Xây dựng ứng dụng nhận diện hình ảnh bàn tay với phương pháp Bag of visual word, áp dụng vào bài toán tính toán nhanh bằng ngón tay

Nguyễn Luôn Mong Đổ

Ngày 29 tháng 12 năm 2023

Tóm tắt nội dung

Trong bài báo cáo này, bằng phương pháp trích xuất đặc trưng SIFT và Bag of visual word, thí nghiệm sẽ tiến hành xây dựng mô hình nhận diện các hình ảnh bàn tay và áp dụng giải quyết bài toán tính toán nhanh bằng ngón tay.

1 Giới thiệu

1.1 Giới thiệu bài toán

Tính toán nhanh bằng ngón tay (hay còn gọi là finger math) là cách tính nhẩm chỉ với đôi bàn tay. Với cách dạy toán finger math, trẻ sẽ được học cách tính nhẩm cộng trừ trong phạm vi từ 0 tới 99.

Theo phương pháp học toán truyền thống, ở cấp tiểu học, học sinh lớp 2, 3 cộng trừ rất chậm khi con số vượt qua đơn vị 10. Trẻ chỉ được dạy đếm từ 1 đến 10 tương ứng với 10 ngón tay. Nhưng với phương pháp Finger Math, trẻ có thể đếm đến 30, 50 hay 99 rất dễ dàng.

Để các em có hứng thú hơn trong học tập và giúp các cô giáo đỡ vất vả hơn trong quá trình dạy học, việc xây dựng ứng dụng để nhận diện các hình ảnh bàn tay là điều cần thiết trong trường hợp này.

Do đó, bài báo cáo này sẽ tập trung vào việc nhận diện các hình dạng của bàn tay để đưa ra kết quả trong giá trị từ 0 đến 9 tương ứng với giá trị của hình ảnh của bàn tay đó. Từ mô hình xây dựng được, tiến hành phát triển ứng dụng, sử dụng cho bài toán tính toán nhanh bằng tay.

1.2 Giới thiệu về tập dữ liệu

Tập dữ liệu được sử dụng cho bài báo cáo này được xây dựng dựa trên các hình ảnh của bàn tay tương ứng với 10 lớp khác nhau, được đánh số từ 0 đến 9.

Tập dữ liệu bao gồm 277 ảnh sử dụng để huấn luyện mô hình và 77 hình ảnh sử dụng để thực hiện đánh giá độ chính xác của mô hình.

Một số hình ảnh mô tả tập dữ liệu:

Hình 1: Hình ảnh bàn tay ứng với số 1

Hình 2: Hình ảnh bàn tay ứng với số 4

1.3 Mục tiêu của bài báo cáo

Các mục tiêu chính của bài báo cáo bao gồm:

- Phát hiện được hình ảnh bàn tay so với hình ảnh nền.
- Xây dựng, đánh giá kết quả thu được của mô hình phân lớp thu được thông qua việc trích xuất đặc trưng SIFT và phương pháp Bag of visual word.
- Xây dựng được ứng dụng từ mô hình phân lớp đã xây dựng, kết hợp kết quả dự đoán của hai bàn tay để đưa ra kết quả cuối cùng.

2 Phương pháp thực hiện

Bằng việc trích xuất đặc trưng SIFT của các hình ảnh và phương pháp Bag Of Visual, tiến hành xây dựng các mô hình phân lớp và đánh giá kết quả thu được.

2.1 Giới thiệu về phương pháp Bag of visual word

Bag of visual word là một phương pháp được sử dụng để xây dựng bộ mã hóa (embedding) cho hình ảnh. Phương pháp này có thể sử dụng để truy xuất hình ảnh dựa trên nội dung, pháp hiện đối tượng và phân loại hình ảnh.

Việc mã hóa hình ảnh (embedding) bằng phương pháp Bag of visual word được xây dựng dựa trên các bước sau:

- Thực hiện trích xuất đặc trưng của hình ảnh(bài báo cáo này sử dụng phương pháp trích xuất đặc trưng SIFT).

– Tiến hành xây dựng Visual Words.

– Xây dựng các vector tần xuất ứng với từng ảnh dựa trên Visual Words thu được.

Trong phương pháp Bag of visual word, các hình ảnh sẽ được mã hóa thành các visual feature.

