BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**  **



**NIÊN LUẬN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Đề tài**

**XÂY DỰNG ỨNG DỤNG TÁCH NỀN VÀ GHÉP VIDEO BẰNG MEDIAPIPE**

**Sinh viên thực hiện: Huỳnh Nhật Tâm**

**Mã số sinh viên: B1812375**

**Khóa: K44**

Cần Thơ, 4/2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**  **



**NIÊN LUẬN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Đề tài**

**XÂY DỰNG ỨNG DỤNG TÁCH NỀN VÀ GHÉP VIDEO BẰNG MEDIAPIPE**

**Giảng viên hướng dẫn: Sinh viên thực hiện:**

**Huỳnh Nhật Tâm**

**Th.S. Phạm Nguyên Hoàng**

**Mã số sinh viên: B1812375**

**Khóa: K44**

Cần Thơ, 4/2022

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

**LỜI CẢM ƠN**

Trong quá trình thực hiện đề tài em đã gặp khá nhiều khó khăn do thiếu kiến thức và chưa làm quen được với quá trình thực hiện tạo ra một ứng dụng. Nếu không có sự giúp đỡ của thầy và các bạn cũng như là các youtuber làm về chia sẻ kiến thức lập trình và các anh chị/ thầy cô giảng dạy trên website Udemy, em sẽ không hoàn thành bài niên luận này một cách hoàn chỉnh.

Đầu tiên, em xin trân trọng gửi lời cảm ơn sâu sắc đến thầy Phạm Nguyên Hoàng là người đã tận tình trực tiếp hướng dẫn và giúp đỡ em trong suốt quá trình hoàn thành bài niên luận ngành. Với sự chỉ dẫn và dạy bảo của thầy, em đã có những định hướng tốt trong quá trình triển khai và thực hiện các yêu cầu của bài niên luận.

Em xin gửi lời cảm ơn chân thành đến quý thầy cô Trường Đại học Cần Thơ đặc biệt là thầy cô trong Khoa CNTT & TT, những người đã truyền đạt cho em những kiến thức và những bài học quý báu trong suốt thời gian qua.

Em xin cảm ơn gia đình đã quan tâm, động viên và luôn tạo cho em những điều kiện tốt nhất để học tập và thực hiện đề tài niên luận ngành.

Ngoài ra, em cũng xin gửi lời cảm ơn đến tất cả bạn bè đã luôn gắn bó cùng em trong quá trình học tập, giúp đỡ trong suốt thời gian qua cũng như cảm ơn đến tất cả những anh chị/ thầy cô giảng dạy trên nền tảng trực tuyến để bản thân em ngày càng phấn đấu, nổ lực hơn trong việc học tập cũng như trong quá trình thực hiện đề tài niên luận ngành này.

Tuy có nhiều cố gắng trong quà trình thực hiện niên luận nhưng vẫn không thể tránh khỏi những sai sót. Em rất mong nhận sự đóng góp ý kiến quý báu của quý thầy cô và các bạn để bài niên luận của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Cần Thơ, ngày 25 tháng 4 năm 2022

Người viết

Huỳnh Nhật Tâm

**MỤC LỤC**

[DANH MỤC HÌNH 3](#_Toc103523810)

[DANH MỤC BẢNG 4](#_Toc103523811)

[ABSTRACT 5](#_Toc103523812)

[TÓM TẮT 6](#_Toc103523813)

[PHẦN GIỚI THIỆU 7](#_Toc103523814)

[1. Đặt vấn đề 7](#_Toc103523815)

[2. Lịch sử giải quyết vấn đề 7](#_Toc103523816)

[2.1. Giải thuật GrabCut 7](#_Toc103523817)

[2.2. Deep Image Matting 8](#_Toc103523818)

[2.3. MODNet 10](#_Toc103523819)

[3. Đối tượng và phạm vi nghiên cứu 10](#_Toc103523820)

[4. Phương pháp nghiên cứu 10](#_Toc103523821)

[5. Kết quả đạt được 11](#_Toc103523822)

[6. Bố cục luận văn 11](#_Toc103523823)

[PHẦN NỘI DUNG 12](#_Toc103523824)

[CHƯƠNG 1 12](#_Toc103523825)

[MÔ TẢ BÀI TOÁN 12](#_Toc103523826)

[1. Mô tả chi tiết bài toán 12](#_Toc103523827)

[2. Vấn đề và giải pháp liên quan đến bài toán 12](#_Toc103523828)

[2.1. Vấn đề liên quan đến bài toán 12](#_Toc103523829)

[2.2. Giải pháp liên quan đến bài toán 12](#_Toc103523830)

[3. Giới thiệu về ngôn ngữ lập trình, các thư viện 14](#_Toc103523831)

[3.1. Ngôn ngữ lập trình Python 14](#_Toc103523832)

[3.2. OpenCV 15](#_Toc103523833)

