

**ĐẠI HỌC PHENIKAA**  
**TRƯỜNG CÔNG NGHỆ THÔNG TIN PHENIKAA**



**MÔN HỌC: HỌC MÁY 1-3-24(N02)**  
**ĐỀ TÀI: “NHẬN DIỆN KHUÔN MẶT SỬ DỤNG MACHINE LEARNING”**

**Thành viên nhóm**

<b>Nguyễn Văn Thắng</b>	<b>23010572</b>	<b>K17-KHMT(AI&amp;KHDL)_1</b>
<b>Phạm Văn Sự</b>	<b>23010523</b>	<b>K17-KHMT(AI&amp;KHDL)_1</b>
<b>Đặng Anh Tuyền</b>	<b>23010912</b>	<b>K17-KHMT(AI&amp;KHDL)_1</b>

**NHÓM 9**

**GIÁO VIÊN GIẢNG DẠY: PGS.TS PHẠM TIẾN LÂM**

**ĐẠI HỌC PHENIKAA**  
**TRƯỜNG CÔNG NGHỆ THÔNG TIN PHENIKAA**



**MÔN HỌC: HỌC MÁY 1-3-24(N02)**

**ĐỀ TÀI: “NHẬN DIỆN KHUÔN MẶT SỬ DỤNG MACHINE  
LEARNING”**

**Thành viên nhóm**

Họ và tên	Mã sinh viên	Điểm bằng số	Điểm bằng chữ	Đóng góp
Nguyễn Văn Thăng	23010572			100%
Phạm Văn Sự	23010523			100%
Đặng Anh Tuyền	23010912			100%

Chữ ký giám thị 1	Chữ ký giám thị 2

**GIÁO VIÊN GIẢNG DẠY: PGS.TS PHẠM TIẾN LÂM**

# Bảng phân công công việc

STT	Thành Viên	Nhiệm vụ
1	Nguyễn Văn Thắng (trưởng nhóm)	Phân tích yêu cầu, thiết kế mô hình các mô hình, huấn luyện mô hình, tìm hiểu tài liệu, viết phần giới thiệu, phương pháp và trình bày báo cáo.
2	Phạm Văn Sự	Xử lý dữ liệu, huấn luyện mô hình, tìm hiểu tài liệu, viết phần phân tích dữ liệu và trình bày báo cáo.
3	Đặng Anh Tuyên	Thu thập dữ liệu, huấn luyện mô hình, viết phần các mô hình học máy, kết quả, chạy mô hình trong thời gian thực và kết luận.

# Mục lục

<b>Phần 1. Giới thiệu .....</b>	<b>6</b>
<b>Phần 2. Phân tích dữ liệu .....</b>	<b>7</b>
2.1. Thu thập dữ liệu .....	7
2.2. Xử lý dữ liệu.....	9
2.3. Phát hiện và trích xuất đặc trưng khuôn mặt .....	10
2.4. Mã hóa nhãn .....	10
2.5. Lưu trữ dữ liệu .....	10
2.6. Trục quan hóa dữ liệu .....	10
<b>Phần 3. Các mô hình học máy .....</b>	<b>12</b>
<b>3.1. K-Nearest Neighbors (KNN) .....</b>	<b>12</b>
3.1.1. K-Nearest Neighbors Model with Extracted Features (knn_ex) .....	13
3.1.2. K-Nearest Neighbors Model with Flattened Image Features (knn_flat) .....	14
<b>3.2. Random forest (RF) .....</b>	<b>16</b>
3.2.1. Random Forest Model with Extracted Features (rf_ex) .....	17
3.2.2. Random Forest Model with Flattened Image Features (rf_flat) .....	18
<b>3.3. Support Vector Machine (SVM).....</b>	<b>20</b>
3.3.1. Support Vector Machine Model with Extracted Features (svm_ex) .....	20
3.3.2. Support Vector Machine Model with Flattened Image Features (svm_flat) ....	22
<b>Phần 4. Kết quả.....</b>	<b>23</b>
<b>4.1. K-Nearest Neighbors (KNN) .....</b>	<b>23</b>
4.1.1. K-Nearest Neighbors Model with Extracted Features (knn_ex) .....	23
4.1.2. K-Nearest Neighbors Model with Flattened Image Features (knn_flat) .....	27
<b>4.2. Random forest (RF) .....</b>	<b>31</b>
4.2.1. Random Forest Model with Extracted Features (rf_ex) .....	31
4.2.2. Random Forest Model with Flattened Image Features (rf_flat) .....	35
<b>4.3. Support Vector Machine (SVM).....</b>	<b>40</b>
4.3.1. Support Vector Machine Model with Extracted Features (svm_ex) .....	40
4.3.2. Support Vector Machine Model with Flattened Image Features (svm_flat) ....	44
<b>4.4. So sánh.....</b>	<b>48</b>

<b>Phần 5. Chạy mô hình trong thời gian thực.....</b>	<b>49</b>
<b>Phần 6. Kết luận.....</b>	<b>50</b>

## Phần 1. Giới thiệu

Nhận diện khuôn mặt là một trong những ứng dụng tiêu biểu và thực tiễn nhất của lĩnh vực thị giác máy tính (Computer Vision) và học máy (Machine Learning). Đề tài này tập trung vào việc xây dựng một hệ thống nhận diện khuôn mặt từ ảnh đầu vào, sử dụng các mô hình học máy truyền thống như K-Nearest Neighbors (KNN), Support Vector Machine (SVM) và Random Forest (RF).

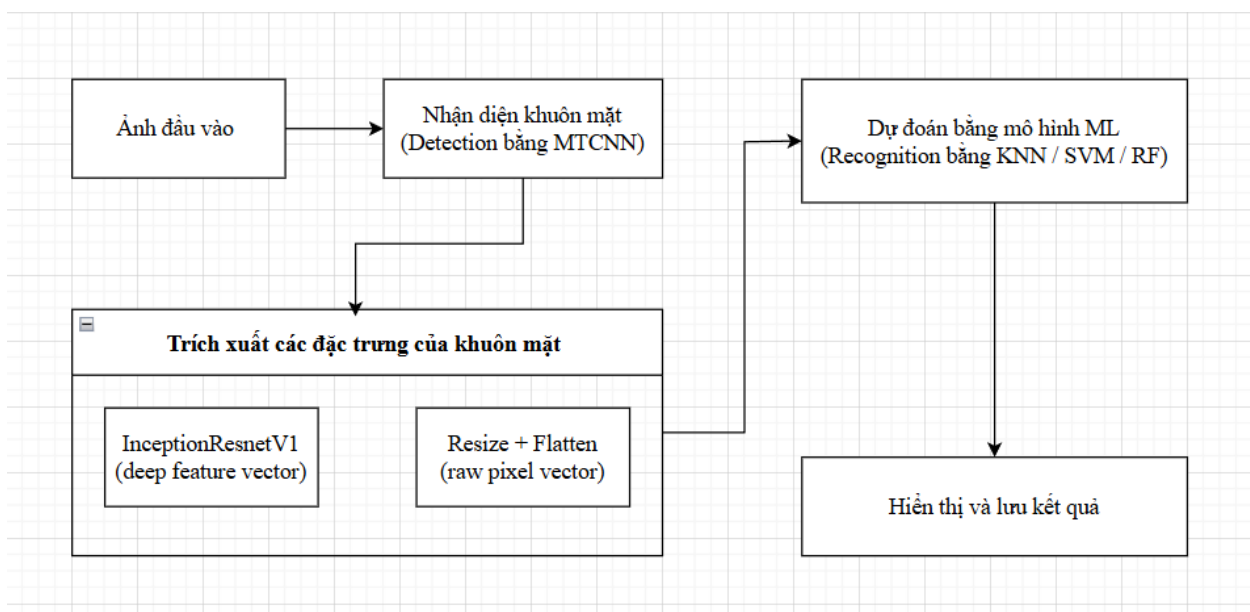
Để nâng cao hiệu quả nhận diện, hệ thống áp dụng hai cách tiếp cận chính:

1. Sử dụng đặc trưng trích xuất từ mô hình deep learning (InceptionResnetV1 – pretrained trên VGGFace2).
2. Sử dụng trực tiếp ảnh đầu vào được xử lý và làm phẳng (flatten) để huấn luyện mô hình.

Các bước chính trong quá trình nhận diện bao gồm:

- Phát hiện khuôn mặt trong ảnh bằng MTCNN.
- Tiền xử lý và trích xuất đặc trưng khuôn mặt (bằng mô hình deep learning hoặc flatten ảnh).
- Dự đoán nhãn khuôn mặt thông qua mô hình đã huấn luyện.
- Hiển thị kết quả nhận diện kèm theo nhãn dự đoán trên ảnh.

Quy trình trên được minh họa rõ ràng thông qua sơ đồ dưới đây:



Ngoài ra, hệ thống còn hỗ trợ phân loại “Unknown” đối với những khuôn mặt không nằm trong tập huấn luyện, thông qua ngưỡng độ tương đồng (cosine similarity).

Đề tài này giúp chúng ta biết được:

- Hiểu rõ hơn về ứng dụng thực tiễn của các thuật toán Machine Learning trong xử lý và nhận diện khuôn mặt.
- Đánh giá và so sánh hiệu năng giữa các mô hình học máy truyền thống (KNN, SVM, RF) khi sử dụng các loại dữ liệu đầu vào khác nhau (đặc trưng deep learning và ảnh flatten).
- Xây dựng được quy trình chuẩn để triển khai các hệ thống nhận diện khuôn mặt trong các ứng dụng cụ thể như điểm danh, giám sát an ninh, và cá nhân hóa trải nghiệm người dùng.
- Có khả năng mở rộng hệ thống để tích hợp thêm các thuật toán mới hoặc cải tiến hiệu suất thông qua điều chỉnh mô hình và tham số huấn luyện.

## Phần 2. Phân tích dữ liệu

### 2.1. Thu thập dữ liệu

- Dữ liệu thô ban đầu gồm khoảng 7000 ảnh, được thu thập bằng cách chụp các bức ảnh chứa số lượng khuôn mặt khác nhau (từ 0 đến 6 khuôn mặt mỗi ảnh) bằng camera. Để đảm bảo tính đa dạng, các bức ảnh được ghi lại dưới nhiều góc xoay khuôn mặt, điều kiện ánh sáng, và vị trí khác nhau. Mỗi mức số lượng khuôn mặt đều được chụp 1000 ảnh.

Label: Thang



Label: Su



Label: Nhung



Label: Tuyen



Label: Vu



Label: Dat



Label: Huy





## 2.2. Xử lý dữ liệu

Các bước tiền xử lý dữ liệu ảnh khuôn mặt:

**- Bước 1:** Loại bỏ ảnh không đạt chất lượng

- + Xóa các ảnh bị nhòe, thiếu sáng, hoặc không rõ khuôn mặt.
- + Lọc bỏ các ảnh không có khuôn mặt hoặc ảnh lặp lại trong tập dữ liệu.

**- Bước 2:** Giảm số lượng ảnh cho mỗi lớp

- + Giảm số lượng ảnh từ khoảng 1000 ảnh ban đầu xuống còn khoảng 100 ảnh cho mỗi lớp (ứng với từng người).
- + Đảm bảo mỗi lớp có số lượng ảnh đồng đều, phù hợp cho bài toán phân loại.

**- Bước 3:** Căn chỉnh và chuẩn hóa kích thước ảnh

- + Tất cả các ảnh còn lại được căn chỉnh và chuyển về cùng một kích thước tiêu chuẩn (ví dụ: 160x160 pixels).

**- Bước 4:** Tăng cường và cải thiện chất lượng ảnh

- + Thực hiện các thao tác như cân bằng sáng, tăng độ tương phản để cải thiện chất lượng hình ảnh.
- + Cắt tỉa chính xác vùng chứa khuôn mặt, loại bỏ các phần nền thừa.

**- Bước 5:** Tạo tập dữ liệu sau xử lý

- + Kết quả thu được là một tập dữ liệu đã tinh gọn, gồm khoảng 700 ảnh đã qua xử lý, sẵn sàng cho bước trích xuất đặc trưng và huấn luyện mô hình nhận diện khuôn mặt.

Label: Thang



Label: Su



Label: Nhung



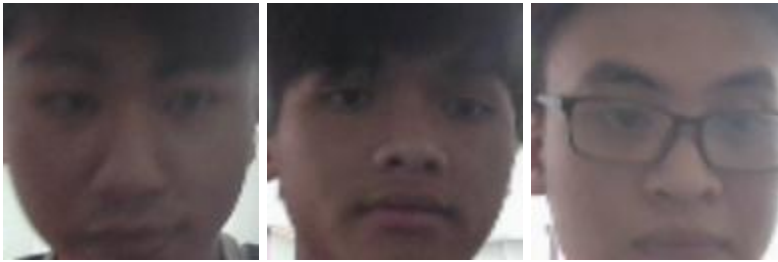
Label: Tuyen



Label: Vu

Label: Dat

Label: Huy



### 2.3. Phát hiện và trích xuất đặc trưng khuôn mặt

- Sử dụng MTCNN để phát hiện khuôn mặt trong ảnh (detector = MTCNN()).
- Sau khi phát hiện, các khuôn mặt được cắt ra và resize về kích thước phù hợp: **160×160** cho trích xuất đặc trưng bằng CNN, **128×128** cho phương pháp làm phẳng ảnh.
- Nếu dùng feature extractor (InceptionResnetV1), ảnh được chuẩn hóa và chuyển thành tensor, sau đó trích xuất đặc trưng bằng mô hình deep learning (feature\_extractor(face\_tensor)).
- Nếu dùng phương pháp flatten, ảnh được làm phẳng thành vector 1D (image.flatten()).

### 2.4. Mã hóa nhãn

- Mỗi thư mục con trong thư mục dữ liệu tương ứng với một nhãn số (0 đến 6), ví dụ: { 0: 'Thang', 1: 'Su', 2: 'Nhưng', 3: 'Tuyen', 4: 'Vu', 5: 'Dat', 6: 'Huy' }
- Khi duyệt ảnh, nhãn số được gán cho từng ảnh và lưu vào list labels\_list.

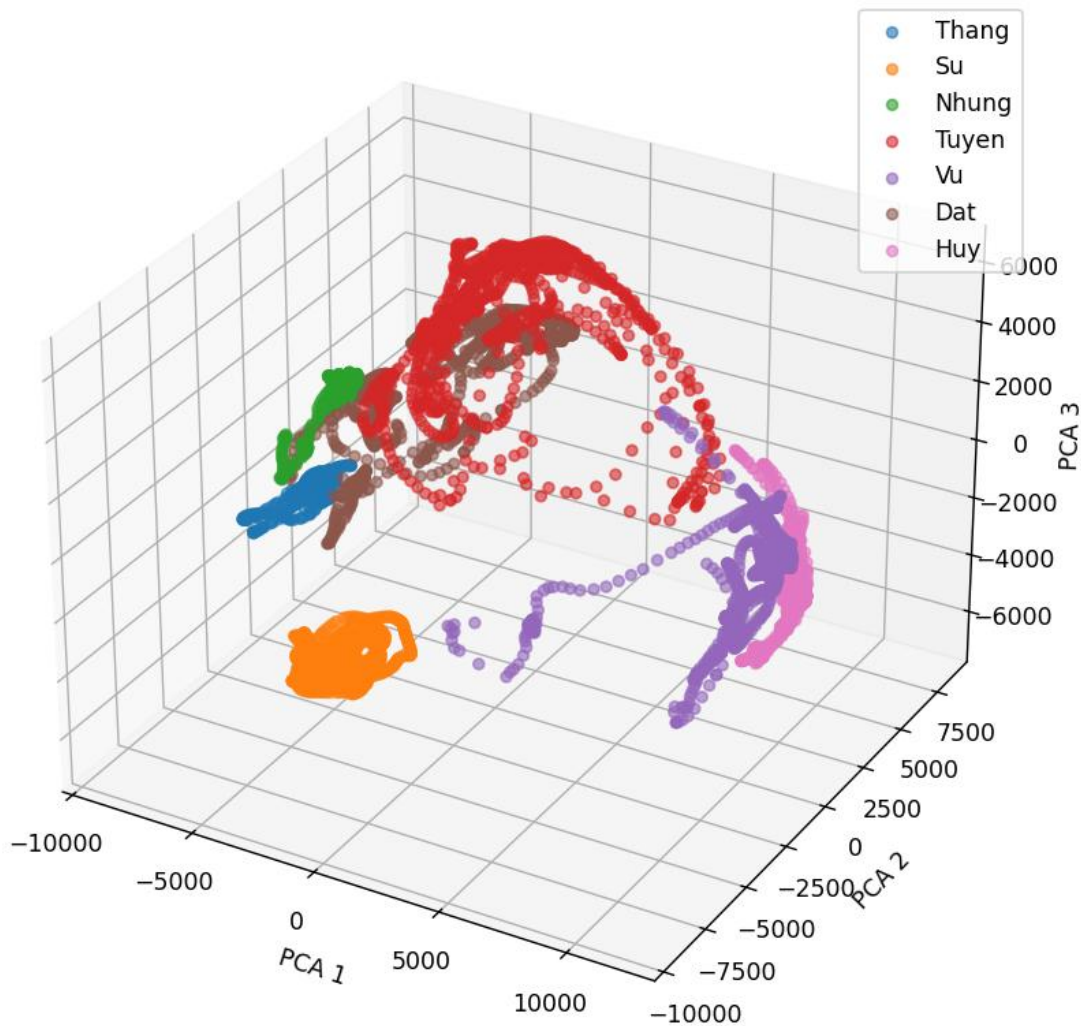
### 2.5. Lưu trữ dữ liệu

- Đặc trưng sau khi trích xuất được lưu vào file /weights/data\_features.npy
- Mô hình sau khi huấn luyện sẽ dùng hàm joblib.dump(model, model\_path) để lưu vào /weights/tên\_mô\_hình\_huấn\_luyện.joblib
- Điều này giúp bạn không phải trích xuất lại đặc trưng hoặc huấn luyện lại mô hình từ đầu mỗi lần chạy.

### 2.6. Trực quan hóa dữ liệu

Biểu đồ PCA 3D cho thấy các mẫu ảnh khuôn mặt của 7 đối tượng trong tập dữ liệu được phân bố thành các cụm riêng biệt khá rõ ràng. Việc giảm chiều dữ liệu từ hàng trăm đặc trưng xuống còn 3 thành phần chính đã giúp trực quan hóa tốt khả năng phân loại của mô hình đối với từng cá nhân. Các cụm có màu sắc phân biệt, giúp dễ dàng quan sát mức độ tách biệt giữa các đối tượng.

3D PCA Visualization of Dataset



**- Điểm mạnh:**

+ **Khả năng phân tách tốt:** Phần lớn các cụm dữ liệu của từng đối tượng không bị chồng lấn nhau, cho thấy đặc trưng đã phản ánh tốt sự khác biệt giữa các khuôn mặt.

+ **Phân bố đều trong không gian:** Các cụm trải rộng trong không gian 3D, không bị ép vào mặt phẳng, chứng tỏ ba thành phần chính của PCA giữ lại nhiều thông tin quan trọng từ dữ liệu gốc.

+ **Tính trực quan cao:** Biểu đồ hỗ trợ rất tốt cho việc kiểm tra trực quan độ phân biệt giữa các lớp, giúp đánh giá chất lượng dữ liệu trước khi huấn luyện mô hình.

**- Điểm yếu:**

+ **Một số cụm gần nhau hoặc giao nhau nhẹ:** Có sự chồng lấn nhất định giữa một số đối tượng như Tuyền và Đạt, hoặc Nhung và Đạt. Điều này có thể ảnh hưởng đến độ chính xác khi mô hình phân loại các khuôn mặt tương đồng.

+ **Biến thiên trong nội lớp:** Một vài cụm như cụm màu đỏ (Tuyền) có hình dạng dàn trải và phức tạp, cho thấy sự đa dạng cao trong dữ liệu nội lớp. Điều này có thể khiến mô hình khó học được ranh giới phân loại rõ ràng.

- **Kết luận:** Biểu đồ PCA 3D cho thấy tập dữ liệu khuôn mặt có chất lượng tốt với khả năng phân biệt giữa các đối tượng tương đối rõ ràng. Dù còn một số cặp đối tượng có vùng dữ liệu gần nhau, nhưng nhìn chung, các đặc trưng đầu vào đã đủ mạnh để mô hình học được sự khác biệt giữa các khuôn mặt. Việc cải thiện thêm dữ liệu đầu vào hoặc sử dụng mô hình học sâu có thể giúp giảm hiện tượng chồng lấn và nâng cao độ chính xác của hệ thống nhận diện.

