

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA KHOA HỌC MÁY TÍNH**



**BÁO CÁO ĐỒ ÁN  
NHẬP MÔN THỊ GIÁC MÁY TÍNH**

**ĐỀ TÀI  
NHẬN DIỆN CẢM XÚC KHUÔN MẶT**

GV hướng dẫn: Mai Tiến Dũng

Lớp: CS231.N21.KHCL

Nhóm thực hiện:

1. Lê Trần Bảo Lợi - 21522295
2. Bùi Đình Quân - 21522487
3. Lê Đào Xuân Thành - 21522595

TP.HCM, tháng 06 năm 2023

# MỤC LỤC

<b>1</b>	<b>GIỚI THIỆU BÀI TOÁN</b>	<b>1</b>
1.1	Lý do lựa chọn đề tài . . . . .	1
1.2	Bài toán . . . . .	1
<b>2</b>	<b>CÁC PHƯƠNG PHÁP SỬ DỤNG</b>	<b>1</b>
2.1	Một số phương pháp . . . . .	1
2.2	Phương pháp sử dụng . . . . .	1
2.3	Các phương pháp phân loại . . . . .	3
2.3.1	Logistic Regression . . . . .	3
2.3.2	K-Nearest Neighbors (KNN) . . . . .	3
2.3.3	Support Vector Machine (SVM) . . . . .	4
<b>3</b>	<b>THỰC NGHIỆM TRÊN DATASET</b>	<b>5</b>
3.1	Dataset . . . . .	5
3.2	Môi trường thực nghiệm . . . . .	5
3.3	Độ đo đánh giá . . . . .	6
3.4	Tổ chức dữ liệu . . . . .	6
3.5	Label Encoder . . . . .	6
3.6	Kết quả thực nghiệm . . . . .	7
3.6.1	Feature Extraction with Sobel . . . . .	7
3.6.2	Feature Extraction with HOG . . . . .	8
<b>4</b>	<b>SO SÁNH CÁC PHƯƠNG PHÁP</b>	<b>9</b>
<b>5</b>	<b>KẾT QUẢ THỰC NGHIỆM TRÊN TẬP TEST</b>	<b>9</b>
<b>6</b>	<b>KẾT LUẬN</b>	<b>9</b>
<b>7</b>	<b>TÀI LIỆU THAM KHẢO</b>	<b>10</b>



# 1 GIỚI THIỆU BÀI TOÁN

## 1.1 Lý do lựa chọn đề tài

- Ứng dụng của bài toán nhận diện cảm xúc con người rất đa dạng
- Dự đoán cảm xúc trở thành nhiệm vụ quan trọng trong các ứng dụng thông minh ngày nay
- Nhiệm vụ chính của bài toán là xác định và phân loại cảm xúc của một người dựa trên hình ảnh khuôn mặt của họ

⇒ Mục tiêu là phân loại hình ảnh khuôn mặt thành các nhãn tương ứng với cảm xúc này

## 1.2 Bài toán

- Input: Một tấm ảnh màu hoặc xám chính diện của khuôn mặt bao gồm mắt, mũi, miệng, tai và được gán nhãn một trong bảy cảm xúc trong tập huấn luyện
- Output: Một trong bảy cảm xúc trong dữ liệu huấn luyện (happy, angry, fear, disgust, sad, surprise, neutral)

# 2 CÁC PHƯƠNG PHÁP SỬ DỤNG

## 2.1 Một số phương pháp

- Dựa trên mô hình học sâu(deep learning): sử dụng các mô hình học sâu như CNN để học và phân loại cảm xúc từ các hình ảnh khuôn mặt. Mô hình CNN có khả năng tự học và rút trích các đặc trưng phức tạp từ dữ liệu hình ảnh.

- Dựa trên các đặc trưng của hình ảnh: Phương pháp này sử dụng các đặc trưng hình ảnh như hình dạng mắt, miệng, mũi và các đường nét khuôn mặt để phân loại cảm xúc. Sau đó dùng các thuật toán phân loại như SVM, KNN... để phân loại.