[3.3. PyQt5 16](#_Toc103523834)

[CHƯƠNG 2 17](#_Toc103523835)

[THIẾT KẾ VÀ CÀI ĐẶT 17](#_Toc103523836)

[1. Thiết kế ứng dụng 17](#_Toc103523837)

[2. Cài đặt ứng dụng 18](#_Toc103523838)

[2.1. Hàm thay đổi kích thước frame: resize() 18](#_Toc103523839)

[2.2. Hàm thêm pixel vào frame: add\_pixel() 18](#_Toc103523840)

[2.3. Các class được sử dụng trong ứng dụng 19](#_Toc103523841)

[CHƯƠNG 3 21](#_Toc103523842)

[KẾT QUẢ THỰC NGHIỆM 21](#_Toc103523843)

[1. Giao diện ứng dụng 21](#_Toc103523844)

[2. Kết quả thực nghiệm 21](#_Toc103523845)

[PHẦN KẾT LUẬN 25](#_Toc103523846)

[1. Kết quả đạt được 25](#_Toc103523847)

[2. Hạn chế 25](#_Toc103523848)

[3. Thuận lợi 25](#_Toc103523849)

[4. Khó khăn 25](#_Toc103523850)

[5. Hướng phát triển 25](#_Toc103523851)

[TÀI LIỆU THAM KHẢO 26](#_Toc103523852)

# DANH MỤC HÌNH

[Hình 1. Giải thuật GrabCut 8](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522778)

[Hình 2. Input và Output của giải thuật GrabCut 8](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522779)

[Hình 3. Hình ảnh và trimap của nó 9](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522780)

[Hình 4. Tiến trình trimap. Phân đoạn đối tượng(trái), xói mòn phân đoạn(giữa), thêm vùng không xác định giãn nở(phải). 9](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522781)

[Hình 5. Quy trình train MODNet 10](#_Toc103522782)

[Hình 6. Các giải pháp được cung cấp và tính khả dụng trên các nền tảng. 13](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522783)

[Hình 7. Danh sách 33 điểm mốc (Landmarks) và các kết nối giữa chúng. 14](file:///C:\Users\hiimh\Downloads\HuynhNhatTam_B1812375_NienLuanNganh.docx#_Toc103522784)

[Hình 8. Lưu đồ giải thuật về cách hoạt động của ứng dụng. 17](#_Toc103522785)

[Hình 9. Lưu đồ giải thuật cho hàm resize. 18](#_Toc103522786)

[Hình 10. Lưu đồ giải thuật cho hàm add\_pixel. 19](#_Toc103522787)

[Hình 11. Giao diện ứng dụng. 21](#_Toc103522788)

[Hình 12. Hình 12a: Cô gái áo đen, hình 12b: Cô gái áo trắng. 22](#_Toc103522789)

[Hình 13a. Khung hình được cắt ra từ video kết quả của trường hợp 1. 22](#_Toc103522790)

[Hình 13b. Khung hình được cắt ra từ video kết quả của trường hợp 1. 22](#_Toc103522790)

[Hình 14. Hình 14a: Cô gái áo đen, hình 14b: Hai cô gái đeo khẩu trang. 23](#_Toc103522791)

[Hình 15. Khung hình được cắt ra từ video kết quả của trường hợp 2. 24](#_Toc103522792)

# DANH MỤC BẢNG

[Bảng 1. Bảng các biến khởi tạo trong class Capture. 19](#_Toc103522772)

[Bảng 2. Bảng các nút chức năng trong class Window. 20](#_Toc103522773)

[Bảng 3. Bảng thông tin video kiểm thử cho trường hợp 1. 22](#_Toc103522774)

[Bảng 4. Bảng thông tin video kiểm thử cho trường hợp 2. 23](#_Toc103522775)

# ABSTRACT

Currently, information technology has been developing non-stop along with the increasing material and spiritual needs of people, of which AI (Artificial Intelligence) is a technological breakthrough, helping to promote the "evolution" of humanity. In addition to inventions, robot products, virtual assistants, etc., inventions and products for object recognition in images and videos are also one of the highly influential applications of AI, such as identifying people wearing masks or not to contribute to reducing damage caused by epidemics, or counting the number of people in an area... From that, I came up with the idea to develop an application that recognizes humanoid form from which the background can be extracted and the video containing the recognized object can be assembled. In the process of implementing the application, I applied the knowledge I learned about programming languages Python, AI, Computer Vision and consulted Mediapipe solution to perform the recognition process and some other libraries to create simple application interface. After completing the installation and testing, the results are quite good when it is possible to combine two videos that have been removed background in a fixed position on the other background.