## Phần 3. Các mô hình học máy

### 3.1. K-Nearest Neighbors (KNN)

**K-Nearest Neighbors (KNN)** là một thuật toán học máy cơ bản nhưng rất hiệu quả, thường được ứng dụng trong các bài toán phân loại và hồi quy. Nguyên lý hoạt động của KNN khá trực quan: khi cần phân loại một điểm dữ liệu mới, thuật toán sẽ tìm kiếm những “hàng xóm” gần nhất trong tập huấn luyện – tức là các điểm dữ liệu có đặc trưng gần với điểm mới này nhất. Ý tưởng cốt lõi là những điểm dữ liệu giống nhau thường nằm gần nhau trong không gian đặc trưng.

Cụ thể, khi gặp một điểm mới, KNN sẽ tính toán khoảng cách từ điểm đó đến tất cả các điểm trong tập dữ liệu, sau đó chọn ra K điểm gần nhất. Dựa vào nhãn của các điểm lân cận này, thuật toán sẽ “bỏ phiếu” để quyết định nhãn của điểm mới – nhãn nào xuất hiện nhiều nhất sẽ được chọn. Giá trị K càng lớn thì mô hình càng ổn định, nhưng nếu quá lớn có thể làm giảm độ chính xác do bao gồm cả các điểm xa. Vì vậy, việc lựa chọn K là yếu tố then chốt để cân bằng giữa sự ổn định và tính chính xác của mô hình.

Trong KNN, có ba siêu tham số quan trọng:

- **n\_neighbors (K):** Số lượng điểm lân cận gần nhất được sử dụng để phân loại hoặc dự đoán.

- **weights:** Cách tính trọng số cho các hàng xóm. Nếu chọn ‘uniform’, tất cả các điểm lân cận đều có ảnh hưởng như nhau; nếu chọn ‘distance’, các điểm gần hơn sẽ có ảnh hưởng lớn hơn đến kết quả cuối cùng.

- **metric:** Hàm đo khoảng cách giữa các điểm, thường dùng nhất là khoảng cách Euclid (Euclidean), Manhattan hoặc Cosine.

Một ưu điểm lớn của KNN là không đòi hỏi giả định về phân phối dữ liệu, nên có thể áp dụng linh hoạt với nhiều loại dữ liệu khác nhau. Tuy nhiên, KNN cũng có một số hạn chế: khi dữ liệu có số lượng lớn hoặc nhiều nhiễu, quá trình dự đoán sẽ trở nên chậm và kém hiệu quả, vì thuật toán cần tính toán khoảng cách đến mọi điểm trong tập dữ liệu mỗi khi dự đoán. Ngoài ra, chuẩn hóa dữ liệu là một bước rất quan trọng khi áp dụng KNN, bởi thuật toán này hoàn toàn phụ thuộc vào khoảng cách giữa các điểm dữ liệu trong không gian đặc trưng.

### 3.1.1. K-Nearest Neighbors Model with Extracted Features (knn\_ex)

Ở phương pháp này, mô hình KNN được huấn luyện trên các đặc trưng khuôn mặt đã được trích xuất thông qua mạng học sâu InceptionResnetV1 (pre-trained trên tập dữ liệu VGGFace2). Việc sử dụng các vector đặc trưng trích xuất từ mạng deep learning giúp giảm đáng kể số chiều dữ liệu, loại bỏ các yếu tố nhiễu và tập trung vào những thông tin quan trọng nhất cho nhận diện khuôn mặt.

Các vector đặc trưng này, dù có kích thước nhỏ hơn rất nhiều so với ảnh gốc, nhưng lại chứa các đặc tính phân biệt cá nhân rõ rệt. Nhờ vậy, mô hình KNN có thể phân loại khuôn mặt một cách hiệu quả và nhanh chóng hơn so với việc sử dụng toàn bộ ảnh gốc hoặc ảnh đã được làm phẳng.

Để tối ưu hóa hiệu suất, các siêu tham số của mô hình KNN được lựa chọn tự động bằng phương pháp GridSearchCV, với các thông số như số lượng hàng xóm gần nhất (n\_neighbors), trọng số (weights) và loại khoảng cách (metric).

- **Tham số tối ưu:**

+ **n\_neighbors:** 3

+ **weights:** 'uniform'

+ **metric:** 'euclidean'

Accuracy: 100.00%

Classification Report:

	precision	recall	f1-score	support
Thang	1.00	1.00	1.00	24
Su	1.00	1.00	1.00	16
Nhung	1.00	1.00	1.00	35
Tuyen	1.00	1.00	1.00	25
Vu	1.00	1.00	1.00	41
Dat	1.00	1.00	1.00	27
Huy	1.00	1.00	1.00	30
accuracy			1.00	198
macro avg	1.00	1.00	1.00	198
weighted avg	1.00	1.00	1.00	198

Mô hình đạt **độ chính xác (Accuracy)** trên tập kiểm tra là **100%**.

Báo cáo phân loại chi tiết cho thấy cả bảy lớp đều đạt giá trị **precision**, **recall**, và **f1-score**: 1.00.

**Giá trị trung bình cross-validation (Mean cross-validation score): 0.9987** => tính ổn định và khả năng tổng quát tốt của mô hình.

Có thể thấy, **K-Nearest Neighbors Model with Extracted Features** với các siêu tham số tối ưu đã đạt hiệu suất tốt nhất:

- **Precision**: 1.0

- **Recall**: 1.0

- **F1 Score**: 1.0

- **Cross-validation scores**: [0.99367089 1. 1. 1. 1. ]

- **Mean cross-validation score**: 0.9987341772151899

=> Từ kết quả này có thể thấy được **K-Nearest Neighbors Model with Extracted Features** là phù hợp và hiệu quả cao đối với bài toán nhận diện khuôn mặt.

### 3.1.2. K-Nearest Neighbors Model with Flattened Image Features (knn\_flat)

Tập dữ liệu sử dụng cho mô hình có kích thước (988, 49152), trong đó mỗi mẫu được biểu diễn bởi một vector làm phẳng từ ảnh RGB kích thước 128×128×3.

Quá trình tối ưu hóa cho các siêu tham số cho KNN được thực hiện bằng GridSearchCV, với các lựa chọn cho số lượng hàng xóm (n\_neighbors), trọng số (weights) và loại khoảng cách (metric). Kết quả tối ưu như sau:

**- Tham số tối ưu:**

+ **n\_neighbors:** 3

+ **weights:** 'distance'

+ **metric:** 'manhattan'

Accuracy: 99.49%

Classification Report:

	precision	recall	f1-score	support
Thang	1.00	1.00	1.00	24
Su	1.00	0.94	0.97	16
Nhung	1.00	1.00	1.00	35
Tuyen	1.00	1.00	1.00	25
Vu	1.00	1.00	1.00	41
Dat	0.96	1.00	0.98	27
Huy	1.00	1.00	1.00	30
accuracy			0.99	198
macro avg	0.99	0.99	0.99	198
weighted avg	1.00	0.99	0.99	198

Mô hình đạt **độ chính xác (Accuracy) trên tập kiểm tra: 99.49%**.

Báo cáo phân loại chi tiết cho thấy hầu hết các lớp đều đạt giá trị **precision**, **recall** và **f1-score** rất cao. Chỉ có một số trường hợp nhầm lẫn nhẹ ở lớp "Su" và "Dat".

**Giá trị trung bình cross-validation (Mean cross-validation score): 0.9949** => mô hình có độ ổn định cao.

Có thể thấy, **K-Nearest Neighbors Model with Flattened Image Features** với các siêu tham số tối ưu đã đạt được hiệu suất rất cao trên tập dữ liệu kiểm tra, thể hiện qua các chỉ số sau:

- **Precision:** 0.9951298701298701

- **Recall:** 0.9949494949494949

- **F1-score:** 0.9949139488728932

- **Cross-validation scores:** [0.9873417 0.98734177 1. 1. 1. ]

- **Mean cross-validation score:** 0.9949367088607595

=> Từ kết quả này có thể thấy được **K-Nearest Neighbors Model with Flattened Image Features** cũng đạt hiệu suất rất cao, đủ đáp ứng cho bài toán nhận diện khuôn mặt trong các điều kiện đa dạng về ánh sáng và tư thế.

### 3.2. Random forest (RF)

**Random Forest (RF)** là một thuật toán học máy mạnh mẽ được sử dụng cho cả các bài toán phân loại và hồi quy. Mục tiêu chính của Random Forest là đưa ra dự đoán chính xác bằng cách kết hợp kết quả của nhiều cây quyết định (decision trees) riêng lẻ, nhờ đó tăng cường độ ổn định và giảm nguy cơ overfitting.

Ý tưởng cốt lõi của Random Forest là: **mỗi cây quyết định được xây dựng dựa trên một tập con ngẫu nhiên của dữ liệu gốc** (thường gọi là bootstrap sampling) và tại mỗi nút phân chia, chỉ một tập con ngẫu nhiên các đặc trưng được chọn để xem xét việc tách nhánh. Nhờ sự ngẫu nhiên này, mỗi cây trong rừng là duy nhất và góp phần tạo nên sự đa dạng cho mô hình.

Khi dự đoán, Random Forest tổng hợp kết quả của tất cả các cây quyết định. Đối với bài toán phân loại, mô hình sẽ lấy nhãn xuất hiện nhiều nhất (đa số phiếu) trong các cây. Đối với bài toán hồi quy, mô hình sẽ lấy giá trị trung bình của tất cả các cây.

Các siêu tham số quan trọng trong Random Forest gồm:

- **n\_estimators:** Số lượng cây quyết định trong rừng. Số lượng cây lớn thường giúp tăng độ chính xác nhưng cũng làm tăng thời gian huấn luyện.
- **max\_depth:** Độ sâu tối đa của mỗi cây. Độ sâu quá lớn có thể dẫn đến overfitting, trong khi quá nhỏ có thể làm giảm độ chính xác.
- **max\_features:** Số lượng đặc trưng được xem xét khi tìm cách phân chia tốt nhất tại mỗi nút.
- **criterion:** Tiêu chí đánh giá độ tốt của một phép chia, phổ biến là "gini" và "entropy" cho phân loại.

Random Forest không yêu cầu chuẩn hóa dữ liệu và thường không nhạy cảm với dữ liệu nhiễu hoặc thiếu, đồng thời có khả năng xử lý tốt các dữ liệu có nhiều đặc trưng. Tuy nhiên, mô hình có thể tiêu tốn nhiều bộ nhớ và thời gian tính toán khi số lượng cây hoặc số đặc trưng quá lớn.



### 3.2.1. Random Forest Model with Extracted Features (rf\_ex)

Ở phương pháp này, mô hình Random Forest được huấn luyện với đầu vào là các đặc trưng khuôn mặt được trích xuất từ mạng học sâu (deep learning), giúp giảm chiều dữ liệu và tập trung vào các thông tin quan trọng nhất cho nhận diện khuôn mặt.

Quá trình tối ưu hóa siêu tham số được thực hiện bằng GridSearchCV, với các tham số được lựa chọn tối ưu như sau:

- **n\_estimators**: 200
- **max\_depth**: None
- **min\_samples\_split**: 2
- **min\_samples\_leaf**: 4
- **bootstrap**: True

Accuracy: 100.00%

Classification Report:

	precision	recall	f1-score	support
Thang	1.00	1.00	1.00	24
Su	1.00	1.00	1.00	16
Nhung	1.00	1.00	1.00	35
Tuyen	1.00	1.00	1.00	25
Vu	1.00	1.00	1.00	41
Dat	1.00	1.00	1.00	27
Huy	1.00	1.00	1.00	30
Dat	1.00	1.00	1.00	27
Huy	1.00	1.00	1.00	30
accuracy			1.00	198
macro avg	1.00	1.00	1.00	198
weighted avg	1.00	1.00	1.00	198

Mô hình đạt **độ chính xác (Accuracy) 100%** trên tập kiểm tra.

Báo cáo phân loại cho thấy với các chỉ số **precision**, **recall** và **f1-score** đều đạt giá trị tuyệt đối cho tất cả các lớp: 1.0.

**Giá trị trung bình cross-validation (Mean cross-validation score): 0.9974** => sự ổn định và khả năng tổng quát hóa rất tốt của mô hình.

Có thể thấy, **Random Forest Model with Extracted Features** với các siêu tham số tối ưu đã đạt được hiệu suất rất cao trên tập dữ liệu kiểm tra, thể hiện qua các chỉ số sau:

- **Precision:** 1.00
- **Recall:** 1.00
- **F1-score:** 1.00
- **Cross-validation scores:** [0.99367089 1. 1. 0.99367089 1. ]
- **Mean cross-validation score:** 0.9974683544303797

=> Từ kết quả này có thể thấy được **Random Forest Model with Extracted Features** là lựa chọn rất hiệu quả cho bài toán nhận diện khuôn mặt trong đề tài.

### 3.2.2. Random Forest Model with Flattened Image Features (rf\_flat)

Trong phương pháp này, mô hình Random Forest được huấn luyện với đầu vào là các đặc trưng ảnh gốc đã được làm phẳng thành vector một chiều kích thước 49152. Bộ dữ liệu sử dụng có kích thước (988, 49152), tương ứng với 988 mẫu.

Quá trình tối ưu hóa siêu tham số được thực hiện bằng GridSearchCV, với các tham số tối ưu tìm được như sau:

- **n\_estimators:** 50
- **max\_depth:** None
- **min\_samples\_split:** 5
- **min\_samples\_leaf:** 2
- **bootstrap:** True

```

Accuracy: 96.46%
Classification Report:
              precision    recall  f1-score   support

   Thang         1.00        1.00        1.00        24
     Su         1.00        1.00        1.00        16
   Nhung         0.92        0.97        0.94        35
   Tuyen         0.96        0.88        0.92        25
     Vu         1.00        0.93        0.96        41
     Dat         0.90        1.00        0.95        27
     Huy         1.00        1.00        1.00        30

 accuracy         0.96        198
 macro avg         0.97        0.97        0.97        198
weighted avg         0.97        0.96        0.96        198

```

Mô hình đạt **độ chính xác (accuracy): 96.46%** trên tập kiểm tra.

Báo cáo phân loại cho thấy mô hình nhận diện rất tốt đối với các lớp như “Thang”, “Su” và “Huy”, khi **precision, recall** và **f1-score** đều đạt mức tuyệt đối (1.00). Đối với các lớp còn lại như “Nhung”, “Tuyen”, “Vu” và “Dat”, các chỉ số **precision, recall** hoặc **f1-score** thấp hơn một chút, nhưng nhìn chung vẫn duy trì ở mức cao, đảm bảo hiệu quả và độ ổn định tổng thể của mô hình.

Giá trị trung bình **cross-validation (Mean cross-validation score): 0.9924** => độ ổn định và khả năng tổng quát hóa tốt của mô hình.

Có thể thấy, mô hình RF với các siêu tham số tối ưu đã đạt được hiệu suất rất cao trên tập dữ liệu kiểm tra, thể hiện qua các chỉ số sau:

- **Precision:** 0.9665414426283992
- **Recall:** 0.9646464646464646
- **F1-score:** 0.964617209925893
- **Cross-validation scores:** [1. 1. 0.99367089 0.98734177 0.98101266]
- **Mean cross-validation score:** 0.9924050632911392

=> Từ kết quả này có thể thấy được **Random Forest Model with Flattened Image Features** đạt hiệu quả và duy trì được sự ổn định trên các lần đánh giá khác nhau. Tuy nhiên, hiệu quả của mô hình này vẫn thấp hơn một chút so với mô hình sử dụng đặc trưng trích xuất từ deep learning.

### 3.3. Support Vector Machine (SVM)

**Support Vector Machine (SVM)** là một thuật toán học máy phổ biến, mạnh mẽ và được sử dụng rộng rãi cho cả các bài toán phân loại và hồi quy. Mục tiêu chính của SVM là tìm ra siêu phẳng (hyperplane) tối ưu nhất để phân tách các lớp dữ liệu trong không gian đặc trưng, từ đó giúp mô hình phân biệt các nhóm dữ liệu một cách hiệu quả.

Ý tưởng cốt lõi của SVM là: thuật toán sẽ xác định siêu phẳng có **khoảng cách biên (margin)** lớn nhất tới các điểm dữ liệu gần nhất của mỗi lớp (gọi là support vectors). Việc tối ưu hóa khoảng cách biên này giúp tăng khả năng tổng quát hóa của mô hình, giảm nguy cơ overfitting và nâng cao độ chính xác khi dự đoán dữ liệu mới.

Một điểm nổi bật của SVM là khả năng xử lý dữ liệu không phân tách tuyến tính bằng cách sử dụng **hàm kernel** (kernel trick). Các hàm kernel như tuyến tính (linear), hàm Gaussian (RBF), hoặc đa thức (polynomial) giúp ánh xạ dữ liệu gốc sang không gian chiều cao hơn, nơi các lớp có thể phân tách rõ ràng hơn.

Các siêu tham số quan trọng trong SVM bao gồm:

- **C**: Hệ số phạt (regularization parameter), điều chỉnh sự đánh đổi giữa việc tối đa hóa khoảng cách biên và giảm số điểm bị phân loại sai. Giá trị C lớn sẽ ưu tiên phân loại đúng mọi điểm nhưng dễ dẫn đến overfitting, còn giá trị nhỏ sẽ cho phép mô hình bỏ qua một số điểm nhiễu để có biên rộng hơn.

- **kernel**: Loại hàm kernel được sử dụng để ánh xạ dữ liệu vào không gian mới. Các lựa chọn phổ biến là 'linear', 'rbf', 'poly', v.v.

- **gamma**: Tham số xác định ảnh hưởng của một điểm mẫu đối với đường phân tách trong các kernel phi tuyến như RBF và polynomial.

- **degree**: Bậc của hàm đa thức khi sử dụng kernel 'poly'.

SVM thường hoạt động rất tốt với dữ liệu có số chiều lớn và ít bị ảnh hưởng bởi vấn đề overfitting khi dữ liệu được chọn lọc và chuẩn hóa tốt. Tuy nhiên, SVM có thể tốn nhiều thời gian tính toán với tập dữ liệu lớn và đòi hỏi phải lựa chọn, tinh chỉnh các siêu tham số phù hợp để đạt hiệu quả tối ưu.

#### 3.3.1. Support Vector Machine Model with Extracted Features (svm\_ex)

Trong phương pháp này, mô hình SVM được huấn luyện với đầu vào là các đặc trưng đã được trích xuất từ mạng học sâu, giúp tăng khả năng phân biệt và nhận diện khuôn mặt. Việc tối ưu hóa siêu tham số được thực hiện bằng GridSearchCV, với các tham số tối ưu tìm được như sau:

- **C**: 1

- **kernel**: 'rbf'

- **gamma:** 'scale'

Accuracy: 100.00%

Classification Report:

	precision	recall	f1-score	support
Thang	1.00	1.00	1.00	24
Su	1.00	1.00	1.00	16
Nhung	1.00	1.00	1.00	35
Tuyen	1.00	1.00	1.00	25
Vu	1.00	1.00	1.00	41
Dat	1.00	1.00	1.00	27
Huy	1.00	1.00	1.00	30
accuracy			1.00	198
macro avg	1.00	1.00	1.00	198
weighted avg	1.00	1.00	1.00	198

Mô hình đạt **độ chính xác (accuracy): 100%** trên tập kiểm tra.

Báo cáo phân loại cho thấy tất cả các lớp đều đạt các giá trị **precision**, **recall** và **f1-score** tuyệt đối: 1.0

**Giá trị trung bình cross-validation (Mean cross-validation score): 1.0** => mô hình rất ổn định và tổng quát tốt.

Có thể thấy, **Support Vector Machine Model with Extracted Features** với các siêu tham số tối ưu đã đạt được hiệu suất rất cao trên tập dữ liệu kiểm tra, thể hiện qua các chỉ số sau:

- **Precision:** 1.00

- **Recall:** 1.00

- **F1-score:** 1.00

- **Cross-validation scores:** [1. 1. 1. 1. 1.]

- **Mean cross-validation score:** 1.0

=> Kết quả này khẳng định **Support Vector Machine Model with Extracted Features** là một giải pháp cực kỳ hiệu quả và đáng tin cậy cho bài toán nhận diện khuôn mặt trong đề tài này.