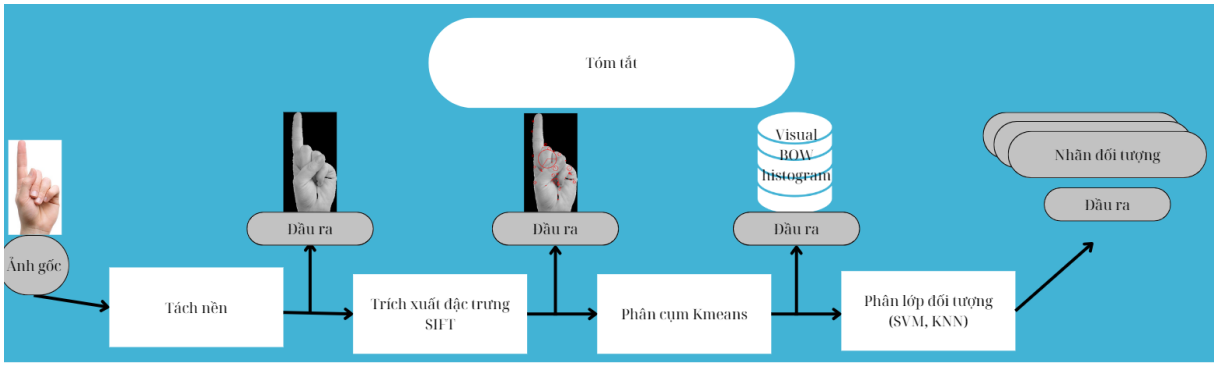
Visual features bao gồm 2 thành phần:

– Keypoints: Những điểm trong hình ảnh không thay đổi nếu ảnh bị xoay, mở rộng hoặc thu nhỏ(bất biến với tỉ lệ).

– Descriptors: Các biểu diễn vector của các keypoints trong một hình ành.

Sau khi thu được visual features, thực hiện xây dựng codebook. Codebook được hiểu là một tập từ vựng chứa tất cả visual features hiện có. Ý tưởng của việc xây dựng visual words là nhóm tất cả các visual features tương tự nhau lại thành các cụm, sử dụng thuật toán phân cụm k-means. Sau khi phân cụm các visual feature sẽ thu được các điểm đại diện cho các cụm đó gọi là các centroid, hay các visual word.

Bước cuối cùng là thực hiện xây dựng các vector histogram của từng hình ảnh thông qua việc đối chiếu các visual feature với các visual word tồn tại trong codebook và xây dựng vector tần xuất dựa trên cơ sở đó.



Hình 3: các bước thực hiện của thuật toán

2.2 Các bước thực hiện

Từ ảnh gốc ban đầu, thực hiện phát hiện vùng màu da trong không gian màu YCbCr dựa vào công thức màu da của Abdellatif Hajraoui và Mohamed Sabri[1] đưa ra để phát hiện ra vùng màu da trong ảnh.

Kết quả công thức màu da trong không gian màu YCbCr:

$$97.5 \leq Cb \leq 142.5 \quad \text{và} \quad 134 \leq Cr \leq 176 \tag{1}$$

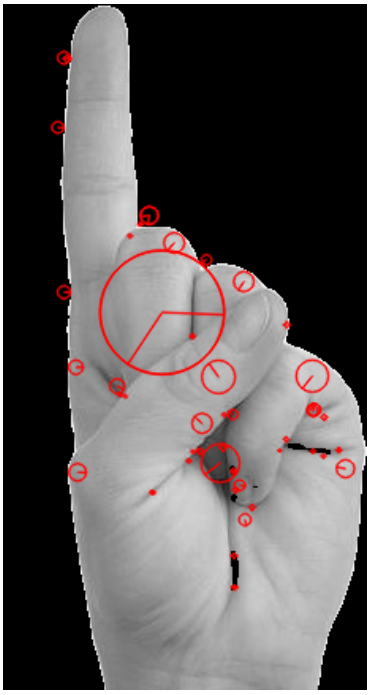
Sau khi thu được kết quả hình ảnh bàn tay đã tách khỏi nền, thực hiện trích xuất đặc trưng SIFT của hình ảnh, kết hợp với thuật toán phân cụm để xây dựng nên codebook và từ đó xây dựng các vector của từng ảnh thông qua việc xây dựng BOW histogram. Xây dựng thuật toán phân lớp dựa trên các vector và đánh giá mô hình.

Kết quả minh họa các bước trên:

Hình 4: Hình ảnh ban đầu

Hình 5: Hình ảnh nhị phân phát hiện vùng màu da

Hình 6: Hình ảnh sau khi tách bàn tay khỏi nền



Hình 7: Hình ảnh với các keypoint của đặc trưng SIFT

2.3 Kết quả thu được

2.3.1 Kết quả với mô hình KNN cho từng giá trị k (số cụm)

Sau khi thử nghiệm với các số cụm khác nhau, ta thu được kết quả độ chính xác của mô hình KNN như sau(kết quả được ghi nhận dựa trên kết quả tốt nhất được tìm thấy bằng cách tối ưu k - số láng giềng):

Bảng 1: Độ chính xác của mô hình KNN tương ứng với số cụm của codebook

Số cụm	100	200	300	500	1000	1600	2000
Độ chính xác	0.23	0.19	0.18	0.25	0.18	0.17	0.19

Nhận xét: Từ bảng 1, độ chính xác của mô hình KNN thu được tương đối thấp, kết quả đều nằm trong phạm vi từ 0.17 đến 0.25. Do đó, hãy tiến hành xây dựng mô hình SVM để so sánh kết quả thu được từ mô hình SVM với kết quả thu được từ mô hình KNN.