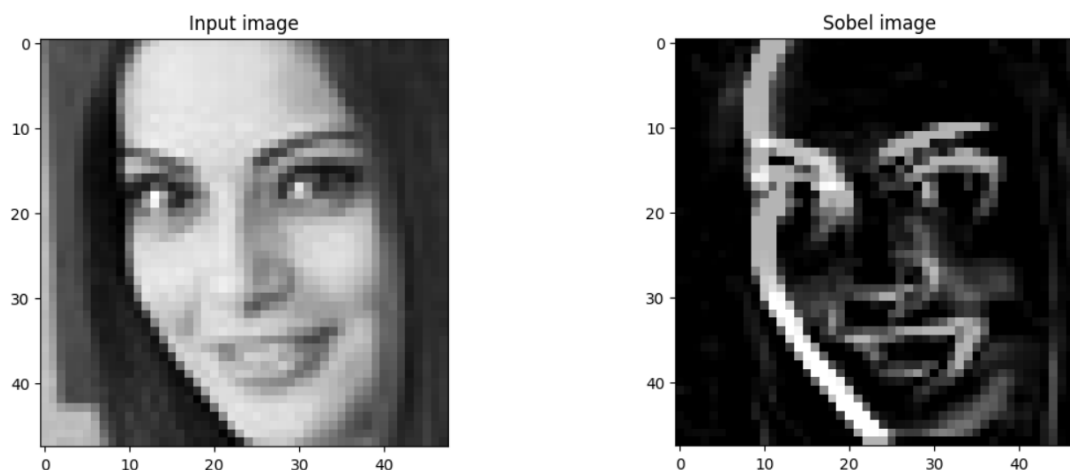
## 2.2 Phương pháp sử dụng

### - Các phương pháp Feature Extraction:

- + Sobel
- + Histogram of Oriented Gradients (HOG)

## - Sobel

Sobel có khả năng tìm ra các vùng cạnh trong hình ảnh. Các cạnh thường tương ứng với các sự thay đổi mạnh về độ sáng hoặc màu sắc trong hình ảnh qua đó giúp xác định hình dạng và cấu trúc các đối tượng trong ảnh.

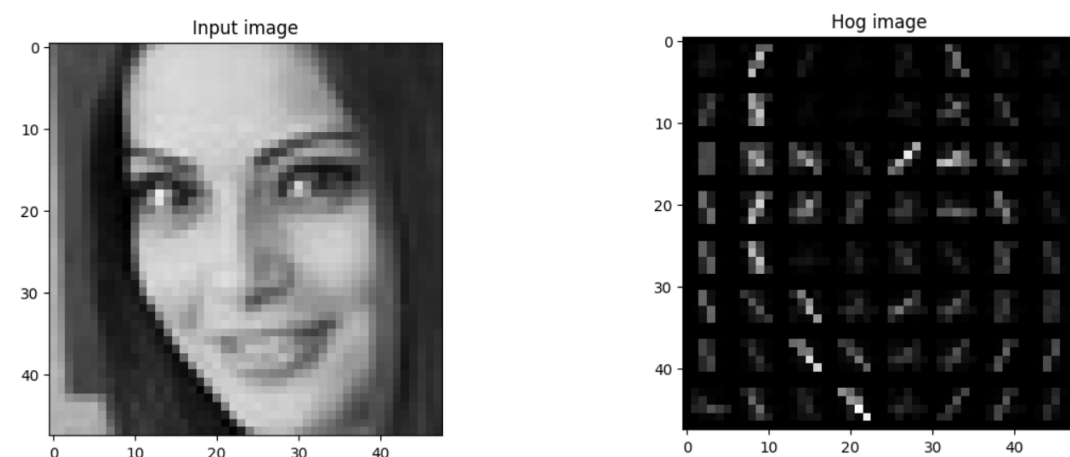


## - HOG

HOG (histogram of oriented gradients) là một feature descriptor được sử dụng trong computervision và xử lý hình ảnh, dùng để detect một đối tượng.

HOG được sử dụng chủ yếu để mô tả hình dạng và sự xuất hiện của một object trong ảnh. Bài toán tính toán Hog thường gồm 5 bước:

- + Chuẩn hóa hình ảnh trước khi xử lý.
- + Tính toán gradient theo cả hướng x và y.
- + Lấy phiếu bầu cùng trọng số trong các cell.
- + Chuẩn hóa các block.
- + Thu thập tất cả các biểu đồ cường độ gradient định hướng để tạo ra feature vector cuối cùng.



