

ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN MÔN HỌC
NHẬP MÔN THỊ GIÁC MÁY TÍNH
ĐỀ TÀI: ROCK SCISSORS PAPER

GIẢNG VIÊN HƯỚNG DẪN

Thầy Nguyễn Vinh Tiệp

SINH VIÊN THỰC HIỆN

Lê Tuấn Anh - 17520237

Nguyễn Hưng - 17520541

TP. HỒ CHÍ MINH, 2020

MỤC LỤC

| | |
|---|----|
| DANH MỤC HÌNH | 3 |
| CHƯƠNG I. TỔNG QUAN | 4 |
| 1. Giới thiệu bài toán | 4 |
| 2. Mục tiêu | 4 |
| CHƯƠNG II. PHƯƠNG PHÁP | 5 |
| 1. Phương pháp trước đây | 5 |
| 1.1.1. Thuần Segmentation | 5 |
| 1.1.2. Máy học | 5 |
| 2. Phương pháp của đề tài | 5 |
| 2.1. Segment bàn tay | 5 |
| 2.1.1. Tìm màu da của người dùng | 5 |
| 2.1.1.1.1. Tách pixel màu để có biểu đồ HSV | 6 |
| 2.1.1.1.2. Tìm ngưỡng Max/Min | 6 |
| 2.1.2. Thresholding | 7 |
| 2.1.3. Adjusting | 7 |
| 2.2. Phát hiện bàn tay và các ngón tay | 7 |
| 2.2.1. Xác định đường Contour | 7 |
| 2.2.2. Nhận dạng các ngón tay, kẽ tay | 8 |
| CHƯƠNG IV. CẢI TIẾN VÀ HOÀN THIỆN ĐỀ TÀI | 13 |
| CHƯƠNG V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN | 13 |
| TÀI LIỆU THAM KHẢO | 14 |

DANH MỤC HÌNH

| | |
|---|----|
| Figure 1. Ảnh màu gốc và Histogram của Hue | 6 |
| Figure 2. Sơ đồ thuật toán tìm kiếm đa ngưỡng | 7 |
| Figure 3. Khoảng giá trị HSV thu được sau khi thực hiện giải thuật trên | 7 |
| Figure 4. Minh họa việc biểu diễn Convex, Defect và Contour | 9 |
| Figure 5. Các trường hợp sai sót trong việc phân biệt | 9 |
| Figure 6. Theo thứ tự Contour của Búa, Kéo và Giấy | 11 |
| Figure 7. Theo thứ tự Convex Hull của Búa, Kéo và Giấy | 12 |
| Figure 8. Output sau khi đánh dấu các biểu tượng | 13 |

CHƯƠNG I. TỔNG QUAN

1. Giới thiệu bài toán

Rock, Scissors, Paper – Bao, Kéo, Giấy là một cho chơi phổ biến cho mọi lứa tuổi. Luật chơi của Búa, Kéo, Giấy cũng hết sức đơn giản, mỗi người chơi dùng một tay để tạo ra các hình khối khác nhau và người chơi sẽ được coi là thắng nếu họ chọn: Búa bẻ gãy Kéo, Kéo cắt Giấy hoặc Giấy bọc búa. Tất cả người chơi phải ra các hình khối cùng lúc để đảm bảo sự công bằng.

Bằng việc sử dụng kỹ thuật Segmentation và tìm Contour, Convex Hull cho từng khung ảnh, ta sẽ có một mô hình đáng tin cậy có hiệu suất tốt.

2. Mục tiêu

Có rất nhiều cách giúp ta xác định tư thế của bàn tay, tuy nhiên, mỗi phương pháp đều đòi hỏi tài nguyên, thời gian và có độ chính xác khác nhau. Sử dụng hướng tiếp cận không quá phức tạp – Segmentation hình ảnh dựa trên việc thresholding màu da và sử dụng Contour, Convex Hull để tìm Cực đại và Cực tiểu. Lúc này, việc xác định tư thế tay là việc đếm số lượng các Cực đại, cực tiểu và khoảng từ điểm Cực tiểu tới đường thẳng nối 2 điểm Cực đại.