### 3.3.2. Support Vector Machine Model with Flattened Image Features (svm\_flat)

Trong phương pháp này, mô hình SVM được huấn luyện với đầu vào là các vector đặc trưng thu được bằng cách làm phẳng (flatten) ảnh gốc thành các vector một chiều kích thước 49152. Tập dữ liệu sử dụng có kích thước (988, 49152), tương ứng với 988 mẫu.

Quá trình tối ưu hóa siêu tham số được thực hiện bằng GridSearchCV, với các tham số tối ưu thu được như sau:

- **C:** 0.1

- **kernel:** 'linear'

- **gamma:** 'scale'

Accuracy: 100.00%

Classification Report:

	precision	recall	f1-score	support
Thang	1.00	1.00	1.00	24
Su	1.00	1.00	1.00	16
Nhung	1.00	1.00	1.00	35
Tuyen	1.00	1.00	1.00	25
Vu	1.00	1.00	1.00	41
Dat	1.00	1.00	1.00	27
Huy	1.00	1.00	1.00	30
accuracy			1.00	198
macro avg	1.00	1.00	1.00	198
weighted avg	1.00	1.00	1.00	198

Mô hình đạt **độ chính xác (accuracy) 100%** trên tập kiểm tra.

Báo cáo phân loại cho thấy tất cả các lớp đều đạt các giá trị **precision**, **recall** và **f1-score** tuyệt đối: 1.0

**Giá trị trung bình cross-validation (Mean cross-validation score): 0.9962** => sự ổn định và khả năng tổng quát hóa rất tốt.

Có thể thấy, mô hình SVM với các siêu tham số tối ưu đã đạt được hiệu suất rất cao trên tập dữ liệu kiểm tra, thể hiện qua các chỉ số sau:

- **Precision:** 1.00

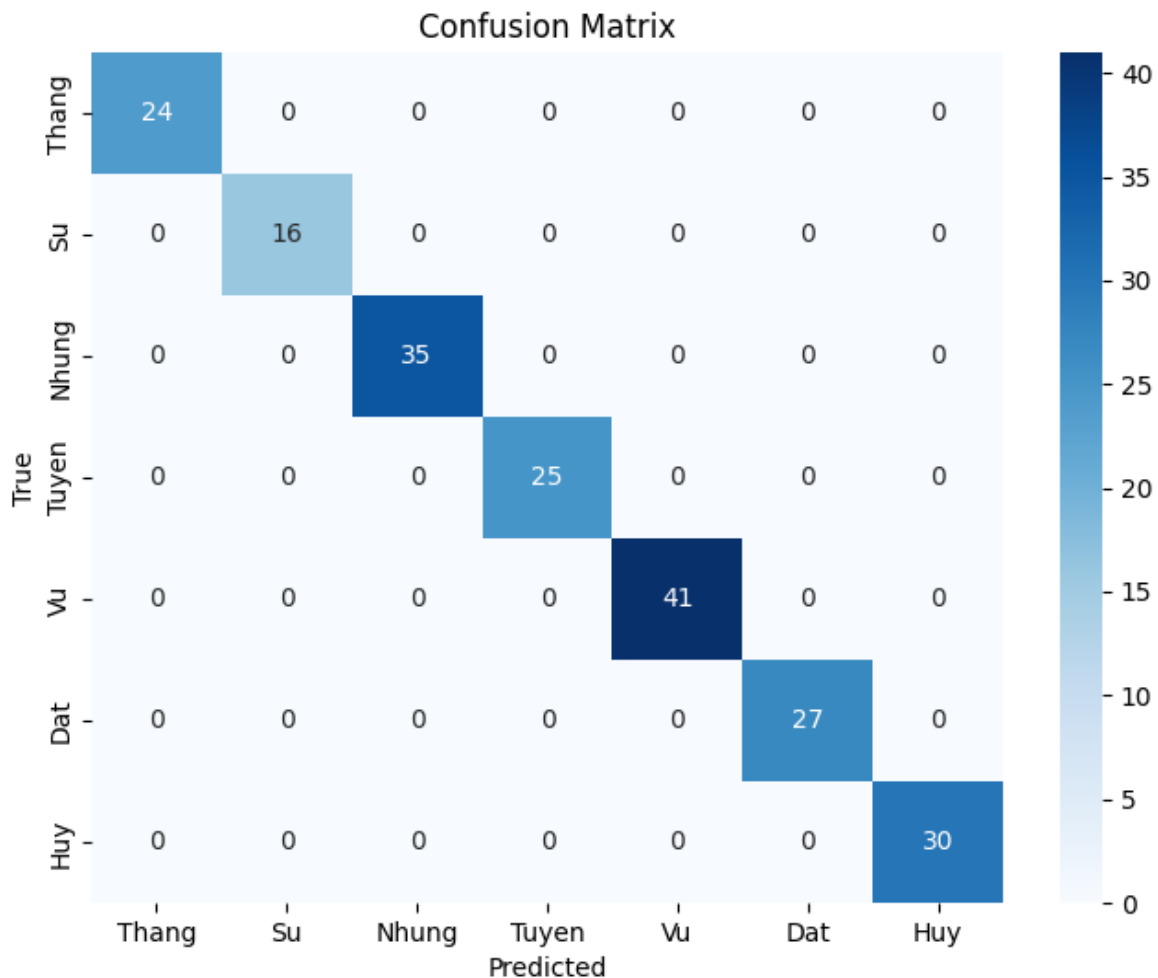
- **Recall:** 1.00
- **F1-score:** 1.00
- **Cross-validation scores:** [1. 0.99367089 0.99367089 1. 0.99367089]
- **Mean cross-validation score:** 0.9962025316455696

=> Kết quả này cho thấy **Support Vector Machine Model with Flattened Image Features** cũng đạt hiệu suất nhận diện rất cao và ổn định trên tập dữ liệu.

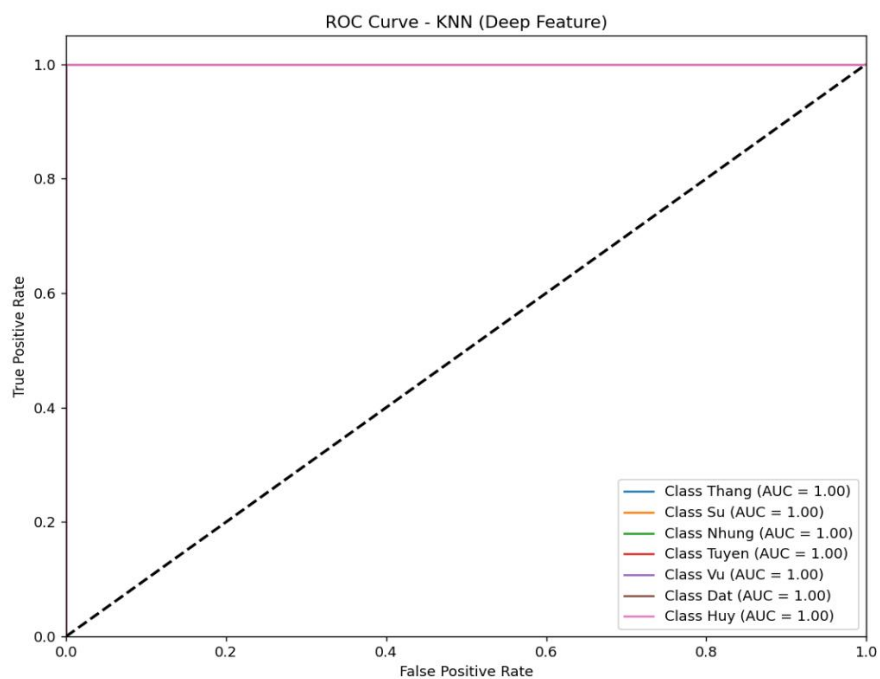
## Phần 4. Kết quả

### 4.1. K-Nearest Neighbors (KNN)

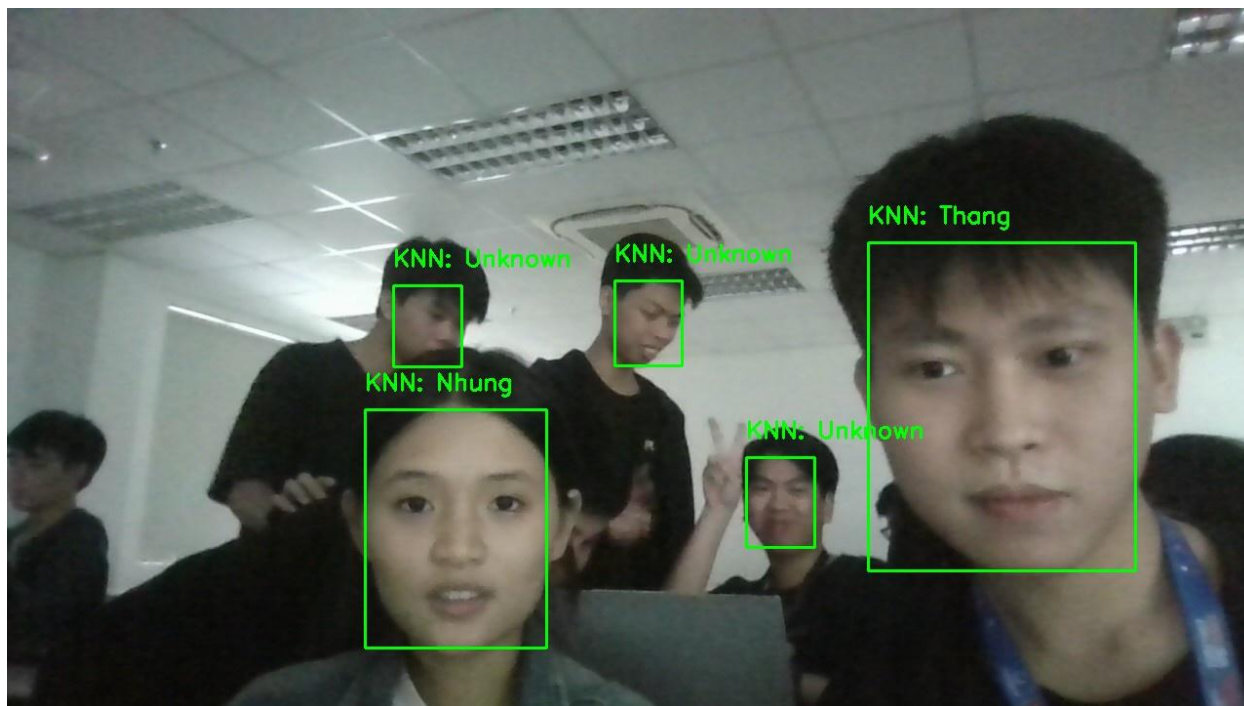
#### 4.1.1. K-Nearest Neighbors Model with Extracted Features (knn\_ex)



*Confusion matrix của K-Nearest Neighbors Model with Extracted Features*

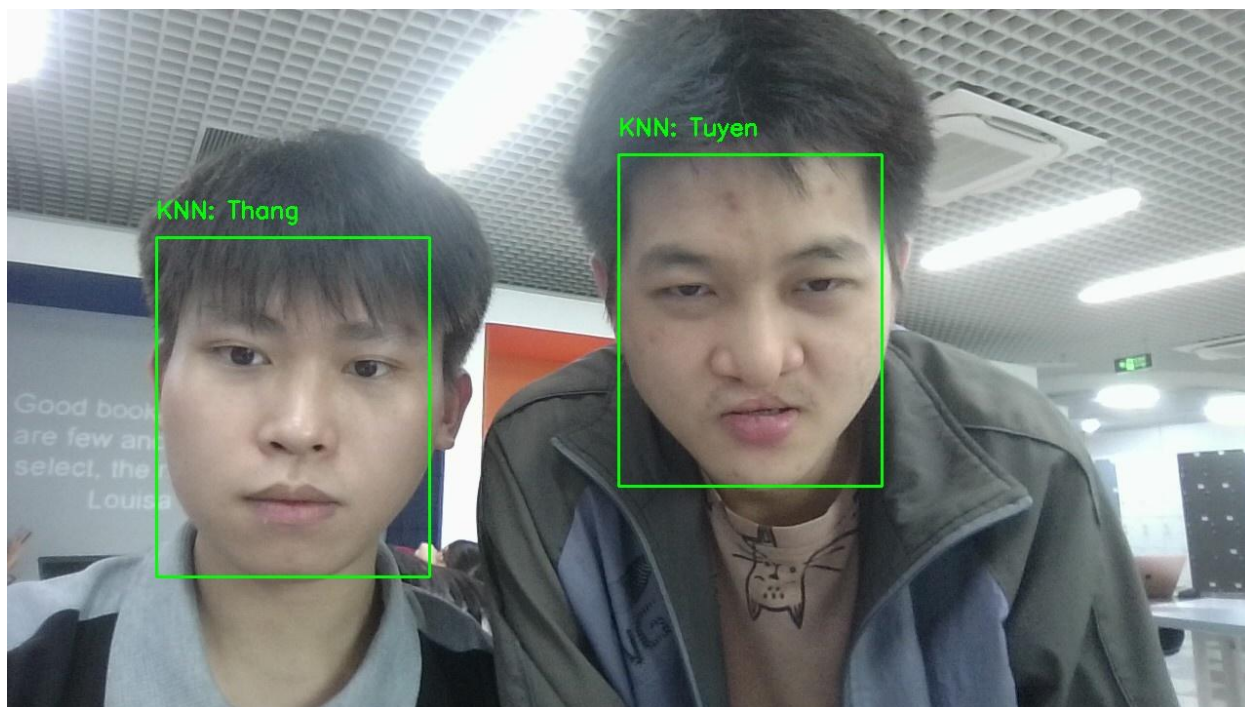


*ROC curve của K-Nearest Neighbors Model with Extracted Features*

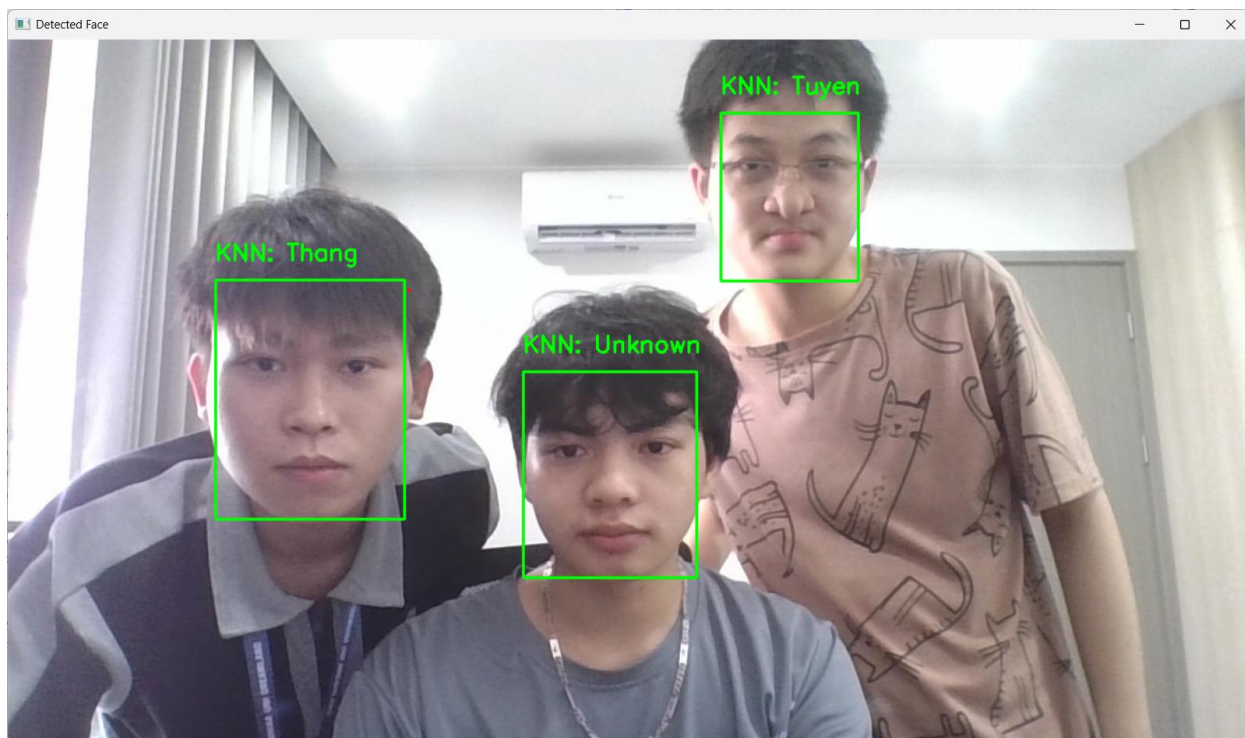


*Ảnh 1.1: Kiểm tra với K-Nearest Neighbors Model with Extracted Features*





*Ảnh 1.2: Kiểm tra với K-Nearest Neighbors Model with Extracted Features*



*Ảnh 1.3: Kiểm tra với K-Nearest Neighbors Model with Extracted Features*

*Kết quả nhận diện khuôn mặt với mô hình K-Nearest Neighbors Model with Extracted Features. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình.*

**=> Kết luận:**

Kết quả thí nghiệm nhận diện khuôn mặt sử dụng **K-Nearest Neighbors Model with Extracted Features** cho thấy hiệu quả vượt trội trên tập dữ liệu hiện tại. Cụ thể:

- **Confusion matrix hoàn hảo:** Trên tập kiểm tra, mô hình phân loại chính xác 100% tất cả các lớp, không xuất hiện nhầm lẫn giữa các đối tượng.

- **Chỉ số đánh giá tuyệt đối:** Tất cả các chỉ số như precision, recall và f1-score đều đạt giá trị 1.00 ở mọi lớp, cho thấy mô hình không bỏ sót hay nhầm lẫn bất kỳ trường hợp nào.

- **Đường cong ROC và AUC:** Đường cong ROC của từng lớp đều đạt  $AUC = 1.00$ , khẳng định khả năng phân biệt hoàn hảo giữa các đối tượng trong tập dữ liệu.

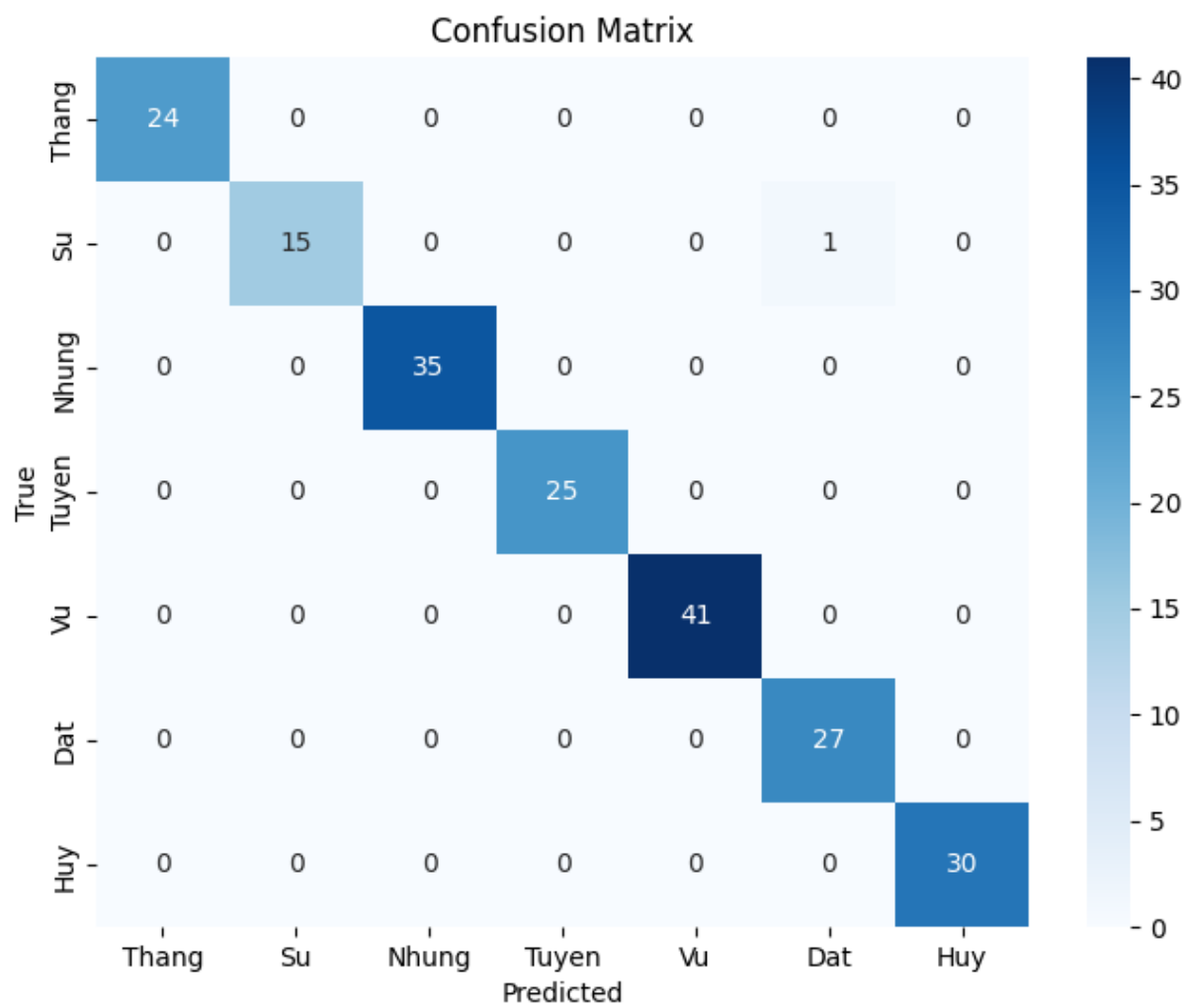
- **Thực nghiệm thực tế:**

- + Trong các bức ảnh thực nghiệm (ảnh 1, ảnh 2), đa số các khuôn mặt đều được mô hình nhận diện và gán nhãn chính xác theo từng cá nhân; các khuôn mặt không có trong tập huấn luyện đều được gán nhãn “Unknown” đúng như mong đợi.

