

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

🙠🙟🕮🙝🙢

****

**BÁO CÁO CUỐI KỲ**

**MÔN HỌC: ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

|  |  |
| --- | --- |
| **GVHD**: | **TS. Trần Nhật Quang** |
| **SVTH:** | **Hoàng Công Mạnh – 21110839**  **Đoàn Huỳnh Trường Thịnh - 21110309** |
| **Mã lớp học:** | **PROJ215879\_23\_1\_12CLC** |

*TP. Hồ Chí Minh, tháng 12 năm 2023*

**DANH SÁCH THÀNH VIÊN THAM GIA ĐỒ ÁN**

***HỌC KỲ I, NĂM HỌC 2023 – 2024***

***\*\*\*\*\****

***Đề tài:***  Xây dựng ứng dụng điểm danh sử dụng nhận diện khuôn mặt

*----------*

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **HỌ VÀ TÊN THÀNH VIÊN** | **MÃ SỐ SINH VIÊN** | **TỶ LỆ THAM GIA** |
| 1 | Đoàn Huỳnh Trường Thịnh | 21110309 | 100% |
| 2 | Hoàng Công Mạnh | 21110839 | 0% |

MỤC LỤC

[CHƯƠNG 1: TÌM HIỂU VỀ ĐẠO VĂN 1](#_Toc153502887)

[1. Khái niệm đạo văn 1](#_Toc153502888)

[2. Phòng tránh đạo văn 1](#_Toc153502889)

[2.1. Những việc nên làm 1](#_Toc153502890)

[2.2. Những việc không nên làm 1](#_Toc153502891)

[3. Lời cam kết 2](#_Toc153502892)

[CHƯƠNG 2: PHẦN LẬP TRÌNH 4](#_Toc153502893)

[1. Đặc tả bài toán: 4](#_Toc153502894)

[2. Thiết kế: 4](#_Toc153502895)

[2.1. Giao diện 4](#_Toc153502896)

[2.1.1. Các thư viện sử dụng: 4](#_Toc153502897)

[2.1.2. Mô tả lớp 5](#_Toc153502898)

[2.2. Thuật toán 6](#_Toc153502899)

[2.2.1. Mô tả phương thức dùng chung 6](#_Toc153502900)

[2.3. Phát hiện khuôn mặt (get\_face.py) 6](#_Toc153502901)

[2.3.1. Mô tả phương thức 6](#_Toc153502902)

[2.4. Xử lý ảnh lưu trữ (Training.py) 7](#_Toc153502903)

[2.4.1. Mô tả lớp 7](#_Toc153502904)

[2.4.2. Mô tả phương thức 7](#_Toc153502905)

[2.5. Nhận diện khuôn mặt (algorithm.dfs\_algorithm.py) 9](#_Toc153502906)

[2.5.1. Mô tả phương thức 9](#_Toc153502907)

[2.6. Trang chủ đồ án Streamlit(FinalCNTT.py) 9](#_Toc153502908)

[2.6.1. Mô tả phương thức 9](#_Toc153502909)

[3. Kiểm thử 10](#_Toc153502910)

[4. Kết luận 11](#_Toc153502911)

[4.1. Ưu điểm 11](#_Toc153502912)

[4.2. Hạn chế 11](#_Toc153502913)

[4.3. Ý tưởng phát triển 11](#_Toc153502914)

[TÀI LIỆU THAM KHẢO 12](#_Toc153502915)

# CHƯƠNG 1: TÌM HIỂU VỀ ĐẠO VĂN

## Khái niệm đạo văn

Đạo văn là chiếm hữu lời nói, suy nghĩ, ý tưởng, văn bản, hay cách diễn đạt của người khác và xem như của mình mà không có sự cho phép của chính chủ. Đạo văn có thể là cố ý hoặc vô ý, nhưng chung quy đều là hành vi thiếu trung thực và vi phạm đạo đức, đặc biệt là trong lĩnh vực nghiên cứu và học thuật. Đạo văn có thể dẫn đến nhiều hậu quả như bị đánh rớt điểm số, bị phạt tiền hay thậm chí cấu thành hành vi vi phạm bản quyền. Điều này làm giảm sút uy tín của bản thân, ảnh hưởng đến công việc và học tập.[1]

## Phòng tránh đạo văn

### Những việc nên làm

Để phòng tránh đạo văn, ta nên làm những việc sau:

* Trích dẫn nguồn khi tham khảo tài liệu của người khác.
* Thực hiện việc trích dẫn nguồn tài liệu tham khảo theo đúng chuẩn yêu cầu.
* Viết lại tài liệu đã tham khảo theo cách hiểu của bản thân và vẫn trích dẫn nguồn tài liệu tham khảo đó.

