



“Project Palm”

Nikhil Dixit | Thejas Prasad | Chirag Sakhuja | Julian Sia

Overview

- Motivation
- Description
- Objectives
- Methods
- Results
 - Progress
 - Performance
- Demo
- Challenges

Motivation

- Extend the range of use (literally) of your machine
- Explore more about tracking with motion

Description

- Capture video from webcam
- Track fingers across frames
- Translate gestures into mouse operations

Objectives

- P1: Track and extract the hand using a built in webcam
- P2: Track individual fingers from the extracted hand
- P3: Analyze the traces of the fingers to identify gestures
- P4: Map the gestures to mouse

Methods

Methods: Languages, Libraries, Hardware

- Python 2.7
- NumPy
- OpenCV
- PyUserInput
- Laptop Specs
 - Intel i5 @ 2.6 GHz
 - 720p webcam

Methods: Algorithm

```
hue = detect_hue()  
while (camera is recording):  
    frame = current_frame(camera)  
    cleaned_frame = preprocess(frame, hue)  
    hand = detect_hand(cleaned_frame)  
    fingers = detect_fingers(hand)  
    map_to_mouse(fingers)
```


Methods: Algorithm – detect_hue()

def detect_hue() :

1. Capture hue from predetermined region for 25 frames
2. Take 5th and 95th percentiles of hues in region
3. Adjust base hue parameters

Methods: Algorithm – preprocess ()

```
def preprocess (frame, hue) :
```

1. Convert frame to HSV
2. “Black out” non-skin parts of frame
3. Apply erosions and dilations to
remove noise
4. Apply Gaussian blur to smooth edges
5. Convert from HSV to gray

Methods: Algorithm – `detect_hand()` , `detect_fingers()`

`def detect_hand(cleaned_frame) :`

1. Find all contours in frame
2. Take contour with largest area (by our definition, this is the “hand”)

`def detect_fingers(detected_hand) :`

1. Draw convex hull using contours
2. Find convexity defects (which correspond to # of fingers)
3. Dismiss spurious fingers

Methods: Algorithm – map_to_mouse()

```
def map_to_mouse(fingers) :  
    if (2 fingers up)  
        lock mouse  
        if (2 fingers down)  
            do left click  
    else if (3 fingers up)  
        lock mouse  
        if (3 fingers down)  
            do right click  
    else  
        track mouse corresponding to palm movement
```

Results

Results

- **P1**: Tracking and extracting the hand using a built in webcam
- **P2**: Track individual fingers from the extracted hand
- **P3**: Analyze the traces of the fingers to identify gestures
- **P4**: Map the gestures to mouse

denotes an **Original/Complete Objective**

denotes an **Original/Partially Complete Objective**

denotes a **Modified/Complete Objective**

Achievement Statistics

- P1: 69.6% accuracy
- P2: 46.7% accuracy**
- P3: *undefined*
- P4: *undefined*

denotes an Original/Complete Objective

denotes an Original/Partially Complete Objective

denotes a Modified/Complete Objective

denotes a non-ideal case

P1: Hand Tracking & Extraction

- Progress: **Completed**
- Performance
 1. Good for clear, consistent, well-lit, non-skin colored backgrounds
 2. Bad with noisy backgrounds

P2: Digit Tracking & Extraction

- Progress: **Completed**
- Performance
 1. Good when hand is neither obfuscated by another object in the front nor partially obfuscating the face
 2. Difficult to distinguish 1 from 0 fingers
 3. Spurious single finger detection in dark area

P3: Gesture Tracking

- Progress: **Completed**
- Performance
 1. Good when hand is neither obfuscated by another object in the front nor partially obfuscating the face
 2. Difficult to distinguish 1 from 0 fingers

P4: Map Arbitrary Gestures

- Progress: Partially Complete
- Performance
 1. Good performance when conditions listed in P1, P2, and P3 are met
 2. Somewhat sluggish mouse movement
 3. Mouse, Left click, and right click (somewhat)
 4. No scrolling, drag and drop, pinch and zoom, arbitrary gesture tracking

Demo

Challenges ☹️

- Spurious Skin Detection (False +)
 - **Partial Fix:** Erosions and Dilations, Blurring
 - **More Robust Potential Fix:** Stereo Vision
- Spurious finger detection (False +)
 - **Fix:** Dismissal Heuristic
 - **More Robust Potential Fix:** Hull points
- Suboptimal Detection (False -)
 - Skin heuristic based on fixed range
 - **Potential Fix:** Dynamic Range

Questions?

Further Reading

- ***Adaptive Skin Color Model for Hand Segmentation*** by Dawod et. al.
- ***Implementation of an Improved HCI Application for Hand Gesture Recognition*** by Joshi et. Al.

Citations

- <http://www.pyimagesearch.com/2014/08/18/skin-detection-step-step-example-using-python-opencv/>
- <http://simena86.github.io/blog/2013/08/12/hand-tracking-and-recognition-with-opencv/>