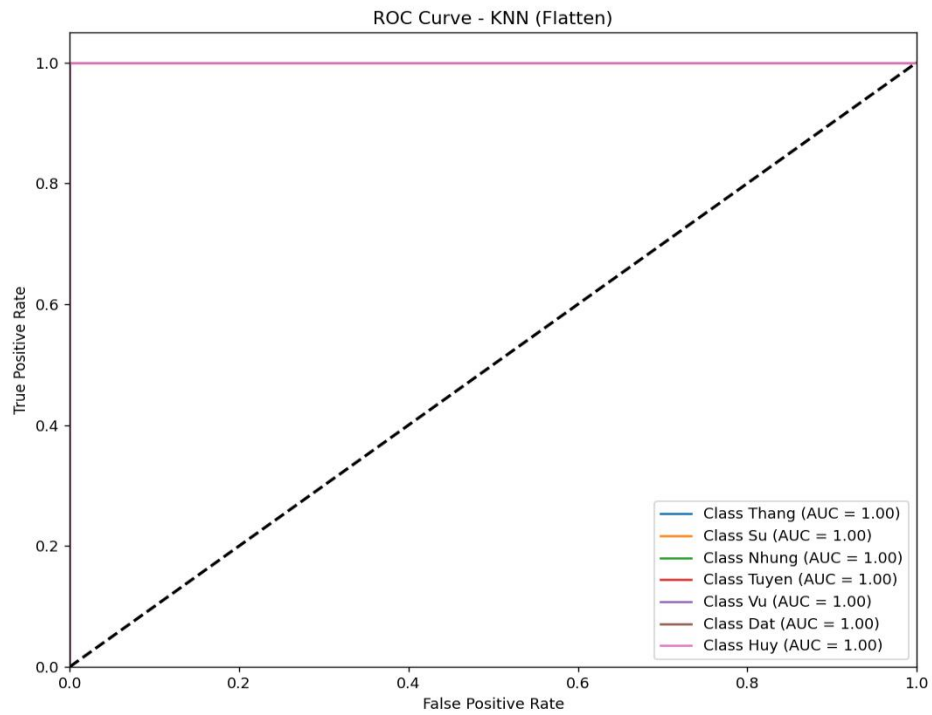
- + Tuy nhiên, ở bức ảnh thứ 3, xuất hiện trường hợp mô hình nhận diện nhầm khuôn mặt thuộc lớp “Su” đã có trong tập huấn luyện lại bị dự đoán thành "Unknown", dù điều kiện chụp ảnh hoàn toàn bình thường. Điều này cho thấy mô hình vẫn có thể gặp lỗi nhận diện trong thực tế, kể cả với ảnh chất lượng tốt hoặc bối cảnh không phức tạp.

Tổng kết lại, **K-Nearest Neighbors Model with Extracted Features** đang hoạt động cực kỳ hiệu quả trên tập dữ liệu thử nghiệm và là nền tảng tốt để triển khai thực tế. Tuy nhiên, để đảm bảo tính tổng quát và độ ổn định khi áp dụng ngoài đời thực, đặc biệt trong những trường hợp ảnh chụp bình thường vẫn có thể nhận nhầm, cần tiếp tục kiểm tra mô hình với các dữ liệu đa dạng hơn về góc nhìn, điều kiện ánh sáng và bối cảnh thực tế phức tạp hơn.

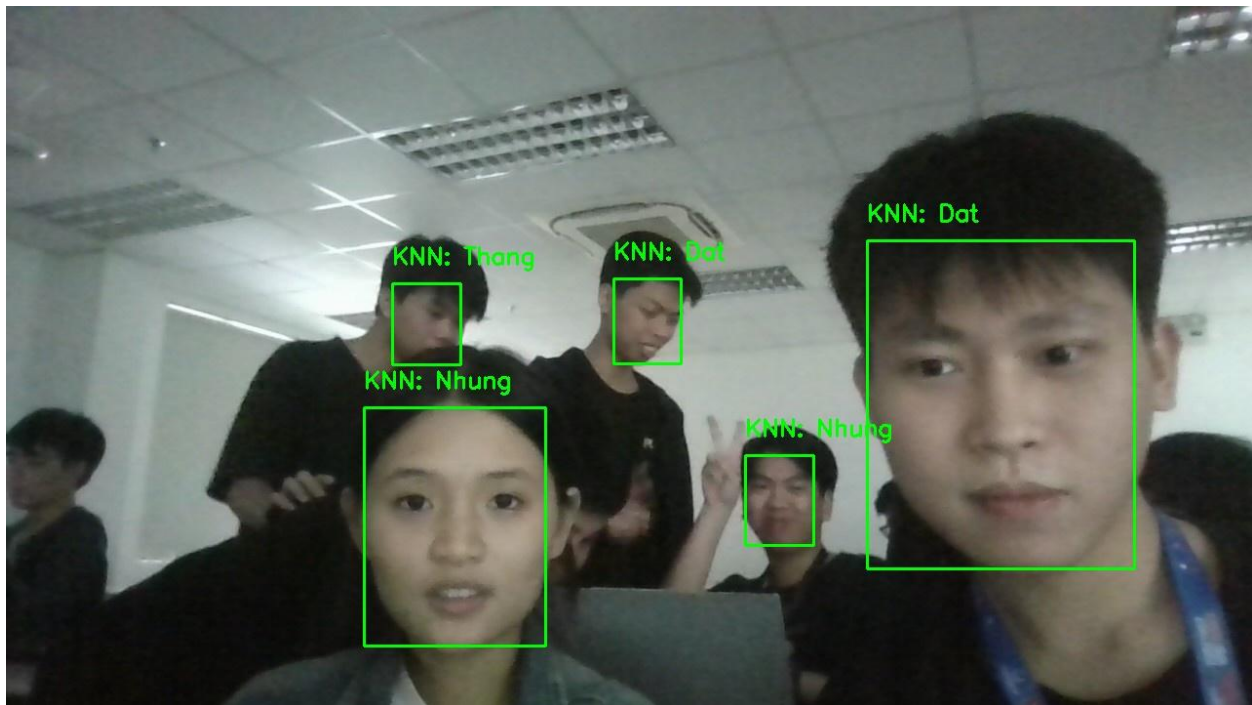
4.1.2. K-Nearest Neighbors Model with Flattened Image Features (knn\_flat)



Confusion matrix của K-Nearest Neighbors Model with Flattened Image Features

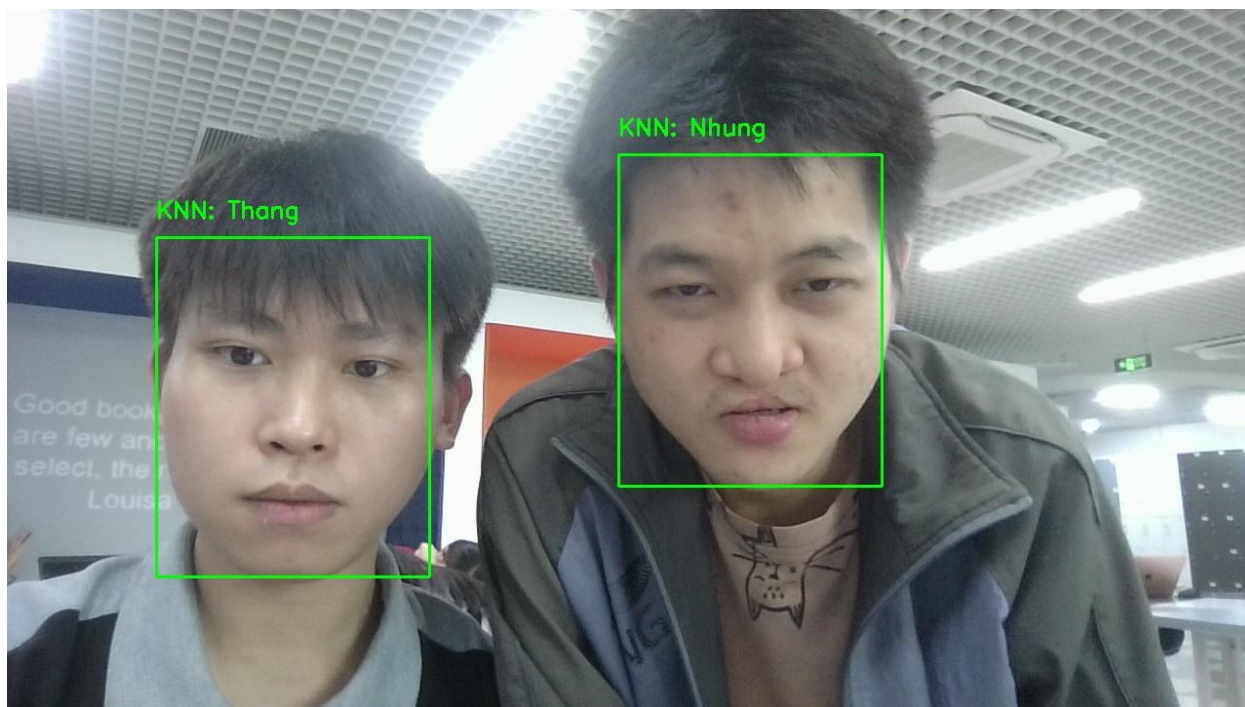


*ROC curve của K-Nearest Neighbors Model with Flattened Image Features*

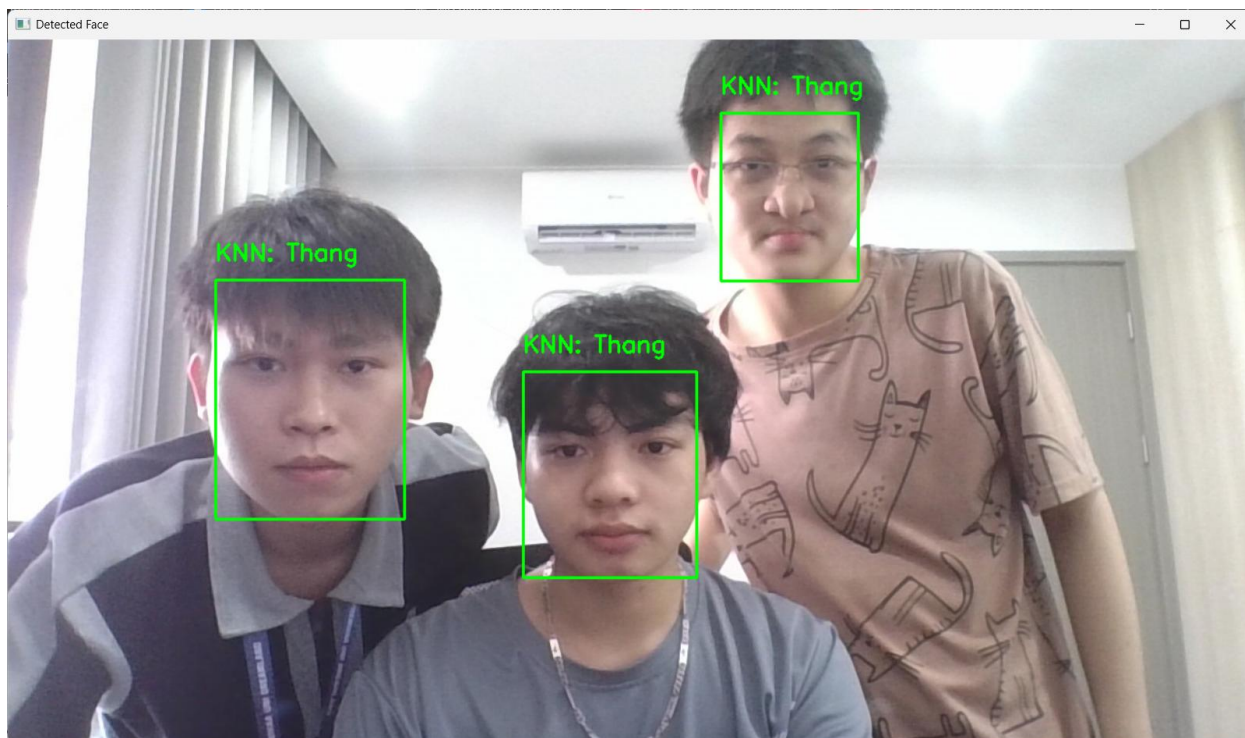


*Ảnh 1.4: Nhận diện với K-Nearest Neighbors Model with Flattened Image Features*





*Ảnh 1.5: Nhận diện với K-Nearest Neighbors Model with Flattened Image Features*



*Ảnh 1.6: Nhận diện với K-Nearest Neighbors Model with Flattened Image Features*

*Kết quả nhận diện khuôn mặt với K-Nearest Neighbors Model with Flattened Image Features. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình*

### = > **Kết luận:**

Mô hình KNN sử dụng đặc trưng ảnh làm phẳng đạt hiệu quả nhận diện khuôn mặt khá tốt trên tập dữ liệu thử nghiệm, song vẫn tồn tại một số hạn chế nhất định:

- **Ma trận nhầm lẫn (confusion matrix):** Phần lớn các lớp đều được dự đoán chính xác hoàn toàn, tuy nhiên vẫn xuất hiện sai sót – điển hình là trường hợp lớp "Su" có 1 mẫu bị nhầm với lớp "Tuyen". Điều này cho thấy, với những trường hợp các khuôn mặt có đặc trưng tương đồng hoặc chưa đủ khác biệt, mô hình có thể đưa ra dự đoán sai.

- **Đường cong ROC và AUC:** Các giá trị AUC của từng lớp đều đạt mức tối đa 1.00, cho thấy khả năng phân biệt giữa các lớp trên tập dữ liệu thử nghiệm vẫn rất tốt. Tuy nhiên, kết quả này cũng có thể bị ảnh hưởng bởi cấu trúc dữ liệu chưa đủ đa dạng hoặc dữ liệu tương đối đơn giản.

### - **Thực nghiệm nhận diện thực tế:**

+ Ở bức ảnh 4, mô hình đã nhận diện sai “Thang” thành “Dat” đồng thời cả 3 khuôn mặt đằng sau đều không có trong tập dữ liệu đều bị mô hình gán nhãn sai. Thực tế nếu đúng thì 3 khuôn mặt đằng sau đây phải là nhãn “Unknown”. Điều này xảy ra trong điều kiện ánh sáng phòng học yếu và góc chụp rộng, có nhiều khuôn mặt xuất hiện cùng lúc.

+ Ở bức ảnh 5, ở trong điều kiện ánh sáng đầy đủ nhưng mô hình đã gán nhãn “Nhưng” cho một khuôn mặt trong khi thực tế đó là “Tuyen”. Có thể thấy được sự nhầm lẫn khi các đặc trưng khuôn mặt tương đối giống nhau, kể cả trong điều kiện thuận lợi.

+ Ở bức ảnh 6, khuôn mặt đầu tiên nhận là “Thang” thì là đúng nhưng 2 khuôn mặt tiếp theo đều bị nhận nhầm là “Thang” có thể kết luận dù có ảnh được chụp ở môi trường trong nhà, ánh sáng trung bình và bố cục khá rõ ràng thì vẫn có sự nhầm lẫn nhất định.

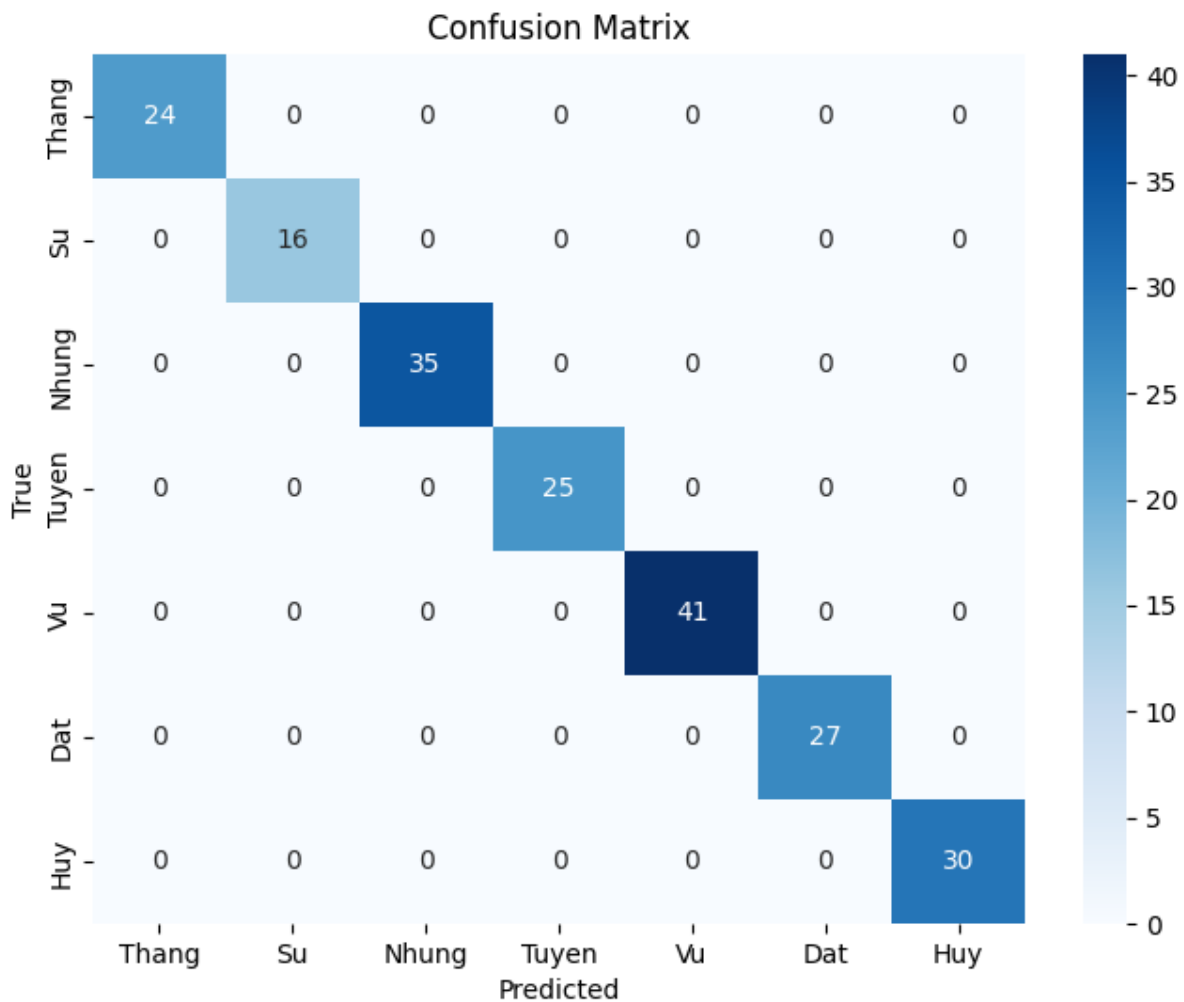
=> Những kết quả thực nghiệm cho thấy **K-Nearest Neighbors Model with Flattened Image Features** còn gặp nhiều hạn chế khi triển khai trong điều kiện thực tế. Cụ thể, mô hình dễ xảy ra nhầm lẫn khi nhận diện khuôn mặt trong các bức ảnh có nhiều người, ảnh chụp dưới ánh sáng yếu hoặc góc mặt không chính diện, khiến khả năng nhận diện giảm rõ rệt. Ngoài ra, việc nhận nhầm nhãn cũng thường xuất hiện khi đặc trưng khuôn mặt không đủ nổi bật hoặc khi bộ dữ liệu huấn luyện chưa bao quát hết các trường hợp đa dạng ngoài thực tế.