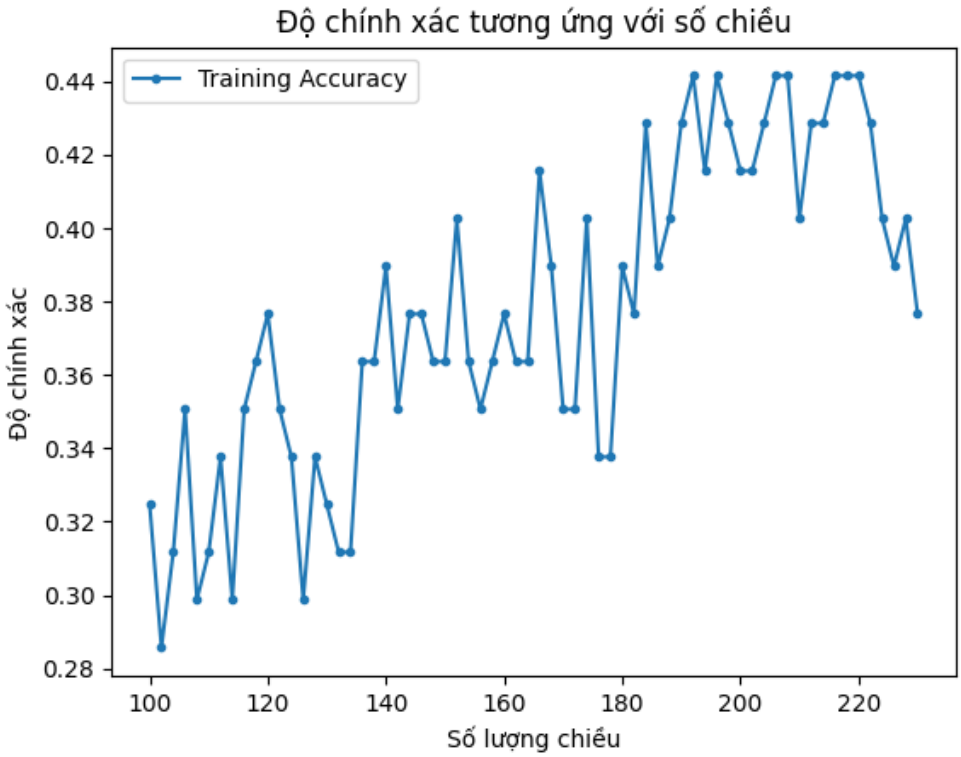
2.3.2 Kết quả với mô hình SVM cho từng giá trị k (số cụm):

Sau khi thử nghiệm với các số cụm khác nhau, ta thu được kết quả độ chính xác của mô hình SVM như sau:

Bảng 2: Độ chính xác của mô hình SVM tương ứng với số cụm của codebook

Số cụm	100	200	300	500	1000	1600	2000
Độ chính xác	0.1818	0.3375	0.2597	0.3506	0.3636	0.3117	0.3766

Nhận xét: Từ bảng 1 và bảng 2, kết quả cho thấy rằng mô hình SVM cho độ chính xác cao hơn mô hình KNN với cùng số cụm của codebook. Do vậy, các kết quả sau đây dựng tiến hành dựa trên số cụm của codebook được lựa chọn là 1000 và thực hiện trên mô hình được chọn là mô hình SVM



Hình 8: Độ chính xác của mô hình tương ứng với số chiều của mỗi vector ảnh sau khi giảm số chiều từ 1000