### Những việc không nên làm

Để phòng tránh đạo văn, ta không nên làm những việc sau:

* Sử dụng tài liệu, ý tưởng của người khác mà không nêu tác giả, trích dẫn nguồn của tài liệu đó.
* Trích dẫn sai nguồn, trích dẫn các nguồn không phải là tư liệu gốc.
* Thay đổi một vài từ trong tài liệu để biến thành tác phẩm của bản thân và không trích dẫn nguồn.

## Lời cam kết

Chúng em xin cam đoan dự án này do các thành viên trong nhóm thực hiện. Chúng em không sao chép, sử dụng bất kỳ tài liệu, mã nguồn… của người khác mà không ghi nguồn. Chúng em xin chịu hoàn toàn trách nhiệm nếu vi phạm đạo văn.

Đoàn Huỳnh Trường Thịnh

Hoàng Công Mạnh

# CHƯƠNG 2: PHẦN LẬP TRÌNH

## Đặc tả bài toán:

Dựa vào những kiến thức của các môn học, nhóm em làm đề tài về bài toán về sử dụng nhận diện khuôn mặt để điểm danh lớp học và xây dựng trên Streamlit:

* Xây dựng ứng dụng trên Streamlit
* Xây dựng chức năng nhận diện khuôn mặt
* Lập trình nhận diện khuôn mặt .

Cách sử dụng:

* B1: Chạy get\_face.py để có thể test camera để phát hiện khuôn mặt.
* B2: Chạy Training.py để có thể đọc dữ liệu hình ảnh có sẵn trong thư mục.
* B3: Chạy predict.py để xem hình ảnh nhận diện
* B4: Chạy FinalCNTT.py để xem đồ án trên streamlit.

## Thiết kế:

### Giao diện

### Các thư viện sử dụng:

* **Pygame:** Streamlit là một công cụ mã nguồn mở được sử dụng để tạo giao diện người dùng (GUI) tương tác cho các ứng dụng dữ liệu. Nó cho phép người dùng sử dụng các ngôn ngữ lập trình như Python để xây dựng các ứng dụng web trực tiếp và trực quan mà không cần kiến thức phức tạp.
* ***Cách cài đặt:*** pip install streamlit
* **Cv2:** cv2 (OpenCV) là một thư viện mã nguồn mở và miễn phí được sử dụng rộng rãi trong lĩnh vực xử lý ảnh và thị giác máy tính.
* **Cách cài đặt :** pip install opencv-python
* **Numpy :** NumPy là một thư viện mã nguồn mở cho Python, được sử dụng rộng rãi trong tính toán khoa học và xử lý số liệu.
* **Cách cài đặt :** pip install numpy
* **Joblib :** Joblib là một thư viện mã nguồn mở trong Python, được sử dụng để lưu trữ và tải các đối tượng Python
* **Cách cài đặt :** pip install joblib
* **Ngoài ra còn một số thư viện khác như :** os.path, maplotlib, sklearn, time

### Mô tả lớp

|  |  |  |
| --- | --- | --- |
| STT | Tên lớp | Mục đích |
| 1 | IdentityMetadata | Mực đích là giúp xác định đường dẫn đầy đủ của hình ảnh dựa trên các thông tin cơ bản như thư mục gốc, tên danh tính và tên tệp hình ảnh. |