Tổng kết lại, **K-Nearest Neighbors Model with Flattened Image Features** vẫn cho kết quả khá tốt trên dữ liệu thử nghiệm chuẩn, nhưng khi áp dụng thực tế, mô hình còn xuất hiện nhiều sai sót. Điều này càng nhấn mạnh sự cần thiết phải sử dụng các phương pháp

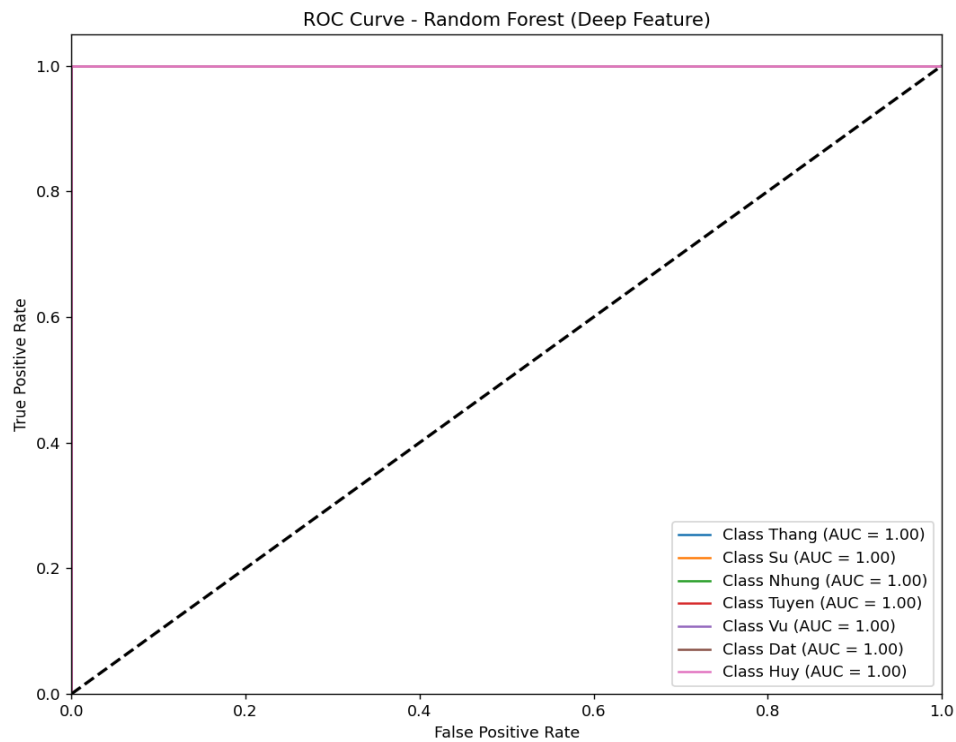
trích xuất đặc trưng sâu (deep feature) nhằm nâng cao độ chính xác và khả năng tổng quát hóa cho bài toán nhận diện khuôn mặt trong thực tiễn.

## 4.2. Random forest (RF)

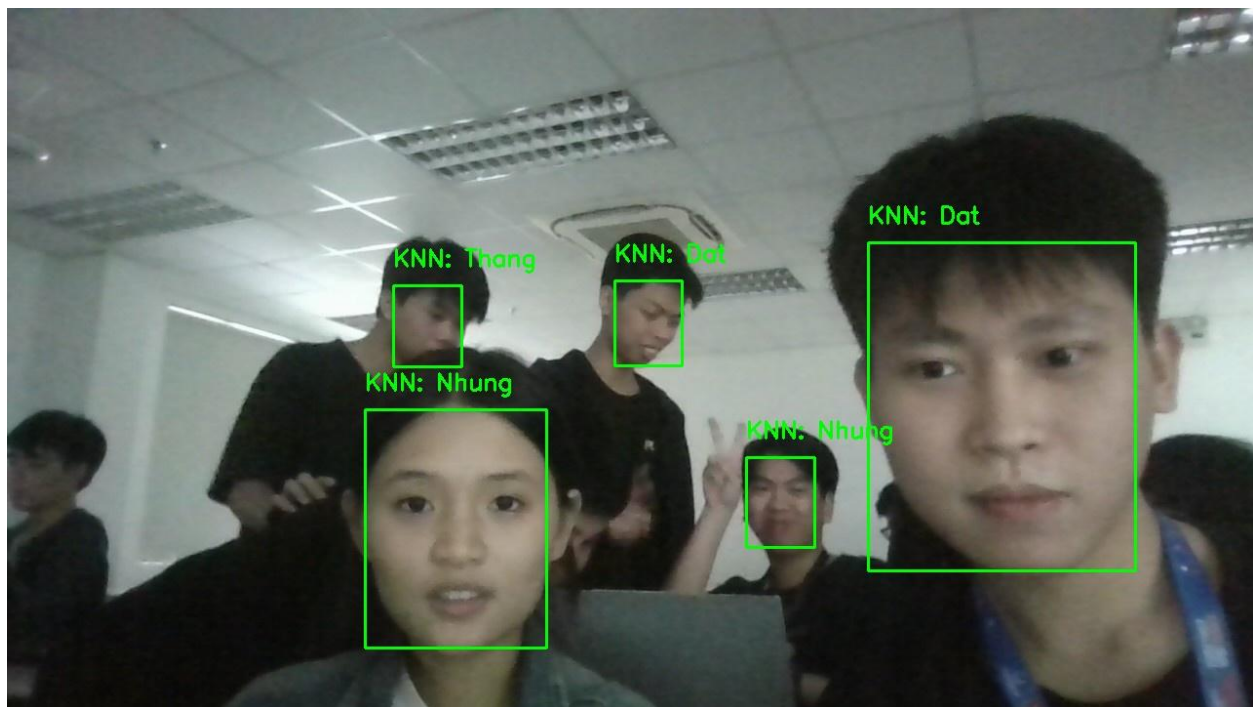
### 4.2.1. Random Forest Model with Extracted Features (rf\_ex)



*Confusion matrix của Random Forest Model with Extracted Features*

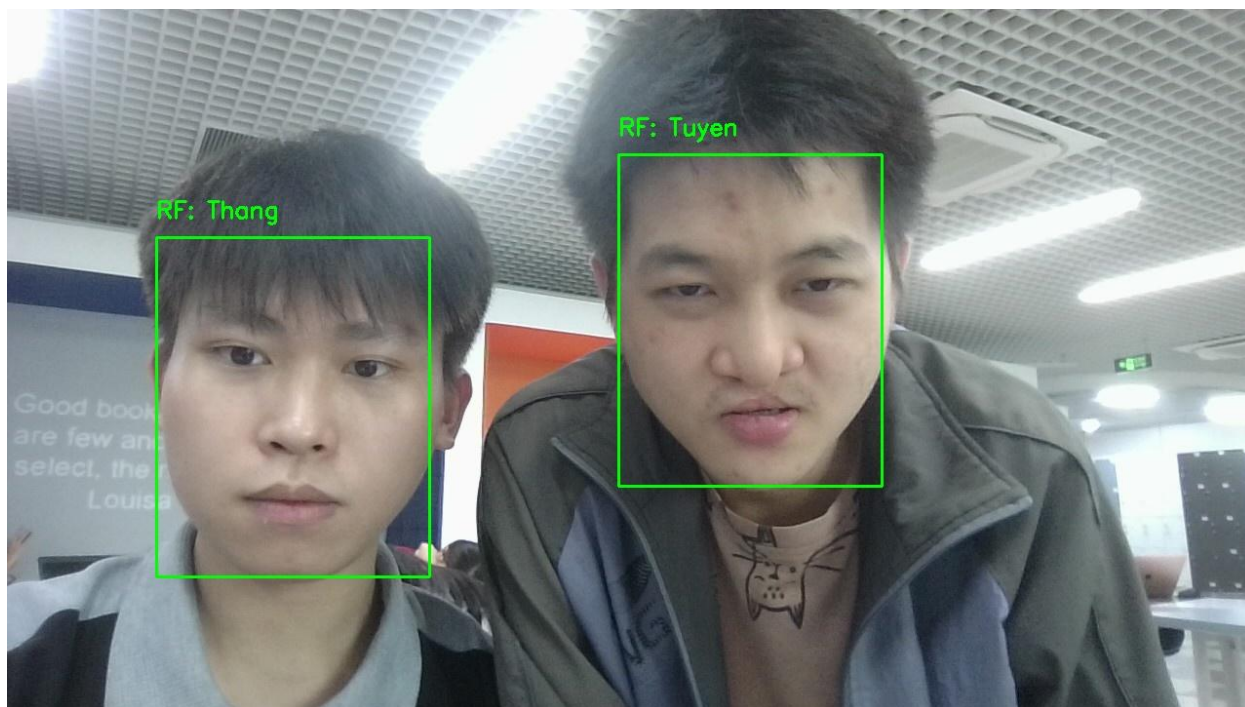


*ROC curve của Random Forest Model with Extracted Features*

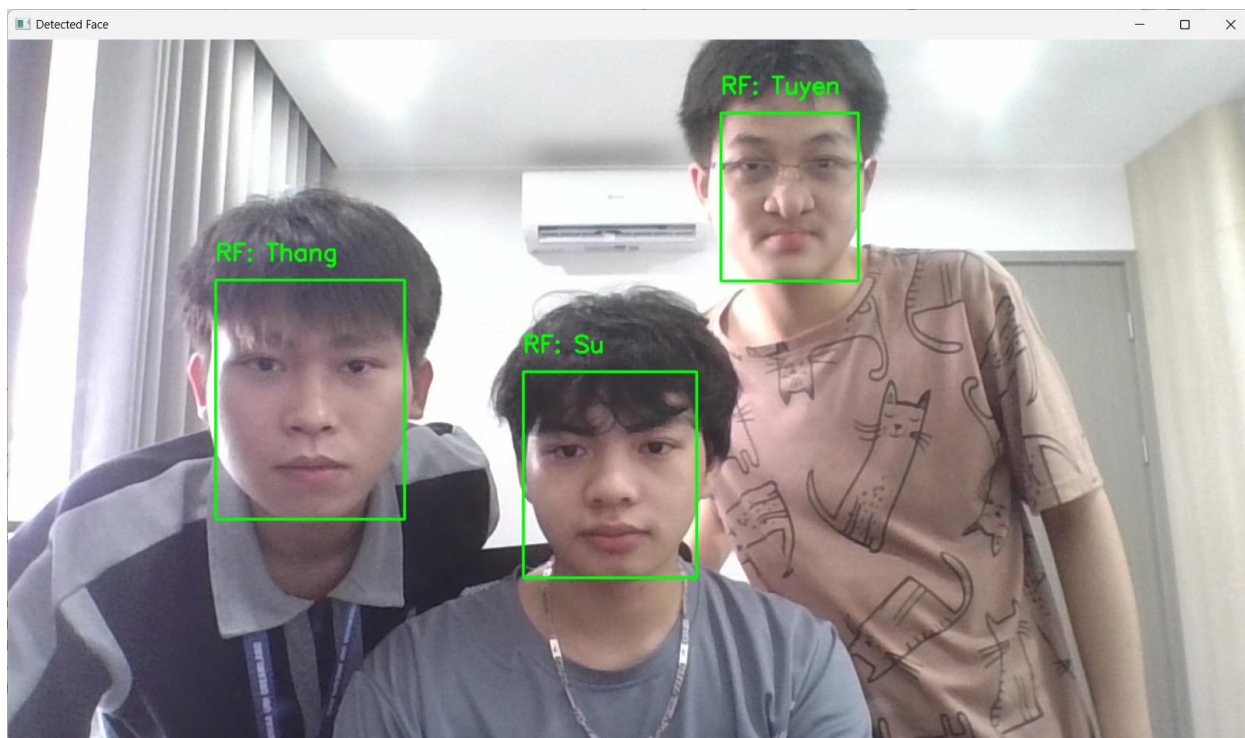


*Ảnh 2.1: Nhận diện với Random Forest Model with Extracted Features*





*Ảnh 2.2: Nhận diện với Random Forest Model with Extracted Features*



*Ảnh 2.3: Nhận diện với Random Forest Model with Extracted Features*

*Kết quả nhận diện khuôn mặt với Random Forest Model with Extracted Features. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình*

## => Kết luận:

Mô hình **Forest Model with Extracted Features** đã thể hiện hiệu quả nhận diện khuôn mặt vượt trội trên tập dữ liệu thử nghiệm:

- **Ma trận nhầm lẫn (confusion matrix):** Tất cả các lớp đều được phân loại chính xác tuyệt đối, không xảy ra bất kỳ trường hợp nhầm lẫn nào. Điều này cho thấy mô hình đã học và nhận biết rất tốt các đặc trưng riêng biệt của từng cá nhân.

- **Đường cong ROC và AUC:** Các chỉ số AUC của tất cả các lớp đều đạt mức tối đa 1.00, khẳng định khả năng phân biệt hoàn hảo giữa các đối tượng trong tập dữ liệu.

### - Nhận diện thực tế:

- + Ở bức ảnh 1: mô hình đã nhận diện sai “Thang” thành “Dat” đồng thời cả 3 khuôn mặt đằng sau đều không có trong tập dữ liệu đều bị mô hình gán nhãn sai. Thực tế nếu đúng thì 3 khuôn mặt đằng sau đây phải là nhãn “Unknown”. Điều này xảy ra trong điều kiện ánh sáng phòng học yếu và góc chụp rộng, có nhiều khuôn mặt xuất hiện cùng lúc.

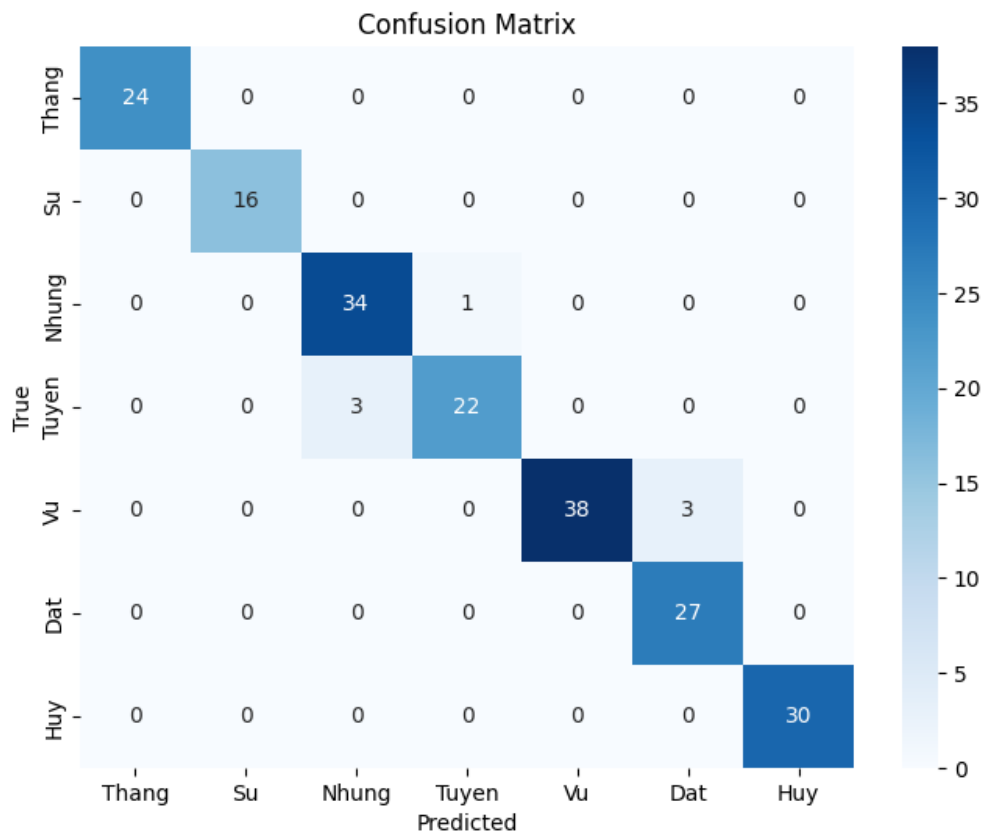
- + Ở bức ảnh 2: mô hình nhận đúng cả khuôn mặt “Thang” và “Tuyen” cho thấy hiệu quả cao khi điều kiện thuận lợi và các đặc trưng của khuôn mặt rõ ràng.

- + Ở bức ảnh 3: mô hình cũng nhận diện đúng tên từng người “Thang”, “Su”, “Tuyen” dù ánh sáng trung bình và bố cục tương đối rõ ràng.

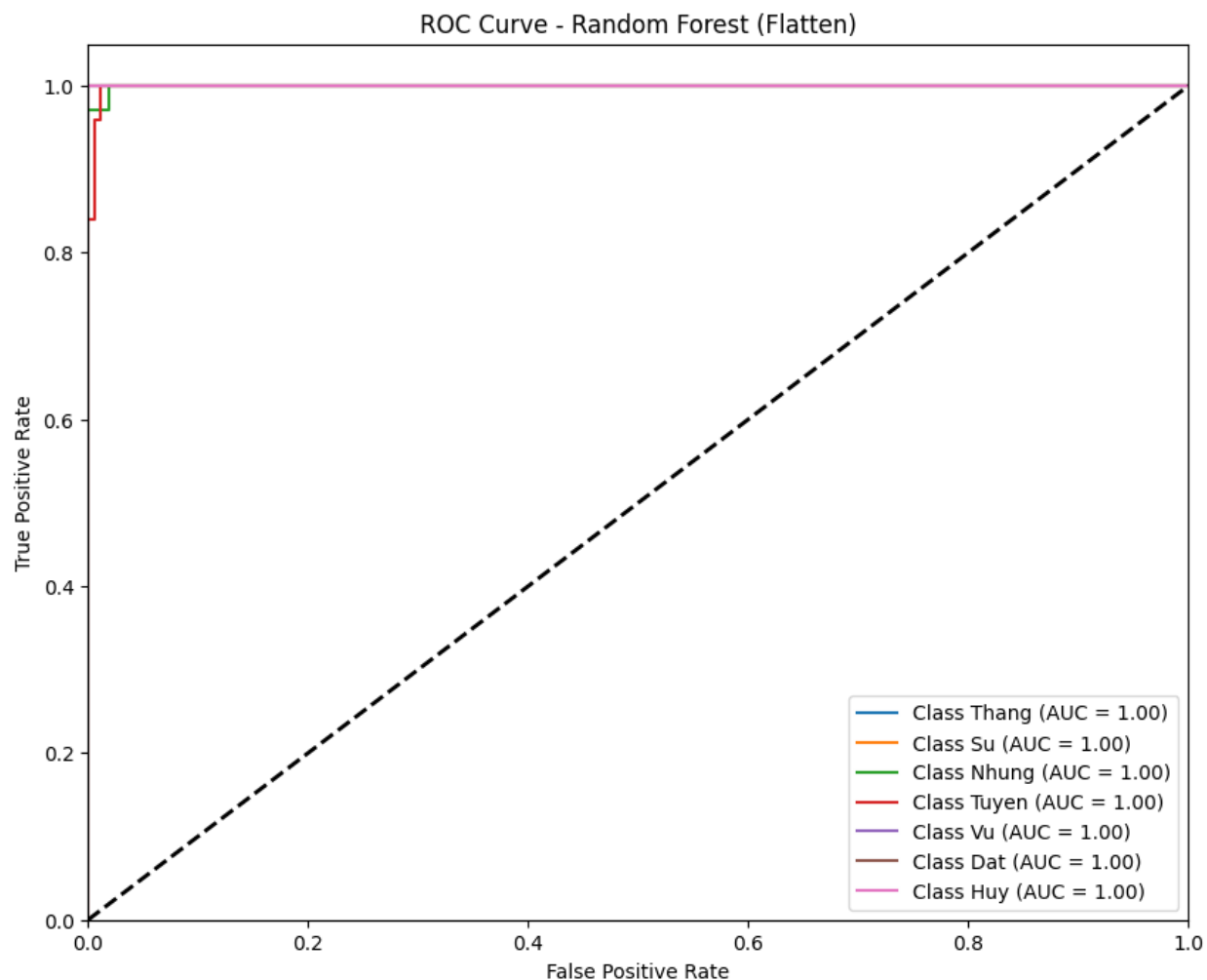
Tổng thể, **Forest Model with Extracted Features** đạt được độ chính xác rất cao trên tập dữ liệu thử nghiệm, cho thấy hiệu quả nhận diện tương đương với các mô hình KNN sử dụng đặc trưng trích xuất sâu (deep feature). Điều này khẳng định giá trị của việc kết hợp giữa các phương pháp học máy truyền thống và đặc trưng do các mô hình học sâu hiện đại cung cấp.