CHƯƠNG II. PHƯƠNG PHÁP

1. Phương pháp trước đây

1.1.1. Thuần Segmentation

Ý tưởng của phương pháp này tương đối dễ hiểu:

- Xác định giá trị màu da để threshold tạo ảnh Binary
- Nhận định vùng liên tục có diện tích lớn nhất là đối tượng cần nhận dạng
- Nhận dạng từng tư thế dựa trên tỷ lệ chu vi và diện tích

1.1.2. Máy học

Các đề tài sử dụng kỹ thuật Supervised Learning hoặc Unsupervised Learning training trên những bộ dữ liệu lớn mang đem lại độ chính xác cao.

2. Phương pháp của đề tài

2.1. Segment bàn tay

Lấy màu da người làm để threshold bàn tay ra khỏi background. Làm cách nào để tìm được giá trị thresholding?

2.1.1. Tìm màu da của người dùng

Nhóm đã chọn ra khoảng giá trị HSV thông qua giải thuật tham khảo bên dưới:

2.1.1.1.1. Tách pixel màu để có biểu đồ HSV

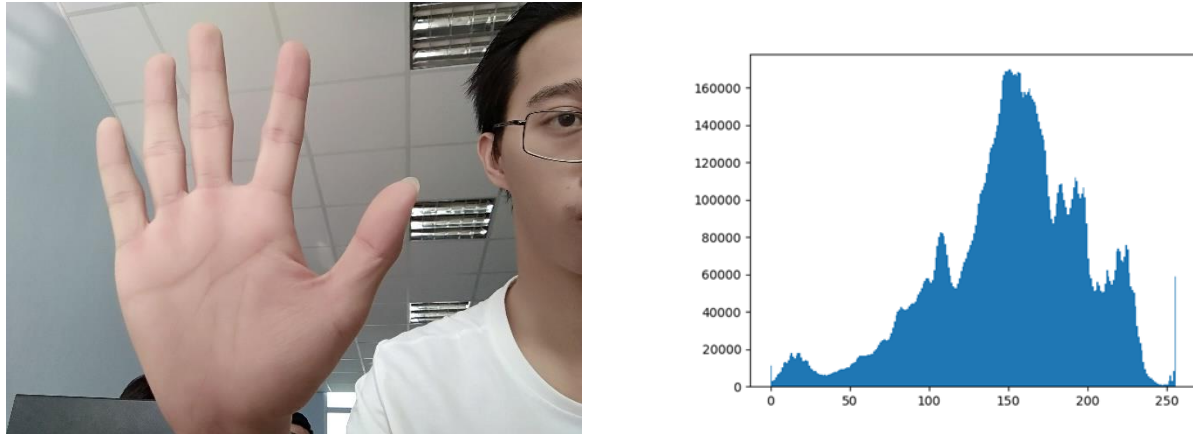


Figure 1. Ảnh màu gốc và Histogram của Hue

2.1.1.1.2. Tìm ngưỡng Max/Min

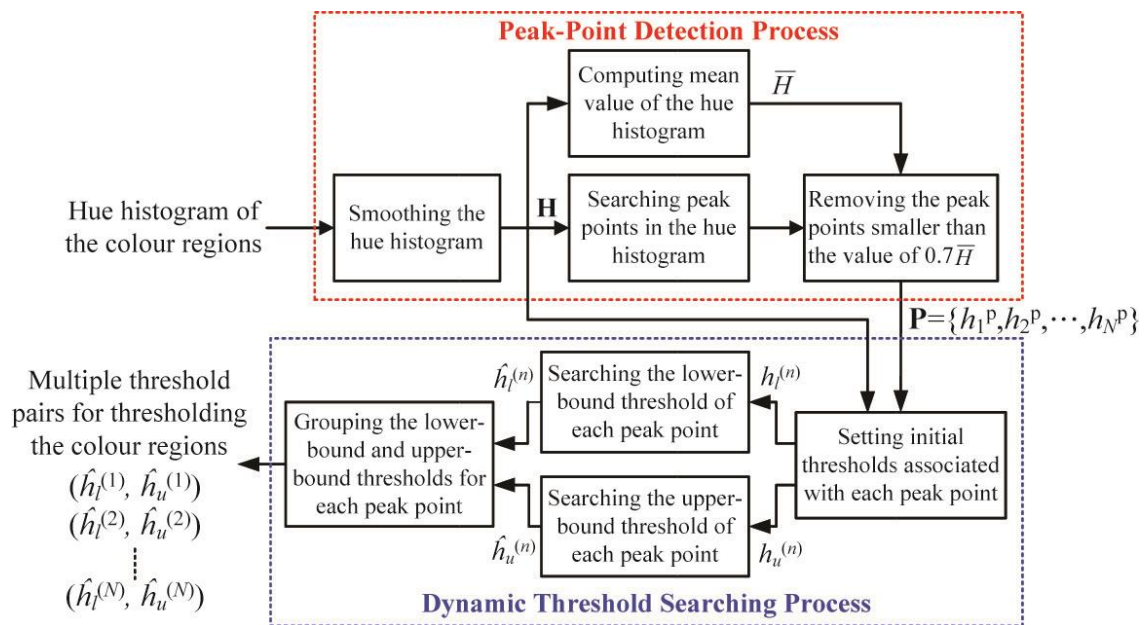


Figure 2. Sơ đồ thuật toán tìm kiếm đa ngưỡng

| Method | Figure | n | $\hat{h}_l^{(n)}$ | $\hat{h}_u^{(n)}$ | $\hat{s}_l^{(n)}$ | $\hat{s}_u^{(n)}$ | $\hat{v}_l^{(n)}$ | $\hat{v}_u^{(n)}$ |
|-----------------|------------|-----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Proposed method | Fig. 13(b) | 1 | 1 | 41 | 0.1843 | 1.0 | 0.0941 | 1.0 |

Figure 3. Khoảng giá trị HSV thu được sau khi thực hiện giải thuật trên

2.1.2. Thresholding

Trong Figure 3, chúng ta có thể thấy rằng phân bố màu da nằm trong khoảng từ 1 đến 40 cho giá trị Hue, 0.1843 đến 1.0 cho giá trị Saturation và 0.0941 đến 1.0 cho giá trị Value.