## 2.3 Các phương pháp phân loại

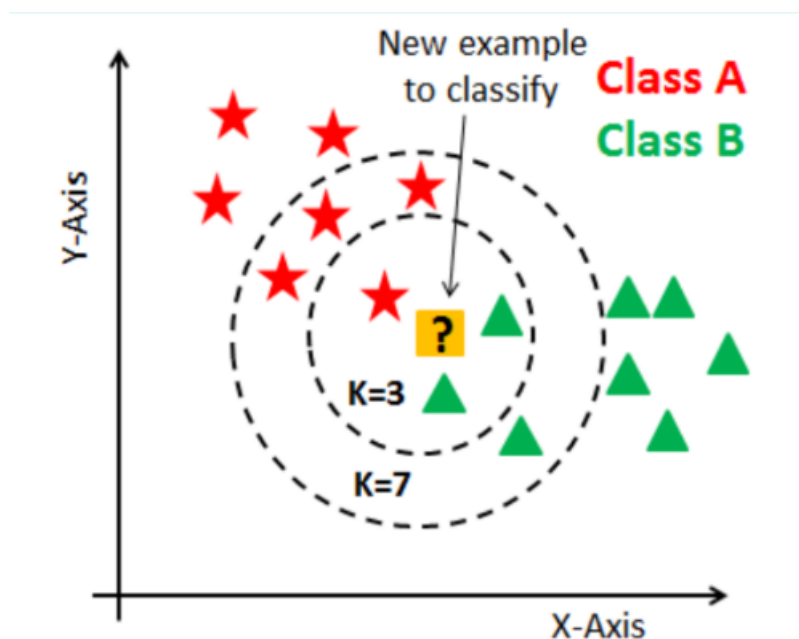
### 2.3.1 Logistic Regression

- Logistic Regression là thuật toán học máy có giám sát.
- Là một kỹ thuật phân tích dữ liệu sử dụng toán học để tìm ra mối quan hệ giữa hai yếu tố dữ liệu.
- Thường được dùng trong các bài toán phân loại và hồi quy.



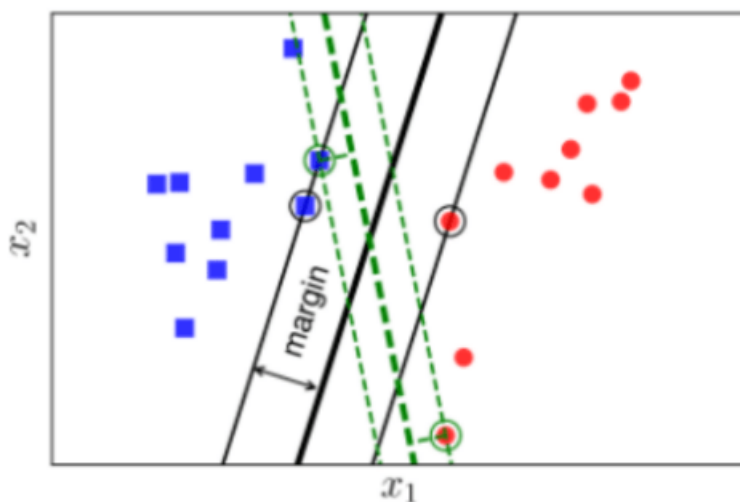
### 2.3.2 K-Nearest Neighbors (KNN)

- K-nearest neighbors là thuật toán học máy có giám sát, đơn giản và dễ triển khai.
- Thường được dùng trong các bài toán phân loại và hồi quy.
- Thuật toán này xác định lớp của một điểm dữ liệu mới bằng cách so sánh nó với k điểm dữ liệu gần nhất trong tập huấn luyện



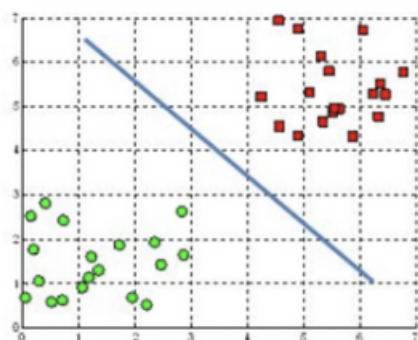
### 2.3.3 Support Vector Machine (SVM)

- SVM là một phương pháp máy học được Vladimir Vapnik và đồng nghiệp của ông tại Viện Tin học Bell Labs công bố lần đầu tiên vào năm 1992
- SVM là một thuật toán khá hiệu quả trong các bài toán phân loại và dự báo của học có giám sát.
- SVM luôn tìm Margin lớn nhất để giảm thiểu việc phân lớp sai đối với điểm dữ liệu mới đưa vào.
- Margin của một lớp được định nghĩa là khoảng cách từ các điểm gần nhất của lớp đó tới mặt phân chia.
- Ý tưởng của SVM là margin của 2 lớp phải bằng nhau và phải lớn nhất có thể.