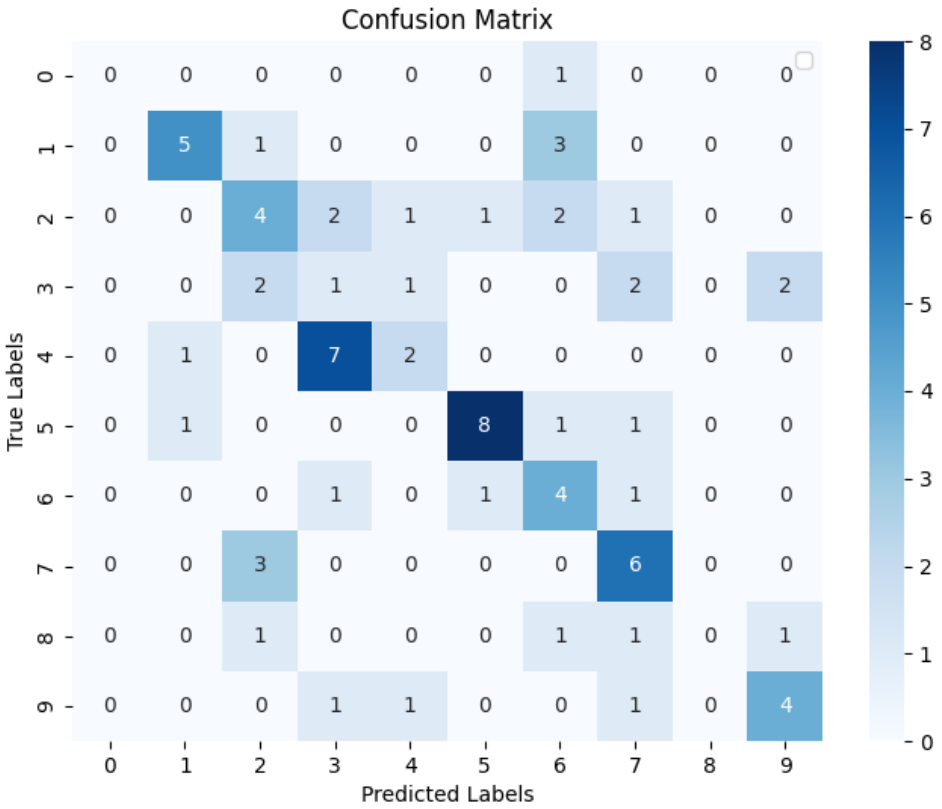
Nhận xét:

- Theo kết quả từ Bảng 2 kết quả trước khi giảm số chiều của vector embedding cho từng ảnh với số cụm 1000 là 0.3636. Sau khi sử dụng phương pháp PCA để giảm số chiều đã thu được kết quả như hình hình 7

- Dựa vào hình 7, kết quả thu được cho thấy rằng độ chính xác của mô hình đã được cải thiện. Với số chiều của dữ liệu là 220, độ chính xác của mô hình cải thiện từ 0.3636 lên 0.4416.

2.3.3 Phân tích, đánh giá kết quả thu được

Phân tích, đánh giá kết mô hình SVM thu được bằng phương pháp PCA, giảm số chiều các vector embedding từ 1000 xuống 220. Kết quả ma trận nhầm lẫn thu được với mô hình SVM:



Hình 9: Ma trận nhầm lẫn thu được từ mô hình SVM với phương pháp PCA áp dụng giảm số chiều từ 1000 xuống 220

Phân tích kết quả thu được của ma trận nhầm lẫn tại hình 8:

- Kết quả thu được tại lớp 1 cho thấy mô hình có sự nhầm lẫn khi dự đoán nhầm lớp 1 vào lớp 6 và lớp 2. Cụ thể mô hình nhầm lẫn lớp 1 vào lớp 2 một ảnh và nhầm lẫn vào lớp 6 ba ảnh.

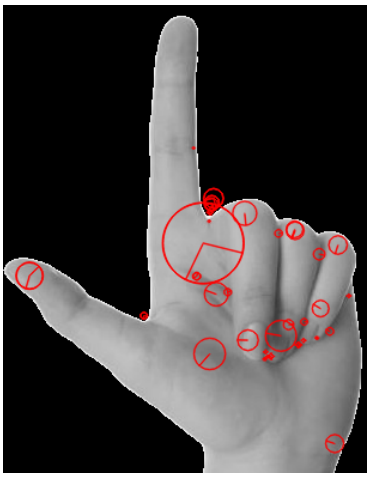
- Nguyên nhân có sự nhầm lẫn trong trường hợp này là vì hình ảnh hai lớp 2 và 6 tương tự với lớp 1, được minh họa trong hình 9

- Kết quả thu được tại lớp 4 cho thấy rằng mô hình dự đoán sai hầu hết kết quả ở lớp này. Mô hình chỉ dự đoán đúng hai trong tổng số mười ảnh đầu vào. Mô hình dự đoán nhầm lẫn lớp 4 vào lớp 3 tổng cộng bảy ảnh. - Nguyên nhân cho sự nhầm lẫn này đến từ việc hai lớp 3 và 4 có hình ảnh tương tự nhau. được minh họa trong hình 10