### Thuật toán

### Mô tả phương thức dùng chung

|  |  |  |
| --- | --- | --- |
| TT | Phương thức | Mục đích |
| 1 | def visualize(input, faces, fps, thickness=2) | Mục đích của hàm này là để vẽ hộp giới hạn và các điểm đặc trưng của các khuôn mặt đã được nhận diện trên ảnh đầu vào và hỗ trợ quá trình trực quan hóa kết quả của hệ thống nhận diện khuôn mặt, giúp người dùng hiểu rõ hơn về vị trí và các điểm đặc trưng của khuôn mặt được nhận diện, cũng như tốc độ khung hình của hệ thống. |

### Phát hiện khuôn mặt (get\_face.py)

### Mô tả phương thức

|  |  |  |
| --- | --- | --- |
| TT | Phương thức | Mục đích |
| 1 | **def str2bool(v)**  Input: v  Output: True , False | Mục đích chuyển đổi một chuỗi đầu vào thành một giá trị boolean. Có tham số:  ‘v’ : đại diện cho giá trị cần chuyển đổi thành boolean. |
| 2 | **parser = argparse.ArgumentParser()** | Mục đích của đoạn code này là định nghĩa các tham số và cách sử dụng của chương trình thông qua các argument được cung cấp khi chạy từ dòng lệnh. |
| 3 | **detector= cv.FaceDetectorYN.create(** | Thiết lập kích thước đầu vào cho detector dựa trên kích thước của khung hình được nhận từ camera. |
| 4 | **visualize(frame,faces, tm.getFPS())** | Mục đích để hiển thị thông tin kết quả của việc nhận diện khuôn mặt lên khung hình gốc (frame). |

### Xử lý ảnh lưu trữ (Training.py)

### Mô tả lớp

* Sử dụng các lớp chung của thuật toán.

### Mô tả phương thức

|  |  |  |
| --- | --- | --- |
| TT | Phương thức | Mục đích |
| 1 | **def \_\_init\_\_(self, base, name, file)**  Input : base ,self, name, file  Output : None | Mục đích chính của hàm khởi tạo là khởi tạo các thuộc tính cho một đối tượng. Có các tham số:  ‘base’: Là thư mục cơ sở (base directory) cho tập dữ liệu.  ‘name’: Là tên của đối tượng trong tập dữ liệu.  ‘file’: Là tên tệp ảnh hoặc đường dẫn đến tệp ảnh liên quan đến đối tượng trong tập dữ liệu. |
| 2 | **def \_\_repr\_\_(self)** | Mục đích là sử dụng để xác định cách biểu diễn chuỗi của một đối tượng khi được gọi hàm repr() hoặc khi đối tượng được sử dụng trong các tình huống yêu cầu biểu diễn chuỗi của nó. |
| 3 | **def image\_path(self)** | Mục đích chính của phương thức này là xây dựng đường dẫn đến tệp ảnh dựa trên các thông tin được cung cấp bởi các thuộc tính của đối tượng. |
| 4 | **def load\_metadata(path)** | Mục đích đọc dữ liệu từ thư mục được chỉ định và tạo metadata cho các tệp ảnh có định dạng .jpg, .jpeg, hoặc .bmp trong các thư mục con. |
| 5 | **def load\_image(path)** | Mục đích là đọc một tệp ảnh từ đường dẫn được cung cấp bằng thư viện OpenCV (cv2.imread()) và sau đó chuyển đổi không gian màu từ BGR (Blue-Green-Red) sang RGB (Red-Green-Blue) |
| 6 | **def distance(emb1, emb2)** | Mục đích là tính toán khoảng cách giữa hai vectơ nhúng |
| 7 | **def show\_pair(idx1, idx2)** | Mục đích chính của hàm này là hiển thị cặp hình ảnh được chọn từ metadata tương ứng với chỉ số idx1 và idx2, và cung cấp thông tin về khoảng cách giữa hai hình ảnh này bằng cách tính toán khoảng cách giữa các vectơ nhúng (embedding) tương ứng với hai chỉ số này. |
| 8 |  |  |