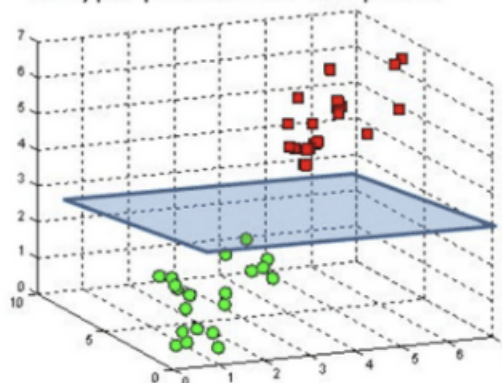


- Phương pháp SVM là tìm một siêu mặt phẳng (hyperplane) để phân tách các điểm dữ liệu.
- Siêu mặt phẳng này sẽ chia không gian thành các miền khác nhau và mỗi miền sẽ chứa một loại dữ liệu, phải có margin cực đại và phân chia 2 miền và cách đều chúng.

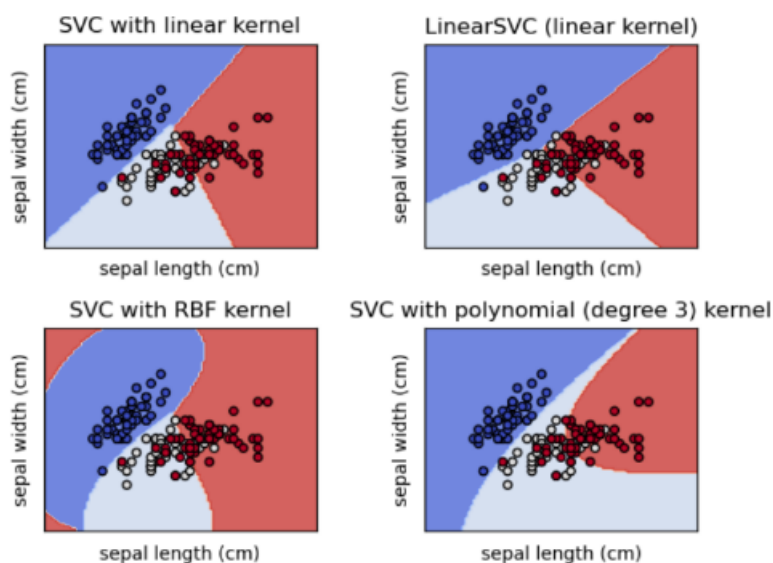
A hyperplane in  $\mathbb{R}^2$  is a line



A hyperplane in  $\mathbb{R}^3$  is a plane



- SVM là công cụ đắc lực trong các bài toán về xử lý ảnh, phân loại văn bản, phân tích dữ liệu ...
- Là một trong những mô hình được nghiên cứu và sử dụng rất nhiều trong các bài toán phân loại dùng để phân chia dữ liệu thành các nhóm riêng biệt.
- SVM rất hiệu quả cho việc giải quyết các bài toán có dữ liệu lớn, giải quyết vấn đề overfitting rất tốt, phân lớp nhanh, có hiệu quả cao.
- Ở bài toán "Nhận diện cảm xúc khuôn mặt" nhóm em sử dụng SVM có sẵn trong thư viện sklearn và kernel đã được sử dụng sẽ là "rbf" và "polynomial".



### 3 THỰC NGHIỆM TRÊN DATASET

#### 3.1 Dataset

- Tập dữ liệu "FER-2013" từ kaggle.
- Dữ liệu bao gồm hình ảnh thang độ xám 48x48 pixel của khuôn mặt.
- Tập dữ liệu huấn luyện bao gồm 28.709 (80%)
- Tập dữ liệu kiểm tra bao gồm 7.178 (20%)
- Dữ liệu được thu thập bởi API tìm kiếm hình ảnh của Google.

#### 3.2 Môi trường thực nghiệm

- Google cung cấp một môi trường thực nghiệm trực tuyến miễn phí đó là Google Colab.
- Colab cung cấp GPU miễn phí để chạy các tác vụ máy học phức tạp.



- Colab cung cấp những thư viện phổ biến mà không cần cài đặt như là: numpy, matplotlib, tensorflow, ...
- Colab giúp làm việc nhóm hiệu quả và dễ sử dụng.