- Tương tự cho sự nhầm lẫn của mô hình khi nhãn thực sự là lớp 7 nhưng mô hình lại dự đoán vào lớp 2, được minh họa hình ảnh hai lớp như hình 11

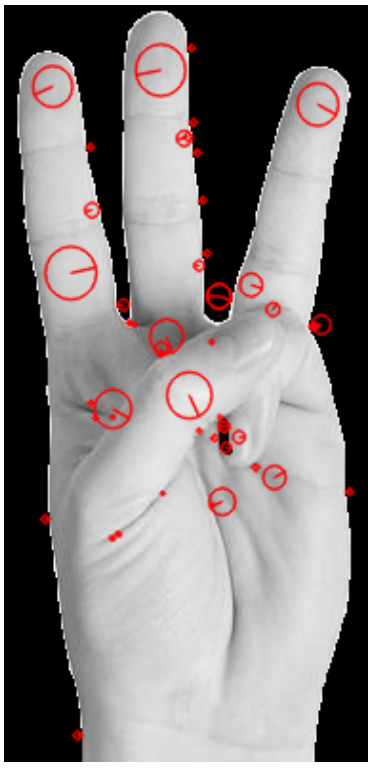
(a) Hình ảnh lớp 1

(b) Hình ảnh lớp 2

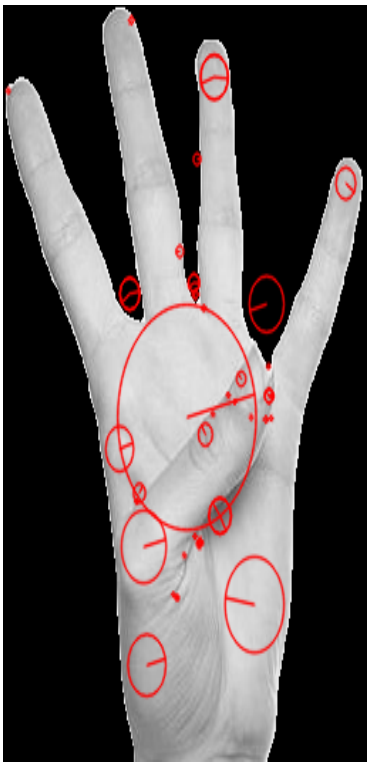


(c) hình ảnh lớp 6

Hình 10: hình ảnh các lớp 1, 2, 6 với các keypoint

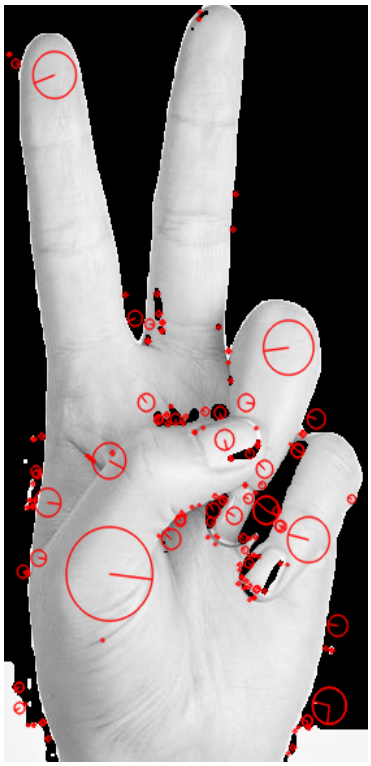


(a) Hình ảnh lớp 3



(b) Hình ảnh lớp 4

Hình 11: hình ảnh các lớp 3, 4 với các keypoint

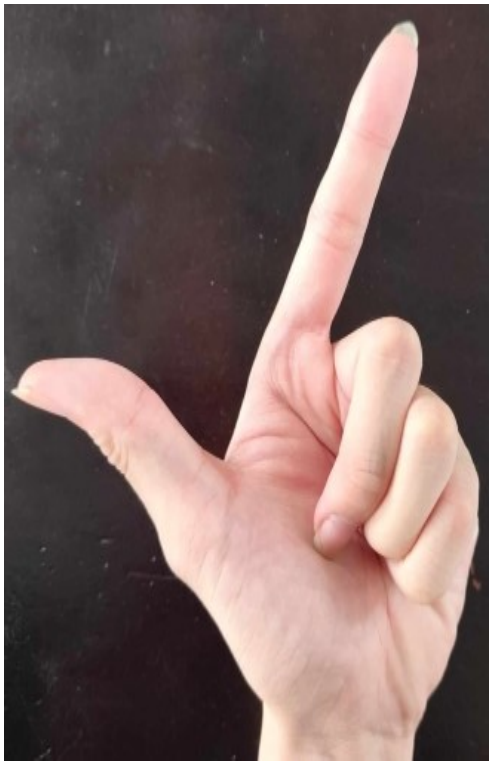


(a) Hình ảnh lớp 2

(b) Hình ảnh lớp 7

Hình 12: hình ảnh các lớp 2, 7 với các keypoint

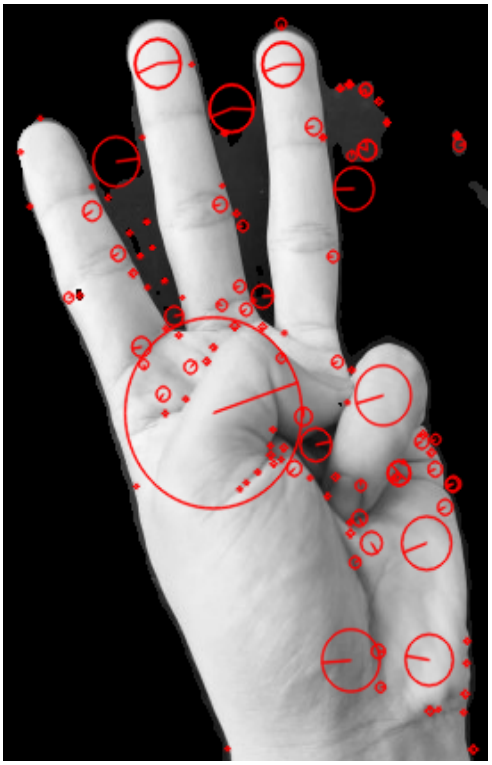
Độ chính xác của mô hình SVM thu được là 0.4416 như hình 7. Nguyên nhân độ chính xác của mô hình thấp đến từ việc trong quá trình loại bỏ nền bằng phương pháp pháp hiện màu da, một số hình ảnh không thể loại bỏ được hết nền ra khỏi hình ảnh bàn tay, dẫn đến việc trích xuất đặc trưng SIFT của hình ảnh không được chính xác. Minh họa kết quả tách nền bằng phương pháp phát hiện màu da của các trường hợp lỗi được mô tả như hình dưới:



(a) Hình ảnh trước khi tách nền

(b) Hình ảnh sau khi tách nền

(c) Hình ảnh trước khi tách nền



(d) Hình ảnh sau khi tách nền

Hình 13: Hình ảnh lỗi từ việc tách nền bằng phương pháp phát hiện màu da

3 Kết luận

3.1 Các kết quả đạt được

Từ bảng 1 và bảng 2, kết quả cho thấy rằng mô hình phân lớp SVM cho độ chính xác cao hơn so với mô hình phân lớp KNN. Khi thực hiện PCA giảm chiều của các vector đặc trưng cho ảnh từ 1000 chiều xuống còn 220 chiều, độ chính các được cải thiện từ 0.3636 lên 0.4416, theo hình 7

Từ ma trận nhầm lẫn hình 8, kết quả cho thấy rằng các hình dạng bàn tay tương tự nhau sẽ khiến mô hình dự đoán nhầm lẫn. Các lớp nhầm lẫn với nhau nhiều nhất: lớp 1 nhầm vào lớp 6, lớp 4 nhầm vào lớp 3,...

Phương pháp phát hiện màu da để tách nền đối tượng bàn tay khỏi nền trong một số trường hợp không thể tách hoàn toàn hình nền khỏi đối tượng, hình 12, gây ra việc các keypoint nhiễu vẫn tồn tại trong ảnh, ảnh hưởng đến độ chính xác của mô hình. Do vậy tùy thuộc vào màu da và màu nền của đối tượng, phải điều chỉnh các ngưỡng quy định trong công thức(1) phù hợp với tập dữ liệu sử dụng.