Sau đó, chúng ta sẽ đổi miền giá trị HSV để phù hợp với Open-CV. Đối với HSV trong Open-CV, miền giá trị Hue là $[0,179]$, miền giá trị Saturation là $[0,255]$ và miền giá trị Value là $[0,255]$. Còn đối với giải thuật trên, miền giá trị Hue là $[0,359]$, miền giá trị Saturation là $[0,1]$ và miền giá trị Value là $[0,1]$

Vì vậy, miền giá trị HSV mới là: 0 đến 20 cho giá trị Hue, 47 đến 255 cho giá trị Saturation và 24 đến 255 cho giá trị Value

2.1.3. Adjusting

Sau đó, ta dùng toán tử hình thái Opening: Erosion sau đó Dilation để lọc nhiễu xung quanh bàn tay.

2.2. Phát hiện bàn tay và các ngón tay

2.2.1. Xác định đường Contour

Sau khi sử dụng màu da để thresholding, ta sẽ đi tìm Contour cho ảnh nhị phân và chọn ra Contour có diện tích lớn nhất (Contour bao quanh viền bàn tay).

2.2.2. Nhận dạng các ngón tay, kẽ tay

Contour được xác định theo kiểu CHAIN_APPROX_NONE là tập hợp tất cả các điểm bao xung quanh bàn tay. Sử dụng toàn bộ tập điểm sẽ ảnh hưởng đến hiệu suất chương trình nên ta tiến hành rút gọn (giảm số lượng điểm biểu diễn) Contour phía trên để xác định các điểm đầu ngón tay – điểm Cực đại hay **Convex**.

Dùng hàm **convexityDefects**, ta xác định được các điểm Cực tiểu – **Defect**, là điểm thấp giữa hai ngón tay.

Số lượng **defect** hàm convexityDefects trả về là số phần tử của list, trong đó, mỗi list chứa 4 thông tin:

Hàm convexityDefect trả về các **defect** là các phần tử trong list. Mỗi phần tử là một vector 4 phần tử: **start_index**, **end_index**, **farthest_index** và **fixpt_depth** là khoảng cách từ **convex** tới **defect**

Tuy nhiên, do đường viền bàn tay là những đường cong mượt mà, lên rồi xuống, ta cần xác định đúng số lượng **defect** thông qua bộ lọc với 2 tiêu chí: góc giữa 2 ngón tay **alpha** và khoảng cách **d** từ **defect** tới đường thẳng nối 2 **convex**

