limitations in computing power and memory, so I chose to use MobileNet with SSD architecture on Tensorflow [8] [9]. With 436 labelled images, it took around 20 hours to train the network on a 1.3 GHz Intel Core i5 CPU (2013 MacBook Air). Running inference on the Raspberry Pi takes 1-2 seconds, but the detection confidence is typically 99% (Fig. 3).

Most of the computational operations are called from the OpenCV library, including the CSR-DCF tracking algorithm [10], RGB value thresholding, and homography estimation using RANSAC. With all of these running sequentially on a single thread, the framerate is reduced to less than 4 frames per second.

The object detection and tracking provide a region of interest that is masked based on the calibrated ranges. After applying pressure and producing a blanch response, bounds

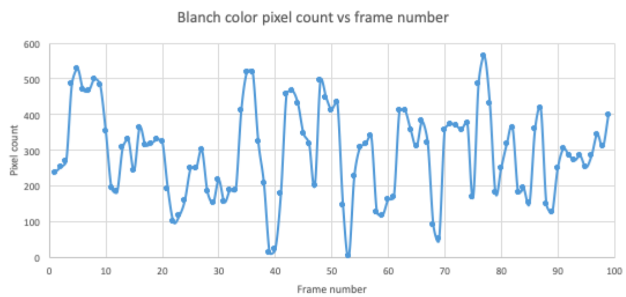

Figure 5. By counting the number of pixels in the fingertip ROI that are also within the calibrated RGB bounds, an analog touch signal is obtained.

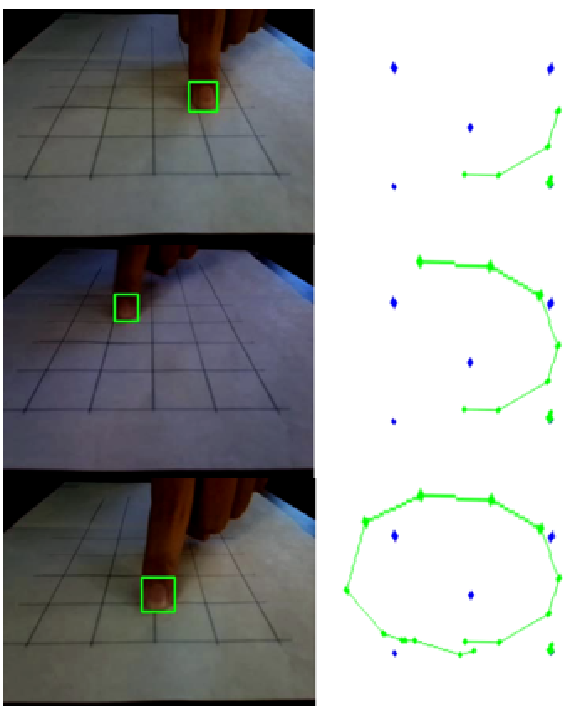

Figure 6. Three frames of the user drawing a circle. The blue points are known calibration points used for estimating the homography. The green points are the user's touch input.

are applied to the pixels of the ROI. For the trial run shown in Figure 4, the range for the red channel is from 80 to 129, green channel range is from 51 to 81, and the blue channel range is from 35 to 67. A mask is applied such that only pixels with RGB intensity values inside the calibrated bounds are counted.

After the RGB thresholding operation is applied, the number of pixels remaining in the ROI gives touch information (Fig. 5). An upward spike in this pixel count signifies the user touching the surface. Once the touch signal has been successfullly calibrated, the user is prompted to touch several known calibration points for homography estimation. Finally the coordinates from the fingertip tracker are transformed to give coordinates on the surface plane, thus retrieving the complete touch input (Fig. 6).

5. Discussion and Conclusion

This project demonstrates a proof of concept for detecting touch by optically measuring the blanch response in the user's fingertip. While this is a viable method for converting an ordinary surface into a touch interface, there are still major limitations and room for improvement.

The described experimental setup can only run at a low framerate, which can greatly hinder usability. The tracking stage suffers the most from the low framerate, and the bounding box is often separated from the fingertip. Framerate can be improved on the Raspberry Pi by using multiple threads in parallel. There are many possible optimizations for the touch detection pipeline, but a better setup would be to use a USB webcam attached to a more powerful processor. This would make sense when using the touch input to control a cursor on a computer.

Thresholding RGB values is the simplest method for detecting the blanch response. Other colorspaces can be used such as HSV or LAB. A heuristic can also be developed that takes into account both the blanch color and the resting color. Dynamic factors can be incorporated, such as using different bounds in different areas to compensate for non-uniform lighting. Other possibilities include implementing k-means clustering or even another neural network. Feeding the ROI into a smaller neural network can be used to detect blanch response and shape deformation.

Code and videos can be found at https://github.com/alexbox23/Single-Camera-Touchpad

References

[1] I. Katz, K. Gabayan, and H. Aghajan, "A multi-touch surface using multiple cameras," 2007. Proceedings of the 9th International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS'07).

[2] A. Agarwal, S. Izadi, M. Chandraker, and A. Blake, "High precision multi-touch sensing on surfaces using overhead cameras," 2007. Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP'07).

[3] J. Liang, K. Lin, H. Chen, and W. Wang, "Turn any display into a touch screen using infrared optical technique," *IEEE Access*, vol. 6, pp. 13033–13040, 2018.

[4] S. Chen, L. Zhang, and S. Serikawa, "Development of single camera finger touch position detection system," *Journal of the Institute of Industrial Applications Engineers*, vol. 2, no. 3, pp. 108–112, 2014.

[5] E. Posner, N. Starzicki, and E. Katz, "A single camera based floating virtual keyboard with improved touch detection,"

2012. 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel.

[6] M. A, S. MG, T. V, T. G, S. BJ, and M. HH., “Eliminating the issue of skin color in assessment of the blanch response.,” *Adv Skin Wound Care*, vol. 14, no. 4, pp. 180–188, 2001.

[7] M. Afifi, “11k hands: Gender recognition and biometric identification using a large dataset of hand images,” 2017.

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” 2017.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” 2016. ECCV 2016.

[10] A. Lukezic, T. Vojir, L. Cehovin, J. Matas, and M. Kristan, “Discriminative correlation filter with channel and spatial reliability,” *International Journal of Computer Vision*, 2016.