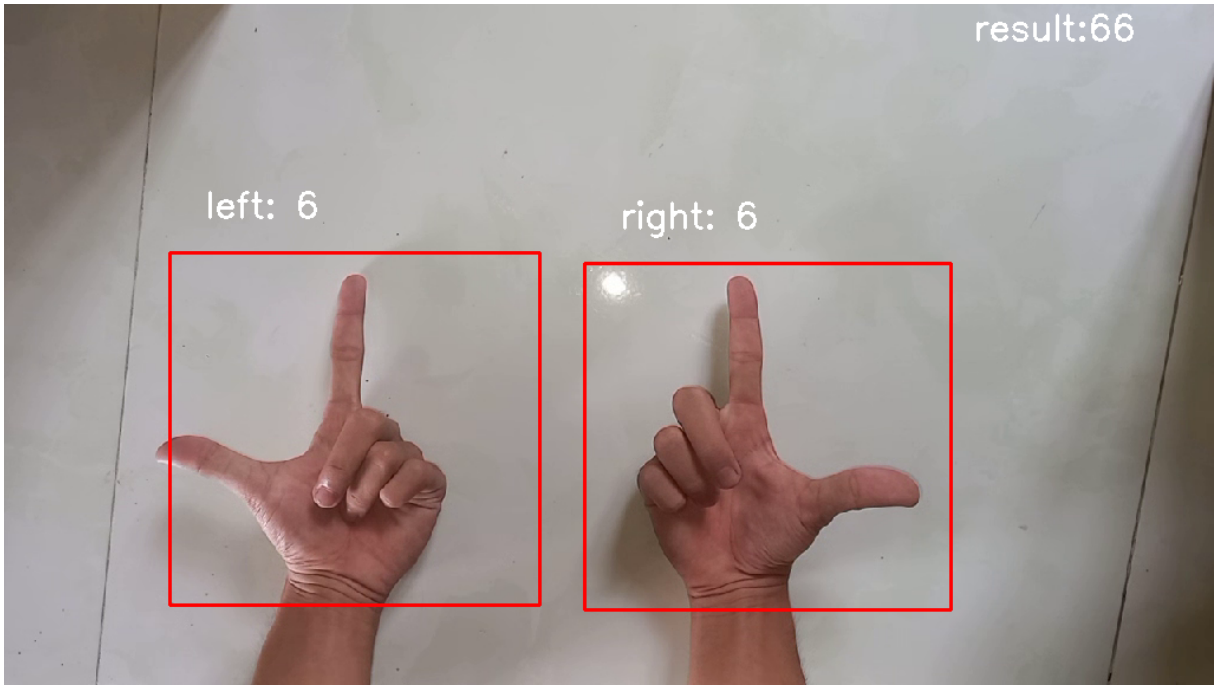
4 Xây dựng ứng dụng

4.1 Các ràng buộc liên quan đến ứng dụng

Để việc xây dựng ứng dụng trở nên đơn giản và đạt hiệu quả với mô hình đã xây dựng, ứng dụng có những ràng buộc sau:

- Hình ảnh chỉ chứa 2 bàn tay, với hình ảnh nền là đồng nhất.

- Để ứng dụng có thể hoạt động với độ chính xác cao nhất, ứng dụng quy định các vị trí cụ thể cho tay trái và tay phải. Trạng thái để ứng dụng cho kết quả tốt là khi bàn tay ở trạng thái ngửa. Minh họa cho các ràng buộc trong hình 13:



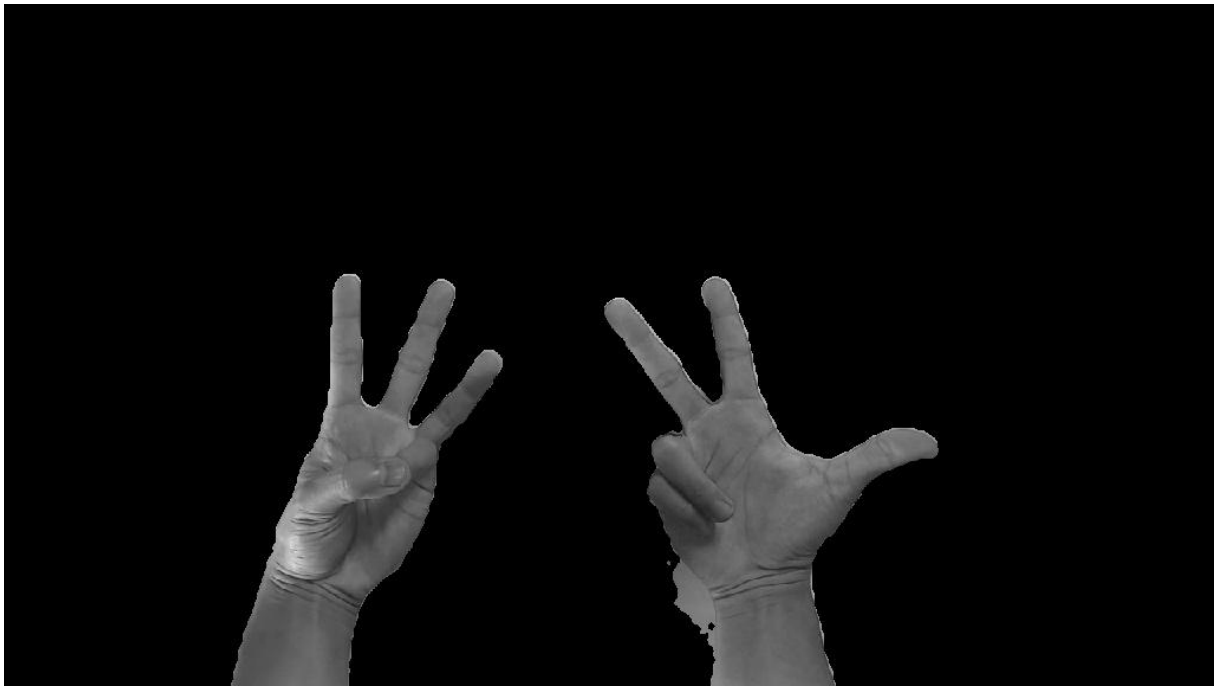
Hình 14: Hình ảnh minh họa ứng dụng nhận diện hình dạng bàn tay