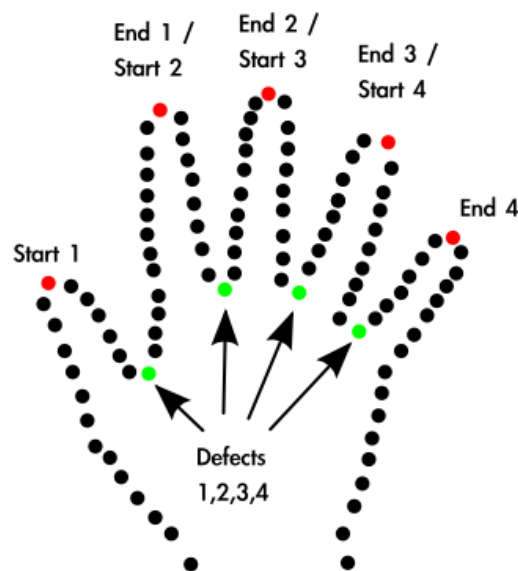


Figure 4. Minh họa việc biểu diễn Convex, Defect và Contour

Giá trị **alpha** ≤ 90 và **d** > 30 được chọn để xác định 1 **defect**.

Tuy nhiên, việc lựa chọn **alpha** và **d** dẫn đến sai sót trong một số trường hợp camera hướng theo phương ngang và thẳng đứng.

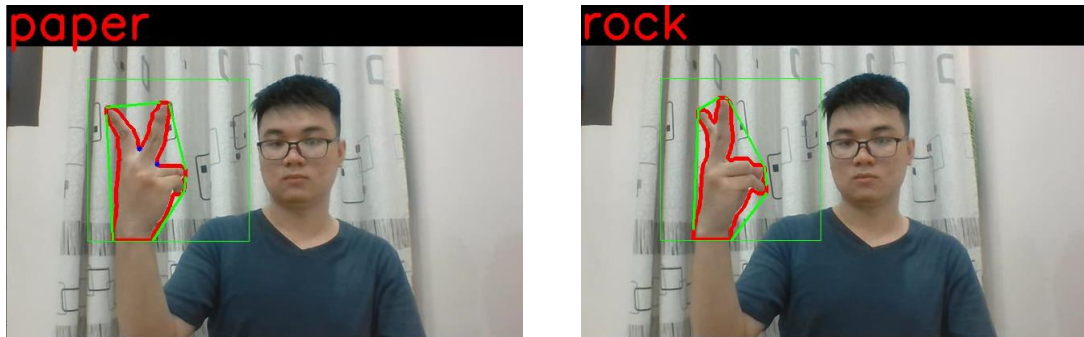


Figure 5. Các trường hợp sai sót trong việc phân biệt

Ở Figure 5 là ví dụ 2 trường hợp sai sót trong việc phân biệt.

- Trường hợp bên phải là do 2 ngón tay tạo thành đường thẳng và khoảng cách giữa các ngón tay không lớn hơn 90 độ nên camera nhận diện được là Búa.
- Trường hợp bên trái là do góc nhìn từ camera cho thấy khoảng cách giữa 2 ngón tay đối với phần còn lại thỏa mãn điều kiện của bài toán nên camera đã nhận diện được 3 ngón tay, do đó kết quả đưa ra là Giấy.