### 3.3 Độ đo đánh giá

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

### 3.4 Tổ chức dữ liệu



(a) label 1



(b) label 2

### 3.5 Label Encoder

```
def label_encoder(emo):  
    if emo == 'angry':  
        return 0  
    elif emo == 'disgust':  
        return 1  
    elif emo == 'fear':  
        return 2  
    elif emo == 'happy':  
        return 3  
    elif emo == 'neutral':  
        return 4  
    elif emo == 'sad':  
        return 5  
    elif emo == 'surprise':  
        return 6
```





## 3.6 Kết quả thực nghiệm

### 3.6.1 Feature Extraction with Sobel

#### Logistic Regression

Accuracy:

```
y_pred = model_lr.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.3358874338255782

#### K-Nearest Neighbors (KNN)

Accuracy:

```
for i in range(len(K)):
    print('K:', K[i], 'Accuracy:', Acc[i])
```

K: 1 Accuracy: 0.3790749512398997  
K: 3 Accuracy: 0.32836444692114797  
K: 5 Accuracy: 0.33324045695179716  
K: 7 Accuracy: 0.3293396489272778  
K: 9 Accuracy: 0.3269713011981053

#### Support Vector Machine (SVM)

RBF Kernel:

```
y_pred = model_rbf.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.4526330454165506

Polynomial Kernel:

```
y_pred = model_poly.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.46447478406241294



### 3.6.2 Feature Extraction with HOG

#### Logistic Regression

Accuracy:

```
y_pred = model_lr.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.4476177208135971

#### K-Nearest Neighbors (KNN)

Accuracy:

```
for i in range(len(K)):
    print('K:', K[i], 'Accuracy:', Acc[i])
```

K: 1 Accuracy: 0.5228475898578991  
K: 3 Accuracy: 0.48704374477570356  
K: 5 Accuracy: 0.4864864864864865  
K: 7 Accuracy: 0.4786848704374478  
K: 9 Accuracy: 0.48063527444970744

#### Support Vector Machine (SVM)

RBF Kernel:

```
y_pred = model_rbf.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.5423516299804959

Polynomial Kernel:

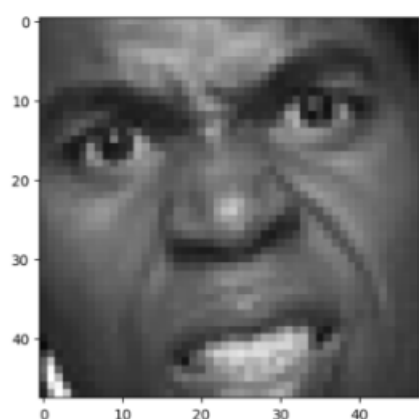
```
y_pred = model_poly.predict(X_test)
acc = accuracy_score(y_pred, y_test)
print("Accuracy:", acc)
```

Accuracy: 0.56993591529674

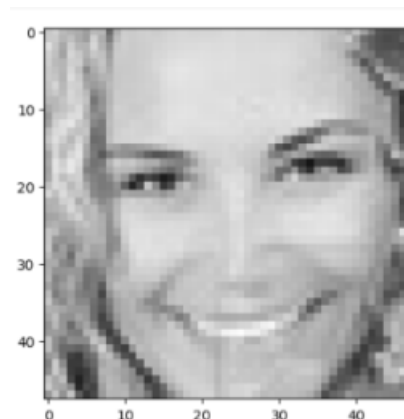
## 4 SO SÁNH CÁC PHƯƠNG PHÁP

	Logistic Regression	KNN	SVM
SOBEL	Accuracy:33.59 %	Accuracy:37.91 %	Accuracy:46.45 %
HOG	Accuracy:44.76 %	Accuracy:52.28 %	Accuracy:56.99 %

## 5 KẾT QUẢ THỰC NGHIỆM TRÊN TẬP TEST



Predict: Angry



Predict: Happy

## 6 KẾT LUẬN

- Từ accuracy, ta thấy phương pháp SVM tốt nhất trong 3 phương pháp dùng cho bài toán nhận diện cảm xúc khuôn mặt.
- Từ accuracy, ta thấy phương pháp trích xuất đặc trưng HOG tốt hơn phương pháp Sobel dùng cho bài toán nhận diện cảm xúc khuôn mặt.



## 7 TÀI LIỆU THAM KHẢO

- [1] Computer Vision - Wikipedia
- [2] Logistic Regression 1
- [3] Logistic Regression 2
- [4] k-Nearest Neighbors
- [5] Support Vector Machine 1
- [6] Support Vector Machine 2
- [7] V.Vapnik - The Nature Of Statistical Learning
- [8] Histogram of Oriented Gradients
- [9] DATASET