Tuy vậy, để đảm bảo mô hình có thể ứng dụng thực tiễn và tổng quát hóa tốt, cần tiếp tục kiểm thử trên các bộ dữ liệu đa dạng hơn – bao gồm các góc nhìn khác nhau, điều kiện ánh sáng thay đổi, nhiều biểu cảm khuôn mặt cũng như các bối cảnh môi trường phong phú. Việc này sẽ giúp phát hiện và khắc phục những hạn chế tiềm ẩn, từ đó nâng cao hơn nữa độ tin cậy của mô hình khi triển khai trong thực tế.

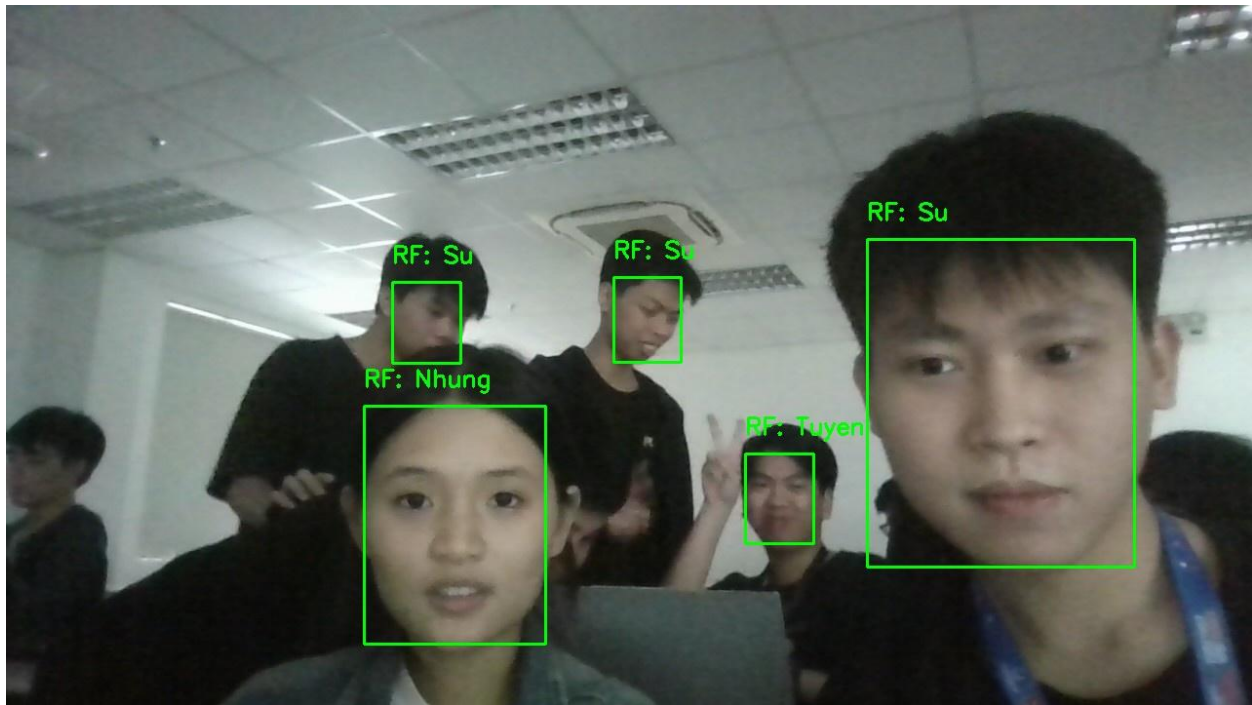
#### 4.2.2. Random Forest Model with Flattened Image Features (rf\_flat)



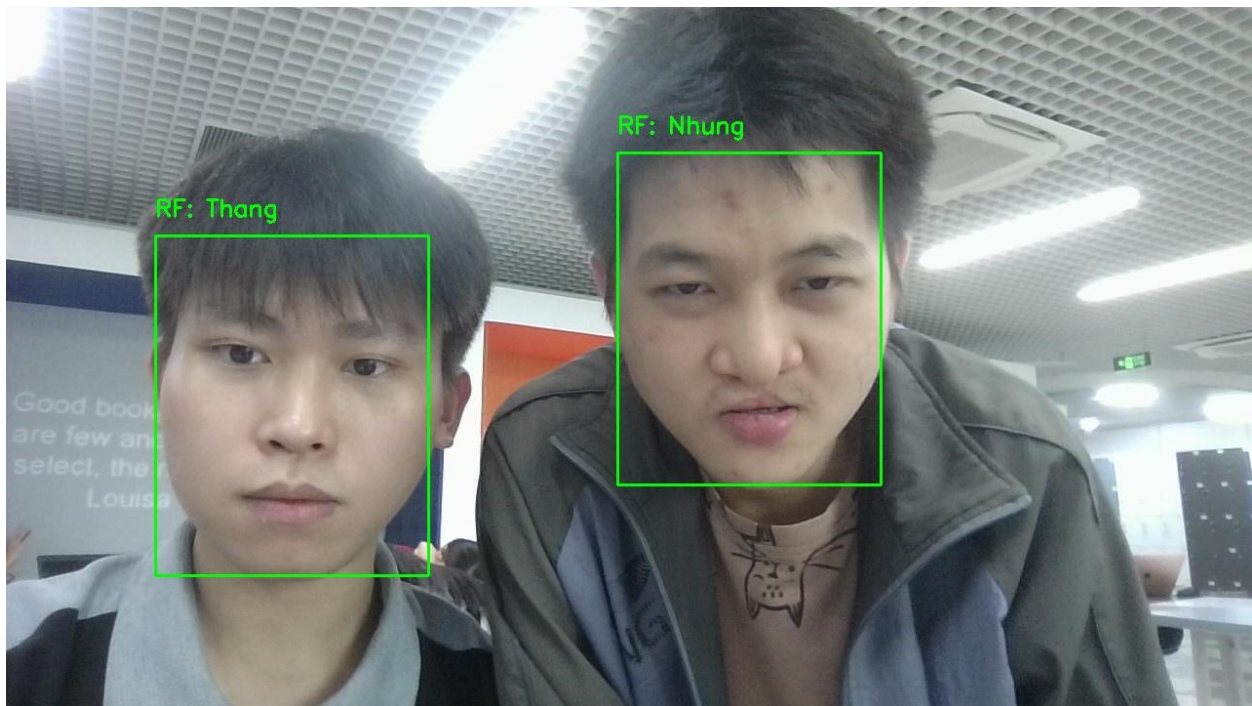
*Confusion matrix của Random Forest Model with Flattened Image Features*



*ROC curve của Random Forest Model with Flattened Image Features*

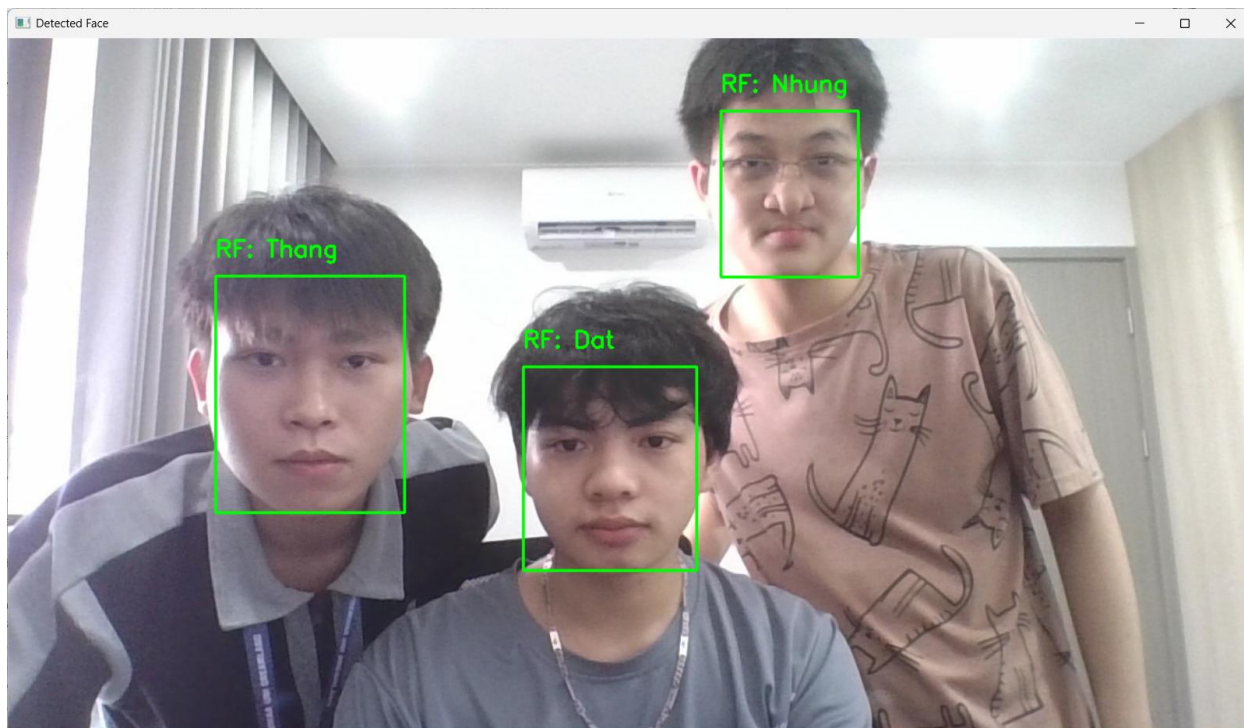


*Ảnh 2.4: Nhận diện với Random Forest Model with Flattened Image Features*



*Ảnh 2.5: Nhận diện với Random Forest Model with Flattened Image Features*





Ảnh 2.6: Nhận diện với *Random Forest Model with Flattened Image Features*

Kết quả nhận diện khuôn mặt với *Random Forest Model with Flattened Image Features*. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình

### => Kết luận:

Mô hình **Random Forest Model with Flattened Image Features** đạt kết quả nhận diện khuôn mặt tương đối tốt trên tập dữ liệu thử nghiệm, nhưng vẫn còn một số hạn chế nhất định:

- **Ma trận nhầm lẫn (confusion matrix):** Mô hình vẫn dự đoán sai nhãn ở một số trường hợp, điển hình như một mẫu của lớp “Nhung” bị nhầm sang “Tuyen”, ba mẫu của lớp “Tuyen” bị nhầm thành “Nhung” và ba mẫu của lớp “Vu” bị nhầm thành “Dat”. Điều này cho thấy khả năng phân biệt giữa các lớp vẫn chưa thực sự tối ưu khi chỉ sử dụng đặc trưng ảnh làm phẳng, đặc biệt là với các khuôn mặt có đặc điểm gần giống nhau.

- **Đường cong ROC và AUC:** Mặc dù các giá trị AUC đều đạt 1.00 cho từng lớp, nhưng thực tế vẫn xảy ra nhầm lẫn. Điều này cho thấy chỉ số AUC cao không phải lúc nào cũng đồng nghĩa với khả năng nhận diện hoàn hảo, đặc biệt khi các lớp có nhiều điểm tương đồng trong không gian đặc trưng.

### - Nhận diện thực tế:

- + Ở bức ảnh 4: mô hình đã nhận diện sai “Thang” thành “Su” đồng thời cả 3 khuôn mặt đằng sau đều không có trong tập dữ liệu đều bị mô hình gán nhãn sai. Thực tế nếu đúng thì

3 khuôn mặt đằng sau đây phải là nhãn “Unknown”. Điều này xảy ra trong điều kiện ánh sáng phòng học yếu và góc chụp rộng, có nhiều khuôn mặt xuất hiện cùng lúc

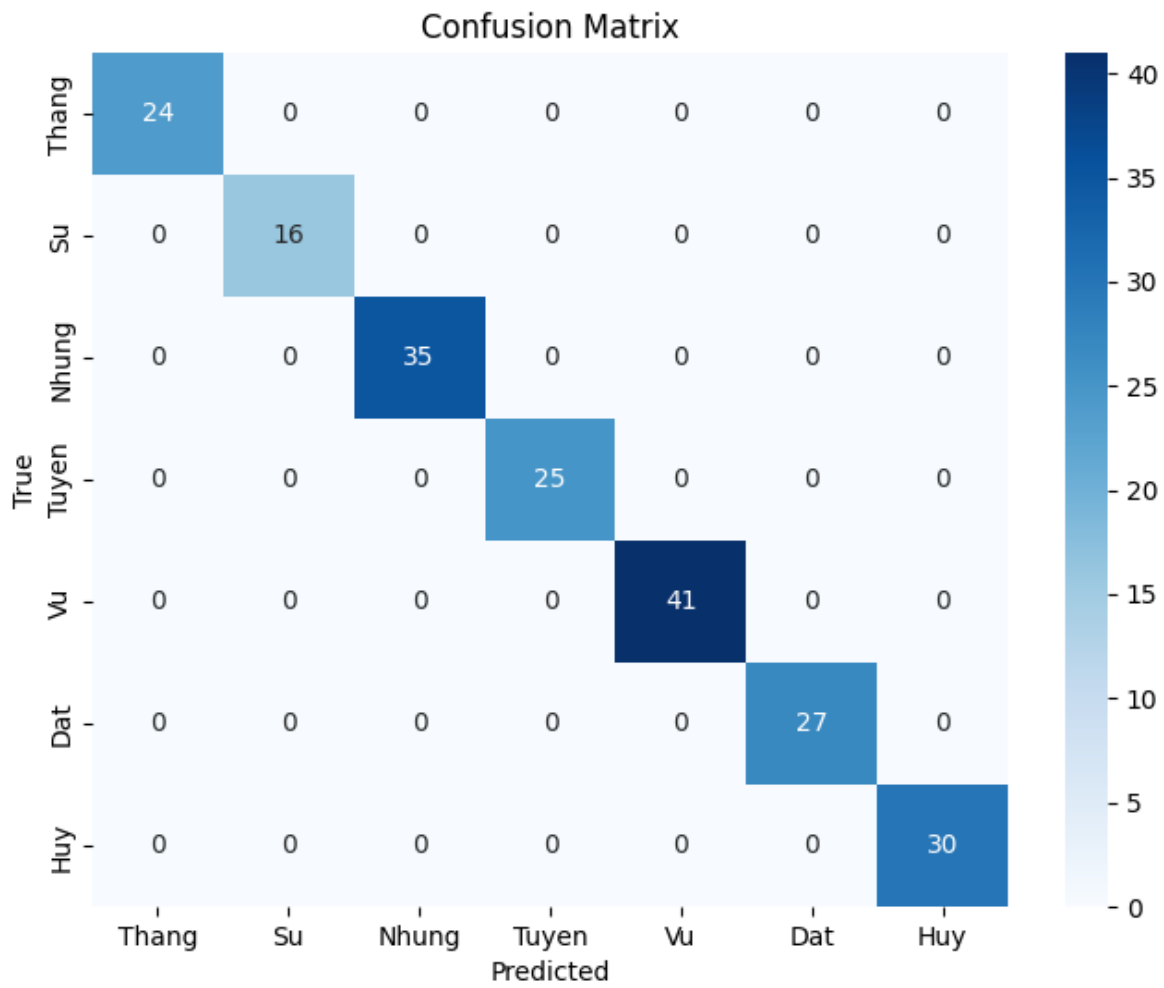
+ Ở bức ảnh 5: ở trong điều kiện ánh sáng đầy đủ nhưng mô hình đã gán nhãn “Nhưng” cho một khuôn mặt trong khi thực tế đó là “Tuyen”. Có thể thấy được sự nhầm lẫn khi các đặc trưng khuôn mặt tương đối giống nhau, kể cả trong điều kiện thuận lợi.

+ Ở bức ảnh 6: mô hình nhận diện “Thang” là đúng nhưng 2 khuôn mặt tiếp theo bị nhận nhầm là “Dat” và “Nhưng” trong thực tế phải “Su” và Tuyen”. Kết quả này cho thấy mô hình vẫn dễ nhầm lẫn giữa các đối tượng có đặc trưng khuôn mặt giống nhau, kể cả khi được chụp trong điều kiện ánh sáng tốt, môi trường rõ nét không có yếu tố gây nhiễu.

**Tổng kết lại: Random Forest Model with Flattened Image Features** có thể đáp ứng nhu cầu nhận diện khuôn mặt ở mức cơ bản, nhưng vẫn bộc lộ nhiều hạn chế về độ chính xác, đặc biệt trong những trường hợp các khuôn mặt có đặc điểm tương đồng hoặc khi điều kiện thực tế trở nên phức tạp (ví dụ như thay đổi ánh sáng, góc nhìn hay biểu cảm khuôn mặt). Kết quả này càng cho thấy tầm quan trọng của việc ứng dụng các phương pháp trích xuất đặc trưng sâu (deep feature) để cải thiện hiệu quả nhận diện và nâng cao khả năng tổng quát hóa của mô hình khi áp dụng trong thực tế.