CHƯƠNG III. THỰC NGHIỆM

1. Xác định Contour

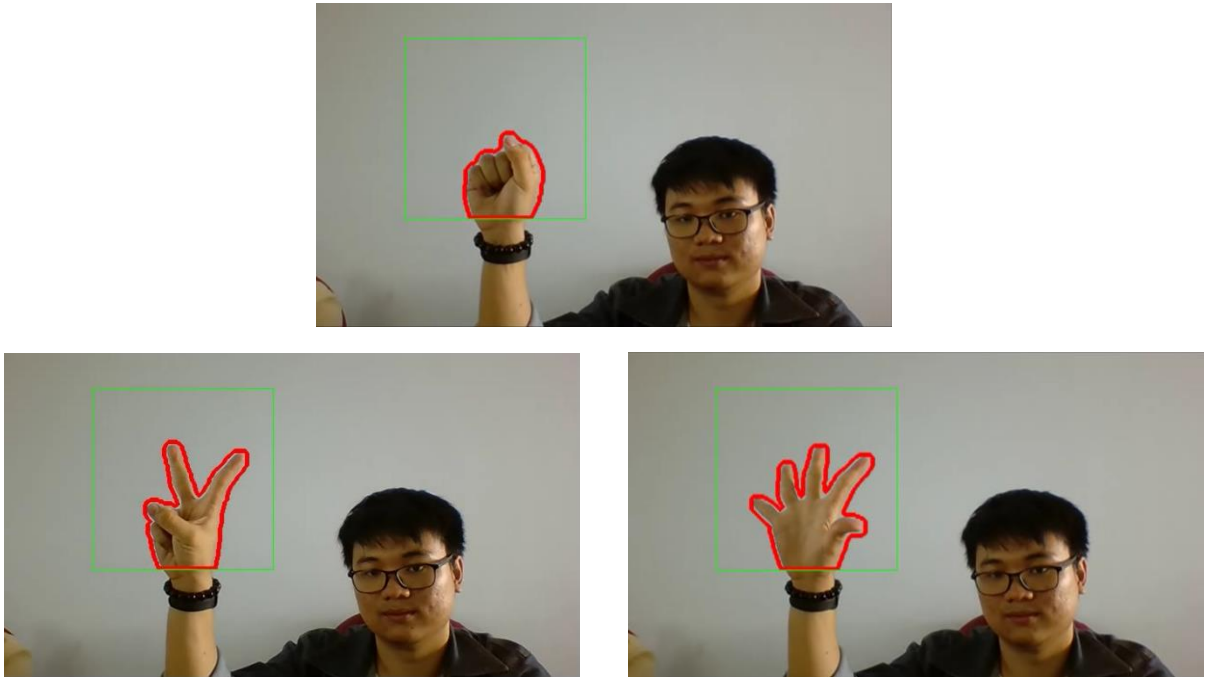


Figure 6. Theo thứ tự Contour của Búa, Kéo và Giấy

2. Xác định Convex Hull

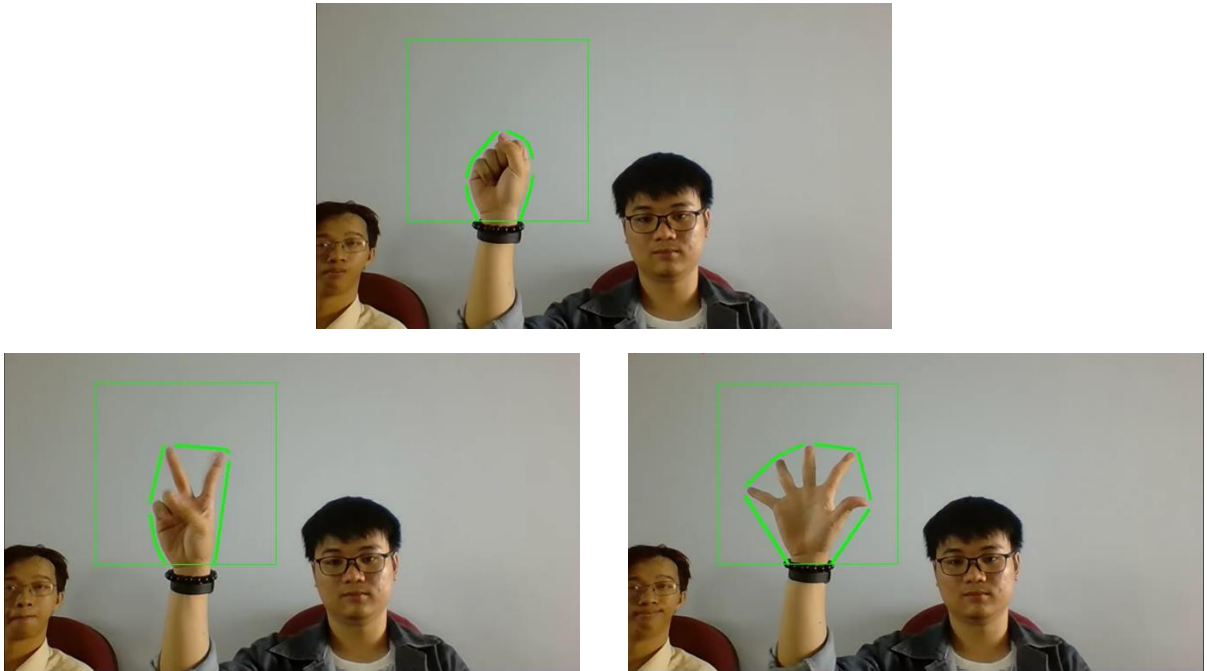


Figure 7. Theo thứ tự Convex Hull của Búa, Kéo và Giấy

3. Ouput

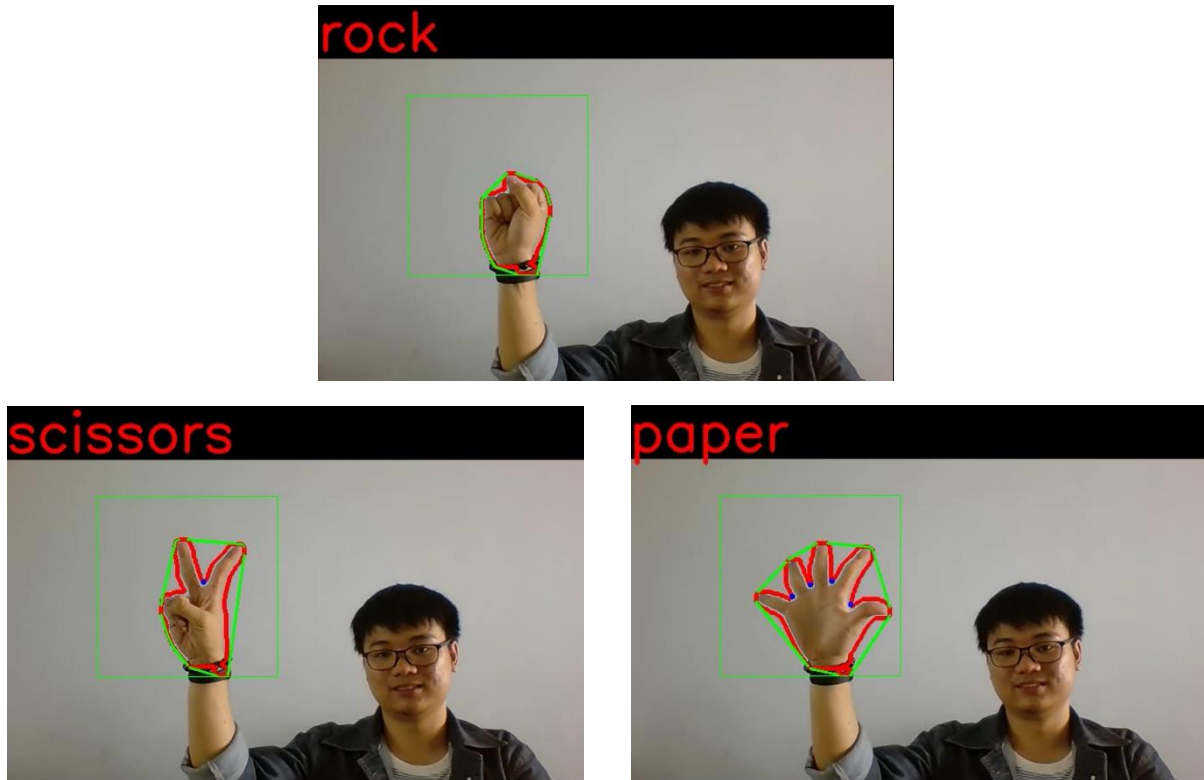


Figure 8. Output sau khi đánh dấu các biểu tượng

CHƯƠNG IV. CẢI TIẾN VÀ HOÀN THIỆN ĐỀ TÀI

Sau lần báo trên lớp, nhóm đã giải thích chi tiết hơn (phần 2.1 Chương II) về cách lựa chọn giá trị màu da người để threshold để có kết quả Segmentation tốt.

CHƯƠNG V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Đề tài có hiệu suất tốt và độ chính xác chấp nhận được.

Tuy nhiên, việc sử dụng thuộc tính màu sắc để segmentation sẽ cho những kết quả sai lệch do nhiễu trong môi trường màu sắc tương đồng với màu da. Chương trình cũng không thể nhận diện chính xác nếu một phần bàn tay bị che khuất.

Để cải thiện độ chính xác, nhóm đề xuất sử dụng kỹ thuật Supervised Learning với dataset từ [rps-cv project](#)

TÀI LIỆU THAM KHẢO

- [1] <https://medium.com/@soffritti.pierfrancesco/handy-hands-detection-with-opencv-ac6e9fb3cec1>
- [2] <https://gogul.dev/software/hand-gesture-recognition-p2>
- [3] Wen-Yuan Chen; Chin-Ho Chung; Sheng-yuan Heish; Cheng-Chao Ku “Rock, Paper and Scissors Image Identification Scheme”