### Nhận diện khuôn mặt (algorithm.dfs\_algorithm.py)

### Mô tả phương thức

|  |  |  |
| --- | --- | --- |
| TT | Phương thức | Mục đích |
| 1 | **svc = joblib.load()** | Mục đích để tải mô hình đã được lưu trữ trước đó từ file svc.pkl |
| 2 | **mydict =[]** | Tạo một danh sách (mydict) chứa các chuỗi là tên của các người được xác định hoặc phân loại. |
| 3 | **detector=cv.FaceDetectorYN.create()** | Mục đích tạo mô hình nhận diện khuôn mặt |
| 4 | **recognizer=cv.FaceRecognizerSF.create()** | Mục đích tạo mô hình nhận diện khuôn mặt đặc trưng |

### Trang chủ đồ án Streamlit(FinalCNTT.py)

### Mô tả phương thức

|  |  |  |
| --- | --- | --- |
| TT | Phương thức | Mục đích |
| 1 | **def add\_bg\_from\_url()** | Mục đích là sử dụng để thêm hình nền (background) từ một URL vào một phần tử giao diện người dùng trong một ứng dụng web |
| 2 | **st.sidebar ()** | Mục đích là để thêm các thành phần giao diện người dùng (widgets) vào thanh điều hướng bên (sidebar) của ứng dụng. |
| 3 | **st.markdown()** | Mục đích là để sử dụng để hiển thị và định dạng văn bản theo cú pháp Markdown trong ứng dụng của bạn. |
| 4 | **st.title()** | Mục đích là để sử dụng để hiển thị tiêu đề lớn và nổi bật trong giao diện ứng dụng của bạn. |
| 5 | **st.subheader()** | Mục đích là để sử dụng để hiển thị tiêu đề phụ, có kích thước nhỏ hơn so với tiêu đề chính. |

## Kiểm thử

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Giải thích |
| 1 | **Tình huống 1:**  Dữ liệu vào: Hình ảnh đã được training  Kết quả dự kiến: Nhận diện đúng khuôn mặt | Kiểm tra xem ứng dụng phát hiện khuôn mặt có phát hiện chính xác khuôn mặt đã được training | Predict sẽ nhận diện đúng khuôn mặt , và nhận diện được khuôn mặt đặc trưng nhờ những model có sẵn đã được người dùng training trước. |
| 2 | **Tình huống 2:**  Dữ liệu vào: Hình ảnh đã được training  Kết quả dự kiến: Không nhận diện được khuôn mặt | Kiểm tra xem ứng dụng phát hiện khuôn mặt có phát hiện chính xác khuôn mặt đã được training | Khi Predict trên Streamlit sẽ không nhận diện đúng khuôn mặt , và không nhận diện được khuôn mặt đặc trưng nhờ những model có sẵn đã được người dùng training trước. Khi đó sẽ kết thúc nhận diện bằng dòng chữ “No faces detected! Stopping the program.” |

## Kết luận

### Ưu điểm

* Ứng dụng dễ sử dụng
* Các thuật toán được lập trình đúng với mục tiêu đề ra và cho kết quả đúng
* Ứng dụng đầy đủ các chức năng để thực hiện các mục tiêu đề ra như: cho phép truy cậm camera , cho phép đọc dữ liệu có sẵn từ người dùng , có thể training và cho ra kết quả đúng

### Hạn chế

* Vì trong khoảng thời gian ngắn nên ứng dụng của chúng em vẫn còn thiếu các chức năng để giúp bài toán được mở rộng hơn.

### Ý tưởng phát triển

* Cho phép người dùng sử dụng để điểm danh những mô hình lớn hơn ngoài quản lí lớp học.
* Cải tiến nhận diện khuôn mặt chính xác và tốn ít thời gian hơn.

# TÀI LIỆU THAM KHẢO

**[1]** *Hệ thống điểm danh bằng nhận diện khuôn mặt qua Camera - Python-OpenCV-Tkinter-MySQL, from https://www.youtube.com/watch?v=4yEv9ome5T8&t=153s*

**[2]** *Hướng dẫn cài đặt Anaconda và Jupyter Notebook , from https://unitrain.edu.vn/huong-dan-cai-dat-anaconda-va-jupyter-notebook/*

**[3]***Face-Recognition-with-OpenCV-Python-DL* GitHub : https://github.com/huytranvan2010/Face-Recognition-with-OpenCV-Python-DL

**[4]** face\_recognition https://github.com/thangdc/face\_recognition