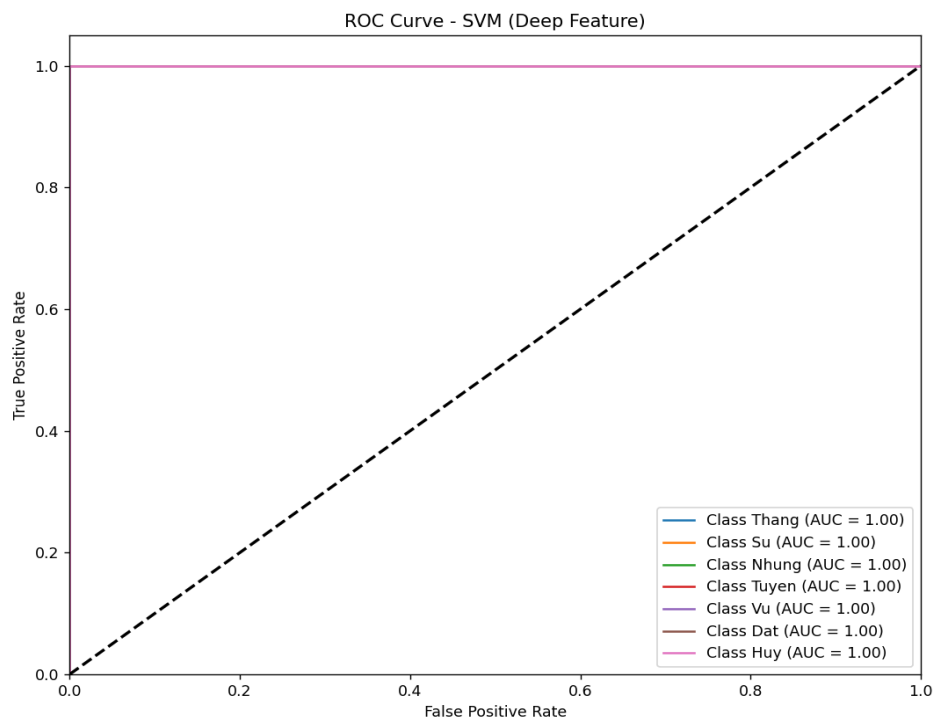
### 4.3. Support Vector Machine (SVM)

#### 4.3.1. Support Vector Machine Model with Extracted Features (svm\_ex)

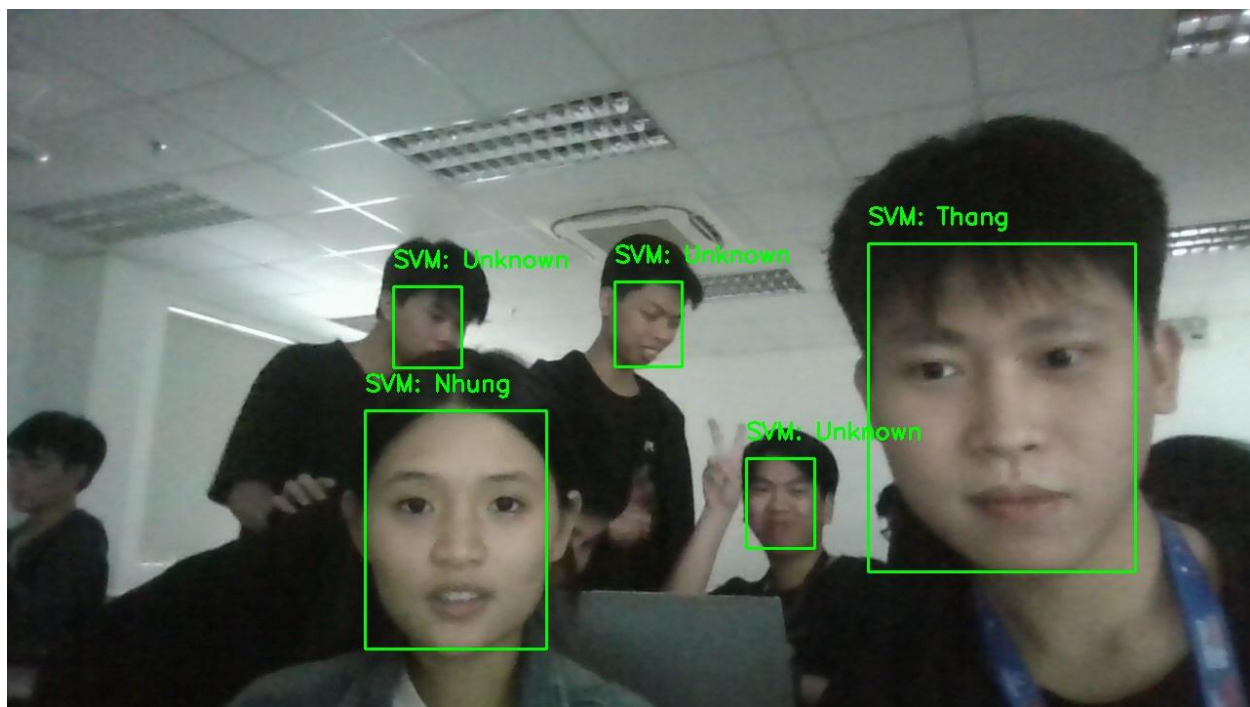


*Confusion matrix của Support Vector Machine Model with Extracted Features*

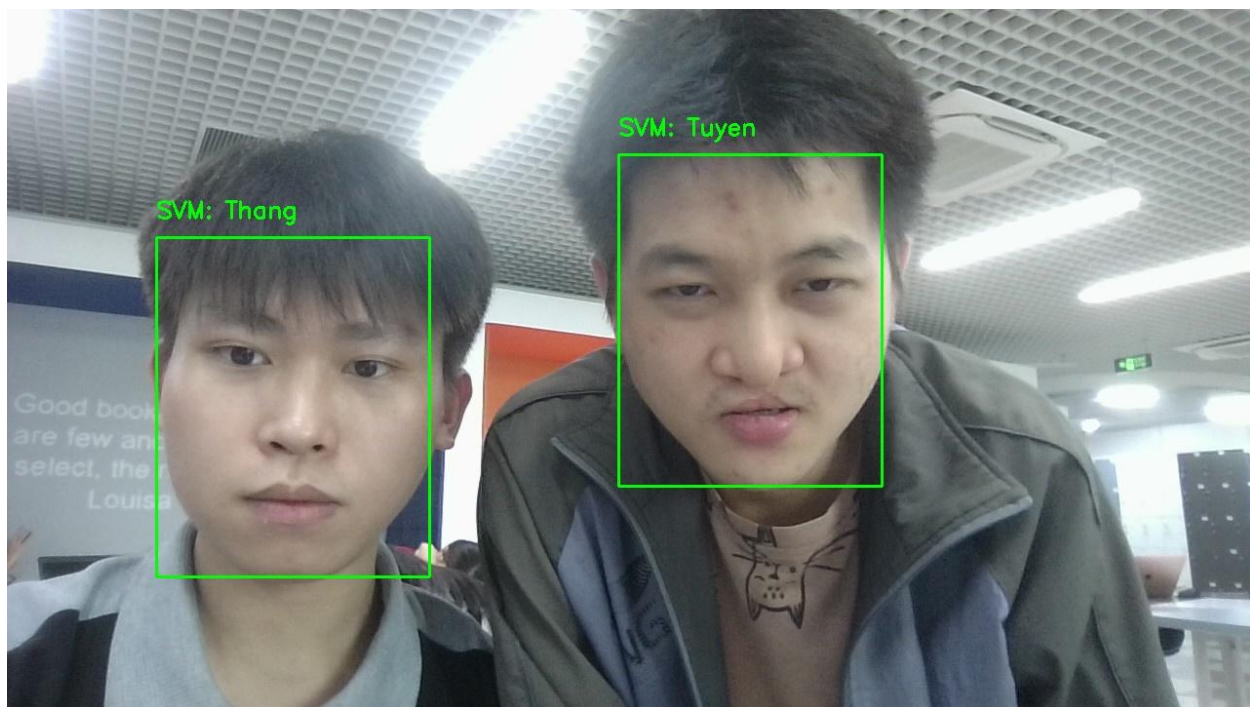




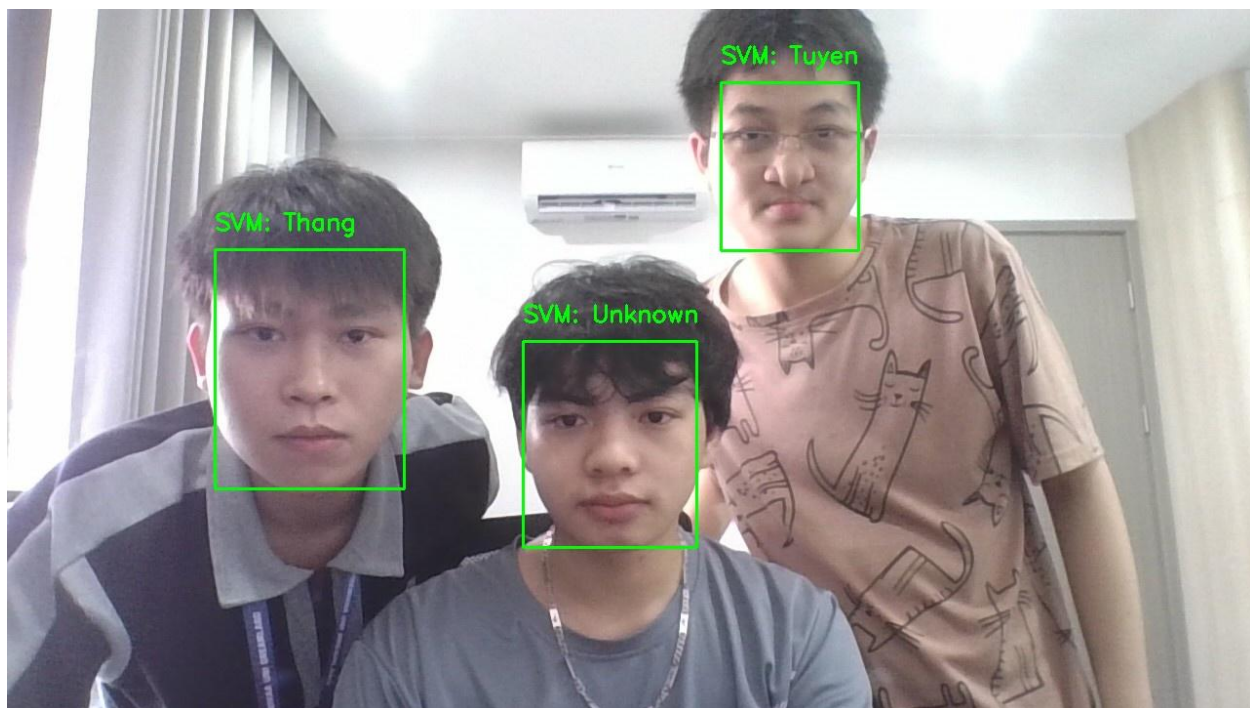
*ROC curve của Support Vector Machine Model with Extracted Features*



*Ảnh 3.1: nhận diện với Support Vector Machine Model with Extracted Features*



*Ảnh 3.2: nhận diện với Support Vector Machine Model with Extracted Features*



*Ảnh 3.3: nhận diện với Support Vector Machine Model with Extracted Features*

*Kết quả nhận diện khuôn mặt với Support Vector Machine Model with Extracted Features. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình*

### => **Kết luận:**

Sau khi thử nghiệm **Support Vector Machine Model with Extracted Features** đạt kết quả ấn tượng và cực kì ổn định:

- **Ma trận nhầm lẫn (confusion matrix):** Tất cả các lớp đều được dự đoán chính xác 100%, hoàn toàn không có trường hợp bị nhầm lẫn giữa các khuôn mặt. Điều này cho thấy mô hình đã học rất tốt các đặc trưng riêng biệt của từng người trong dữ liệu.

- **Đường cong ROC và chỉ số AUC:** Khi kiểm tra trên từng lớp, tất cả đều đạt AUC bằng 1.00, tức là khả năng phân biệt của mô hình giữa các lớp gần như tuyệt đối, rất khó có chuyện bị nhầm lẫn.

### - **Áp dụng vào thực tế:**

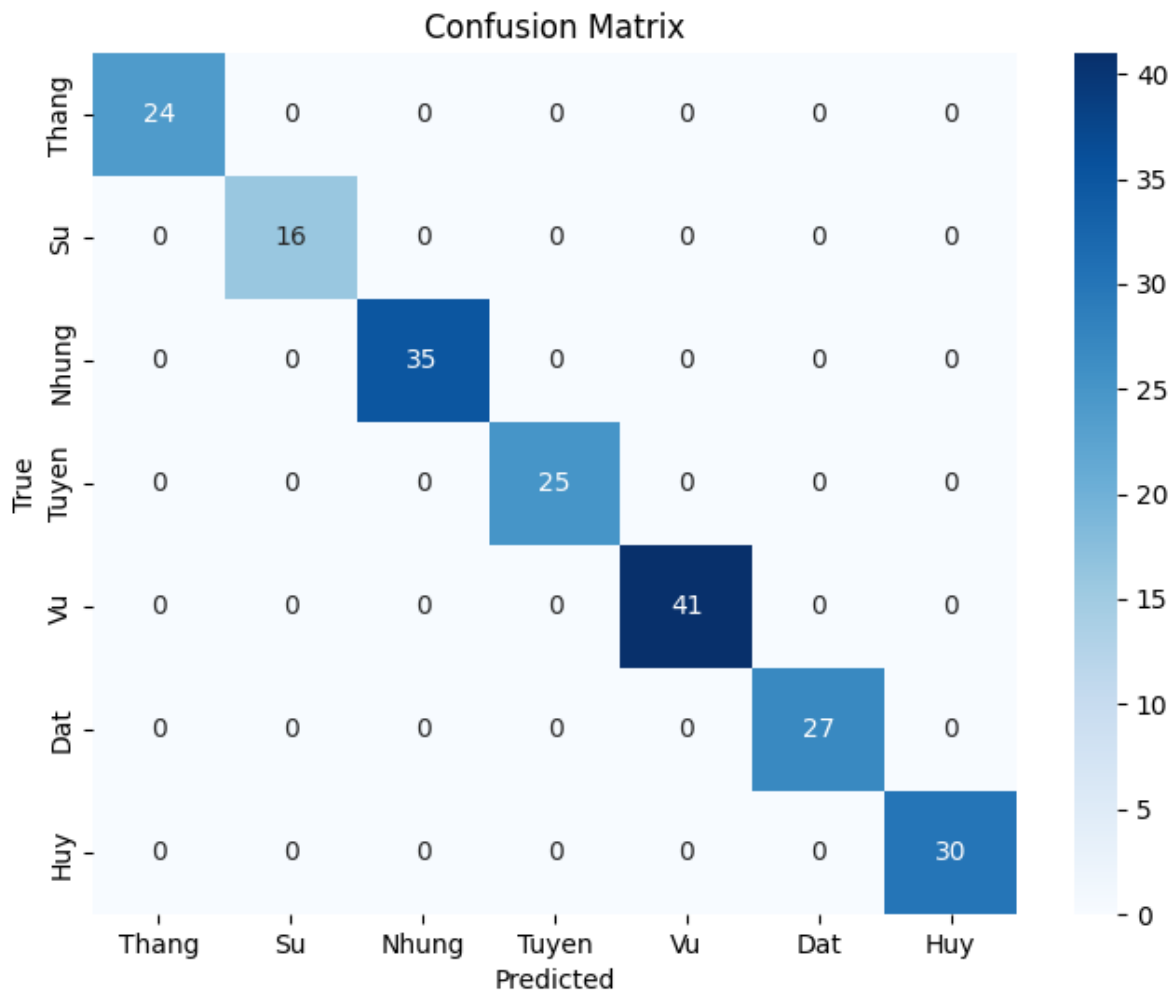
+ Ở bức ảnh 1 và 2: Mô hình đều nhận diện chính xác toàn bộ khuôn mặt trong điều kiện nhiều người và ánh sáng tốt, bố cục rõ ràng.

+ Ở bức ảnh 3: Khi nhận diện trên ảnh camera chụp trong điều kiện bình thường, đa số khuôn mặt đều được gán đúng tên. Tuy nhiên, vẫn có trường hợp khuôn mặt của “Su” bị mô hình nhận nhầm thành "Unknown", mặc dù đã có trong dữ liệu huấn luyện. Điều này cho thấy, dù mô hình rất hiệu quả trên tập kiểm tra, vẫn có thể xảy ra nhầm lẫn khi áp dụng thực tế.

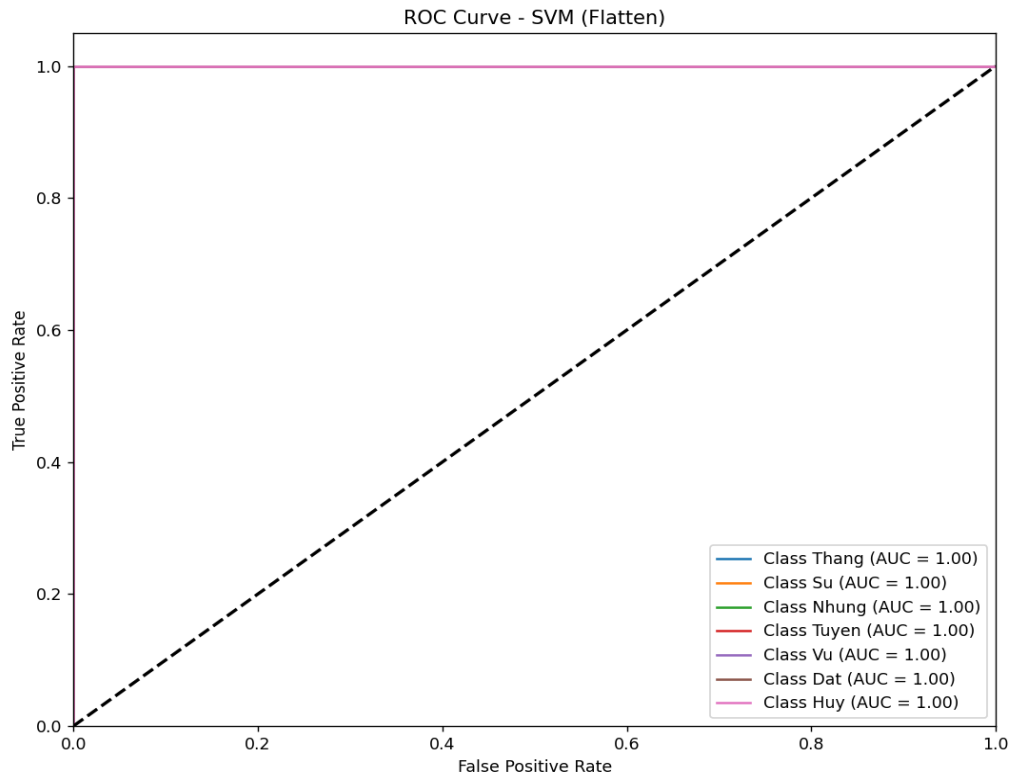
**Tổng kết lại, Support Vector Machine Model with Extracted Features** hoạt động cực kỳ hiệu quả trên tập dữ liệu thử nghiệm, không hề thua kém các mô hình mạnh như Random Forest hay KNN khi cùng sử dụng đặc trưng trích xuất sâu.

Tuy nhiên, để đảm bảo tính ổn định và tổng quát hóa khi triển khai thực tế, cần tiếp tục kiểm thử với dữ liệu đa dạng hơn về bối cảnh, ánh sáng và biểu cảm khuôn mặt. Kết quả cũng nhấn mạnh rằng sự kết hợp giữa đặc trưng học sâu và mô hình máy học truyền thống đem lại hiệu quả rất cao cho bài toán nhận diện khuôn mặt, nhưng vẫn cần tối ưu thêm để hạn chế tối đa các trường hợp nhận nhầm trong thực tế.

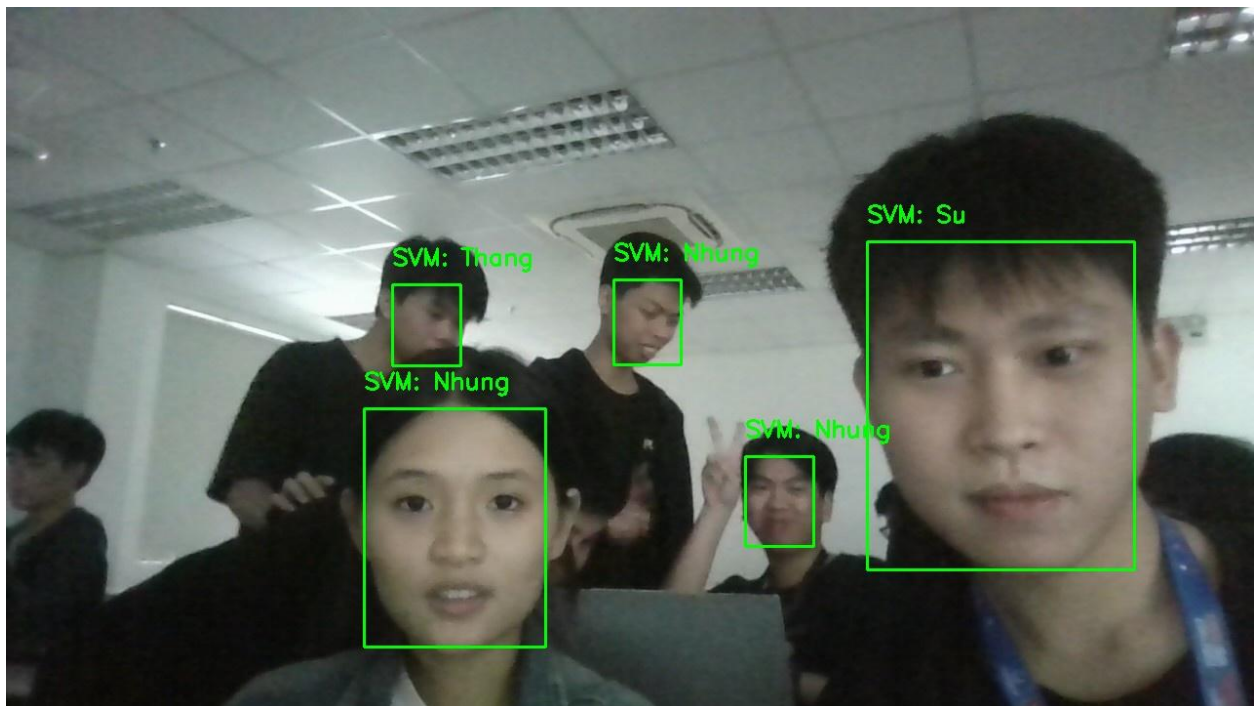
### 4.3.2. Support Vector Machine Model with Flattened Image Features (svm\_flat)



*Confusion matrix của Support Vector Machine Model with Flattened Image Features*