4.2 Các bước thực hiện

Bước 1: Với hình ảnh thu được từ camera, tiến hành tách nền dựa vào phương pháp phát hiện màu da như hình 14

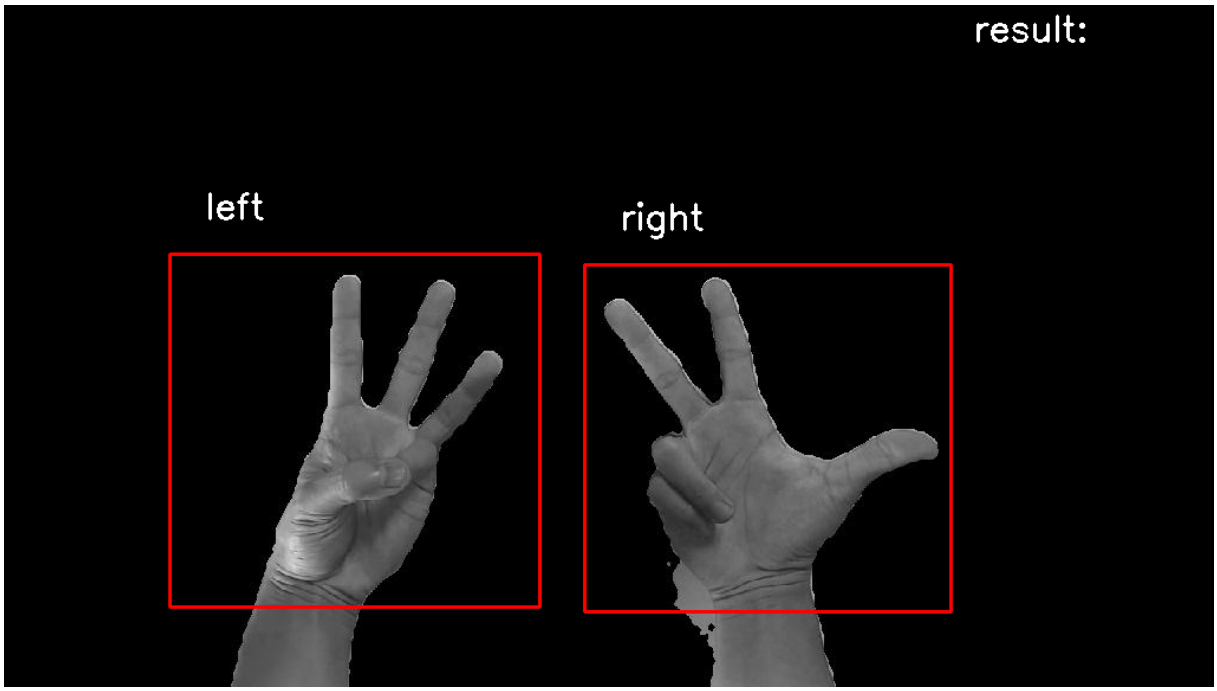
Bước 2: Quy định các vị trí đặt tay phải và tay trái, sau đó tiến hành tách các hình dạng bàn tay để đưa vào mô hình SVM đã xây dựng, minh họa trong hình 15

(a) Hình ảnh trước khi tách nền



(b) Hình ảnh sau khi tách nền

Hình 15: Kết quả tách nền của hình ảnh



(a) Quy định vị trí đặt tay

(b) Hình ảnh tay trái trích xuất được

(c) Hình ảnh tay phải trích xuất được

Hình 16: Kết quả trích xuất hình ảnh làm đầu vào cho mô hình

Bước 3: Sau khi tách được hình ảnh tay trái và tay phải từ ảnh gốc ban đầu, tiến hành dự đoán giá trị ứng với hình dạng của bàn tay và đưa ra kết quả của từng bàn tay cũng như kết quả cuối cùng biểu biễn bởi 2 bàn tay. Được minh họa như hình 13

5 Tài liệu tham khảo

[1] Abdellatif Hajraoui, Mohamed Sabri, "Face Detection Algorithm based on Skin Detection, Watershed Method and Gabor Filters", May 2014