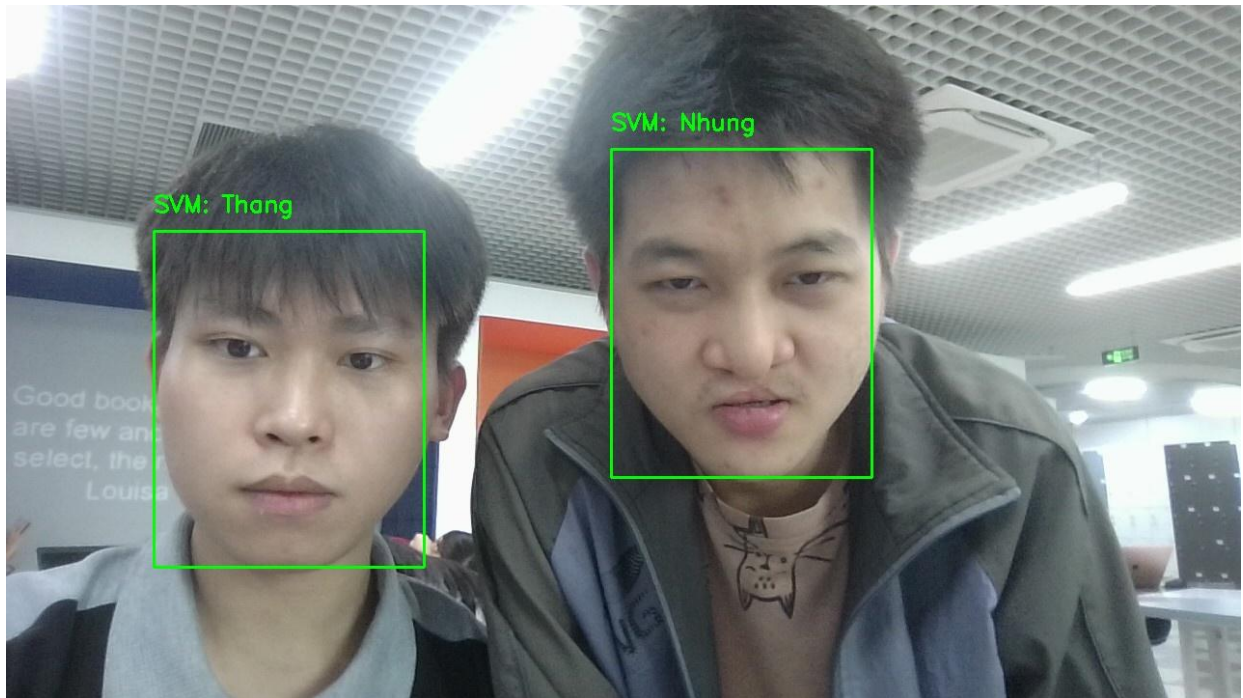


*ROC curve của Support Vector Machine Model with Flattened Image Features*

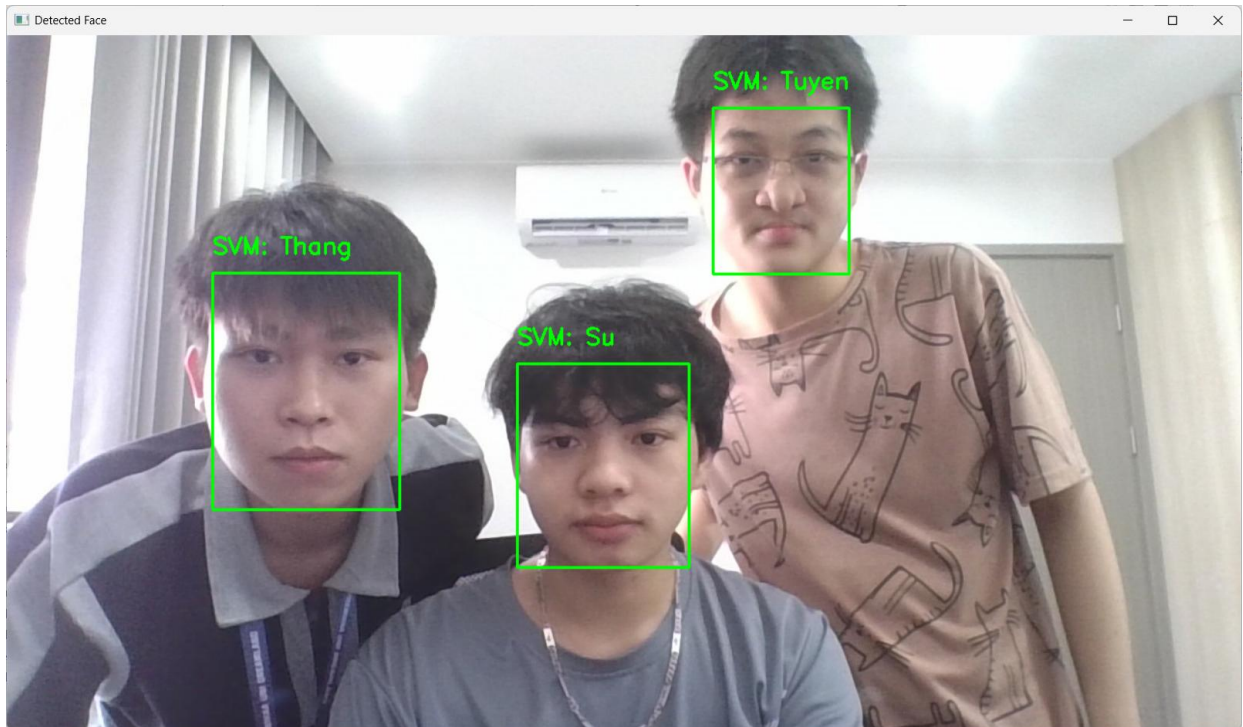


*Ảnh 3.4: Nhận diện với Support Vector Machine Model with Flattened Image Features*





*Ảnh 3.5: Nhận diện với Support Vector Machine Model with Flattened Image Features*



*Ảnh 3.6: Nhận diện với Support Vector Machine Model with Flattened Image Features*

*Kết quả nhận diện khuôn mặt với Support Vector Machine Model with Flattened Image Features. Mỗi khuôn mặt được phát hiện và gán nhãn dựa trên kết quả dự đoán của mô hình.*

## => Kết luận

Mô hình **Support Vector Machine Model with Flattened Image Features** tiếp tục cho thấy hiệu quả nhận diện rất tốt trên tập dữ liệu hiện tại:

- **Ma trận nhầm lẫn (confusion matrix):** Tất cả các lớp đều được nhận diện đúng 100%, không hề có trường hợp nào bị nhầm lẫn giữa các khuôn mặt.

- **Đường cong ROC và AUC:** Mỗi lớp đều đạt chỉ số AUC tuyệt đối 1.00, cho thấy khả năng phân biệt giữa các lớp của mô hình là hoàn hảo.

- **Kết quả nhận diện thực tế:**

- + Ở bức ảnh 4: mô hình đã nhận diện sai “Thang” thành “Su” đồng thời cả 3 khuôn mặt đằng sau đều không có trong tập dữ liệu đều bị mô hình gán nhãn sai. Thực tế nếu đúng thì 3 khuôn mặt phía sau phải là nhãn “Unknown”. Điều này xảy ra trong điều kiện ánh sáng phòng học yếu và góc chụp rộng, có nhiều khuôn mặt xuất hiện cùng lúc.

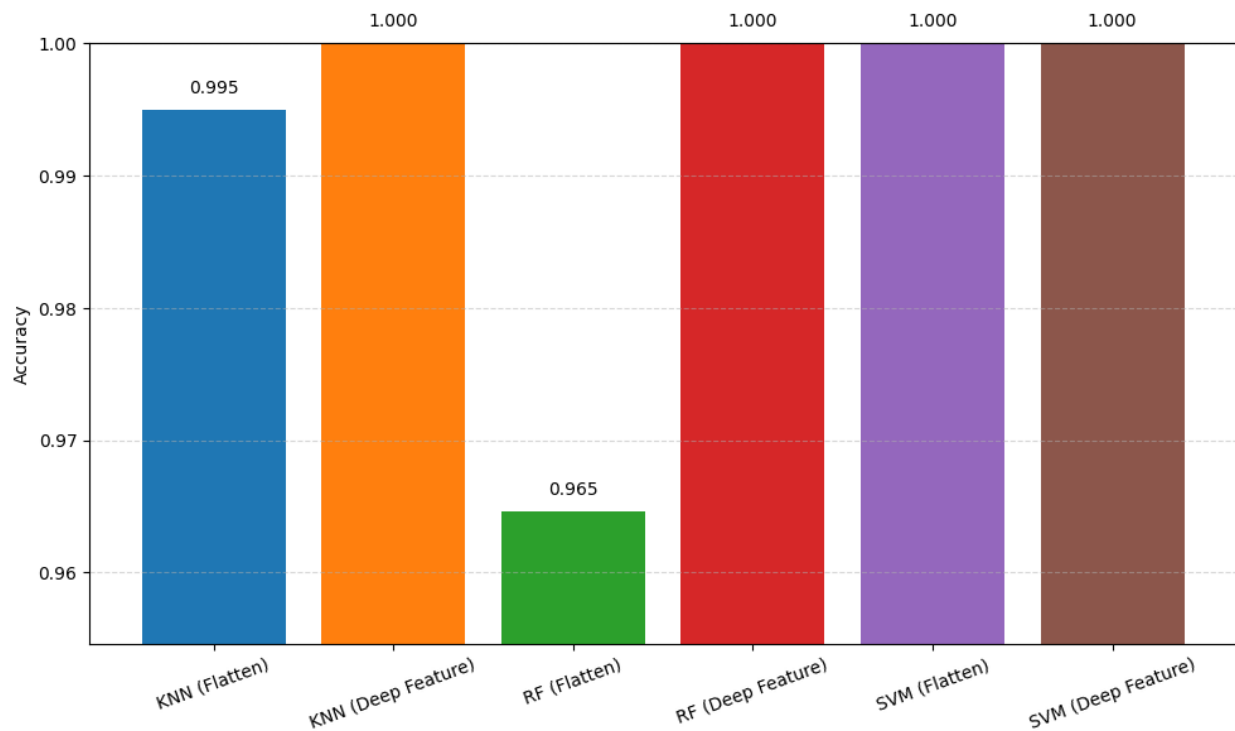
- + Ở bức ảnh 5: ở trong điều kiện ánh sáng đầy đủ nhưng mô hình đã gán nhãn “Nhưng” cho một khuôn mặt trong khi thực tế đó là “Tuyen”. Có thể thấy được sự nhầm lẫn khi các đặc trưng khuôn mặt tương đối giống nhau, kể cả trong điều kiện thuận lợi.

- + Ở bức ảnh 6: Khi thử nghiệm nhận diện trên ảnh chụp từ webcam, mô hình cũng gán nhãn chính xác cho từng khuôn mặt, không có sai sót nào.

**Tóm lại: Support Vector Machine Model with Flattened Image Features** đã nhận diện chính xác hoàn toàn trên bộ dữ liệu thử nghiệm, chứng tỏ mô hình này học và phân loại rất tốt, mặc dù chỉ dùng đặc trưng đơn giản. Tuy vậy, để đảm bảo mô hình thực sự ổn định và ứng dụng tốt ngoài thực tế, vẫn nên thử nghiệm thêm với nhiều loại dữ liệu đa dạng hơn như các góc nhìn khác nhau, ánh sáng khác nhau, hay các trường hợp phức tạp hơn về biểu cảm hoặc bối cảnh.

## 4.4. So sánh

Model	Knn_ex	knn_flat	rf_ex	rf_flat	svm_ex	svm_flat
Accuracy	100%	99.49%	100%	96.46%	100%	100%



*Bảng so sánh độ chính xác giữa 6 models*

Sau khi huấn luyện và kiểm thử trên tập dữ liệu, kết quả của các mô hình như sau:

- **KNN (knn\_ex):** Đạt độ chính xác 100% trên tập kiểm tra.
- **KNN (knn\_flat):** Đạt độ chính xác 99.49% trên tập kiểm tra.
- **Random Forest (rf\_ex):** Đạt độ chính xác 100% trên tập kiểm tra.
- **Random Forest (rf\_flat):** Đạt độ chính xác 96.46% trên tập kiểm tra.
- **SVM (svm\_ex):** Đạt độ chính xác 100% trên tập kiểm tra.
- **SVM (svm\_flat):** Đạt độ chính xác 100% trên tập kiểm tra.

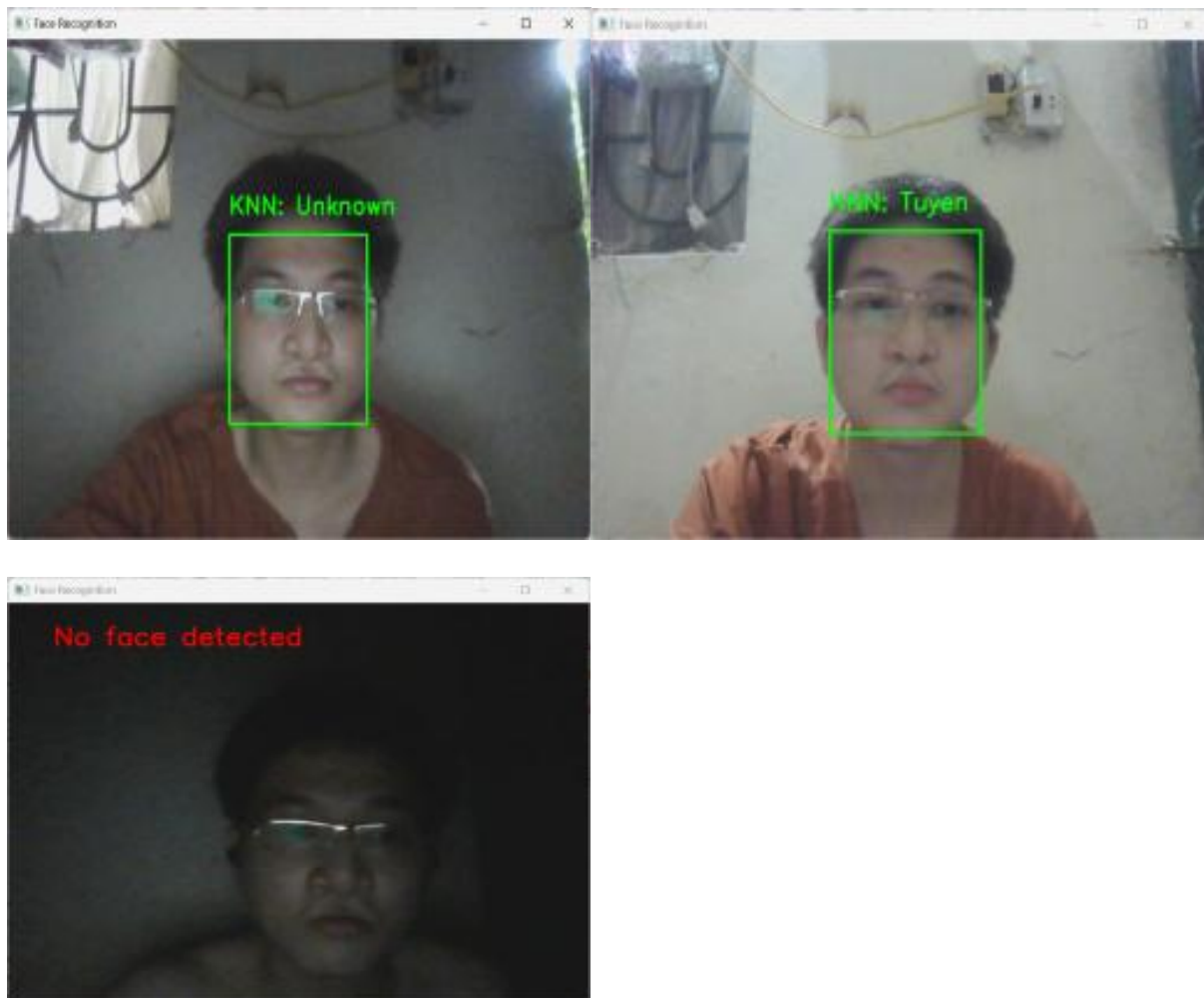
Kết quả cho thấy các mô hình KNN, Random Forest và SVM với pipeline \*\_ex đều đạt độ chính xác tối đa 100% trên tập kiểm tra. Ngược lại, các pipeline sử dụng ảnh làm phẳng



có độ chính xác thấp hơn, trong đó rf\_flat đạt thấp nhất với 96.46%. Điều này khẳng định việc sử dụng đặc trưng trích xuất từ deep learning giúp mô hình nhận diện hiệu quả và ổn định hơn so với sử dụng ảnh làm phẳng.

## Phần 5. Chạy mô hình trong thời gian thực

Trong quá trình đánh giá trên tập dữ liệu thử nghiệm của cả 3 mô hình, SVM và RF đều đạt hiệu suất tốt hơn so với KNN. Tuy nhiên, khi triển khai nhận diện khuôn mặt thời gian thực, chúng em đã lựa chọn mô hình KNN nhờ ưu điểm về cấu trúc đơn giản, dễ triển khai và tốc độ xử lý nhanh. Trên tập dữ liệu hiện tại, KNN cho thấy khả năng nhận diện ổn định và mang lại kết quả đáng tin cậy khi điều kiện ánh sáng thuận lợi. Tuy nhiên, hiệu quả của KNN giảm rõ rệt trong môi trường ánh sáng yếu, đặc biệt là khi quá tối thì mô hình hầu như không thể nhận diện được khuôn mặt. Ngoài ra, mô hình cũng bị ảnh hưởng nếu góc nhìn khuôn mặt thay đổi nhiều. Ưu điểm lớn nhất của KNN là nhẹ, dễ tích hợp vào hệ thống và vận hành ổn định trong các điều kiện tiêu chuẩn.



*Chạy mô hình K-Nearest Neighbors nhận diện khuôn mặt trong thời gian thực bằng camera*

## Phần 6. Kết luận

Trong đề tài này, chúng em đã nghiên cứu, xây dựng và đánh giá các mô hình học máy khác nhau cho bài toán nhận diện khuôn mặt từ ảnh chụp bằng camera. Mục tiêu của đề tài là phát triển một phương pháp nhận diện khuôn mặt tự động có độ chính xác cao, góp phần phục vụ các ứng dụng thực tiễn như kiểm soát an ninh, điểm danh thông minh và các hệ thống giao tiếp người-máy.

Chúng em đã thử nghiệm sáu pipeline mô hình khác nhau, bao gồm ba thuật toán phổ biến là K-Nearest Neighbors (KNN), Support Vector Machine (SVM) và Random Forest, kết hợp với hai phương pháp xử lý đầu vào là ảnh làm phẳng (flattened images) và đặc trưng trích xuất từ deep learning. Các mô hình đều được huấn luyện và đánh giá trên tập dữ liệu được gán nhãn rõ ràng với nhiều cá nhân khác nhau. Kết quả thực nghiệm cho thấy các pipeline sử dụng đặc trưng trích xuất từ deep learning (knn\_ex, svm\_ex, rf\_ex) đều đạt độ chính xác tối đa 100%, trong khi các pipeline sử dụng ảnh làm phẳng cho kết quả thấp hơn một chút, với mô hình rf\_flat đạt 96.46%.

Những kết quả này cho thấy việc lựa chọn đặc trưng đầu vào có vai trò quan trọng trong việc nâng cao hiệu suất của mô hình nhận diện khuôn mặt. Đặc biệt, việc sử dụng đặc trưng trích xuất từ deep learning giúp mô hình tổng quát hóa tốt hơn và nhận diện hiệu quả hơn trong nhiều tình huống khác nhau.

Mặc dù hệ thống hiện tại mới dừng lại ở mức đánh giá trên tập dữ liệu ảnh tĩnh, kết quả thu được đã khẳng định tiềm năng ứng dụng của các mô hình vào các hệ thống nhận diện khuôn mặt tự động trong thực tế. Trong các nghiên cứu tiếp theo, chúng em dự định sẽ triển khai mô hình trên dữ liệu video hoặc trực tiếp từ camera để kiểm chứng khả năng nhận diện khuôn mặt trong môi trường thời gian thực, đánh giá độ ổn định và tốc độ xử lý của hệ thống.

Ngoài ra, các hướng phát triển tiếp theo có thể bao gồm việc mở rộng và đa dạng hóa tập dữ liệu (nhiều điều kiện ánh sáng, góc chụp, biểu cảm khuôn mặt...), áp dụng các kiến trúc học sâu tiên tiến như mạng nơ-ron tích chập (CNN), transformer hoặc các kỹ thuật tăng cường dữ liệu hiện đại để tiếp tục nâng cao độ chính xác và độ bền vững của mô hình.

Nhìn chung, đề tài này đã đặt nền tảng vững chắc cho các bước phát triển sâu hơn trong lĩnh vực nhận diện khuôn mặt tự động và mở ra nhiều tiềm năng ứng dụng trong các hệ thống thông minh phục vụ đời sống hiện đại.