# TÓM TẮT

Hiện nay, công nghệ thông tin đã và đang phát triển không ngừng nghỉ cùng với nhu cầu về vật chất lẫn tinh thần ngày một tăng của con người, trong đó AI (Artificial Intelligence) hay còn gọi là Trí Tuệ Nhân Tạo là một bước đột phá về công nghệ, giúp thúc đẩy quá trình “tiến hóa” của nhân loại. Ngoài những phát minh, những sản phẩm về robot, trợ lý ảo...thì các phát minh, các sản phẩm về nhận dạng đối tượng trong hình ảnh, video cũng là một trong những ứng dụng có tầm ảnh hưởng lớn của trí tuệ nhân tạo, ví dụ như nhận dạng người có đeo khẩu trang hay không để góp phần giảm thiểu thiệt hại do dịch bệnh, hay đếm số lượng người trong một khu vực...Từ đó em lên ý tưởng phát triển một ứng dụng nhận dạng dáng người từ đó có thể tách nền và ghép video chứa đối tượng đã được nhận dạng đó. Trong quá trình thực hiện ứng dụng em đã áp dụng các kiến thức đã học về ngôn ngữ lập trình Python, AI, Computer Vision (Thị Giác Máy Tính) và tham khảo giải pháp Mediapipe để thực hiện quá trình nhận dạng cùng một số thư viện khác để tạo ra giao diện ứng dụng đơn giản. Sau khi đã hoàn tất việc cài đặt và kiểm thử, kết quả cho ra khá tốt khi có thể ghép hai video đã được tách nền ở một vị trí cố định trên cùng một hình nền khác do người dùng chọn tùy ý.

# PHẦN GIỚI THIỆU

### Đặt vấn đề

Để có thể ghép những hình ảnh, video của mình hay những hình ảnh, video thú vị vào những nền khác đẹp mắt, con người trước đó phải chụp hình hay quay video trên nền phông xanh (green screen) mới có thể chèn hình hay video đó vào nền có sẵn. Để thực hiện điều này, người dùng phải chi ra rất nhiều tiền và thời gian để tạo ra một bức hình hay một chiếc phim như ý. Nhưng với sự phát triển của AI (Artificial Intelligence) hay còn gọi là Trí Tuệ Nhân Tạo, Thị Giác Máy Tính, các nhà phát triển đã cho ra mắt hàng ngàn ứng dụng có thể tách nền và ghép hình ảnh hay video vào một phông nền khác giúp tiết kiệm rất nhiều chi phí và thời gian, kéo theo các trào lưu thay đổi phông nền đang rất thịnh hành ngày nay trên các trang mạng xã hội như Facebook, Tiktok…

Để có thể bắt kịp xu hướng này, em quyết định áp dụng các kiến thức đã học của ngành Khoa học Máy tính, về Trí Tuệ Nhân Tạo, Thị Giác Máy Tính… để xây dựng một ứng dụng tách nền và ghép video tự động bằng Mediapipe ưu tiên về sự tiện lợi và tốc độ xử lý, người dùng chỉ cần chọn hai video và nền là có thể sở hữu ngay một chiếc video được ghép từ hai video đã chọn vào chung một nền.

### Lịch sử giải quyết vấn đề

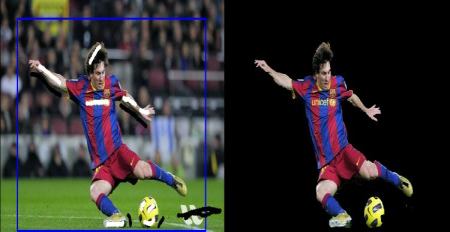
#### 2.1. Giải thuật GrabCut

Thuật toán GrabCut được thiết kế bởi Carsten Rother, Vladimir Kolmogorov & Andrew Blake từ Microsoft Research Cambridge. Trong bài báo của họ, “GrabCut”: trích xuất foreground bằng cách sử dụng các đường cắt biểu đồ lặp đi lặp lại.

Ban đầu, người dùng vẽ một hình chữ nhật xung quanh vùng foreground (vùng foreground nằm hoàn toàn bên trong hình chữ nhật). Sau đó, thuật toán phân đoạn sẽ được lặp đi lặp lại để có được kết quả tốt nhất. Nhưng trong một số trường hợp, phân đoạn sẽ không ổn, chẳng hạn như nó có thể đã đánh dấu một số vùng foreground làm background và ngược lại. Trong trường hợp đó, người dùng cần phải thực hiện các thao tác chỉnh sửa tốt. Về cơ bản, thuật toán này cần phải đánh dấu đúng vị trí của vật thể, đối tượng thì mới cho ra kết quả tốt. Điều này khá bất tiện và mất thời gian.



Hình 1. Giải thuật GrabCut



Hình 2. Input và Output của giải thuật GrabCut

#### 2.2. Deep Image Matting

Deep Image Matting, do Adobe Research thực hiện vào năm 2017. Mô hình này hiện được sử dụng trong hầu hết các trang web sử dụng để tự động xóa nền khỏi ảnh. Kỹ thuật này cần hai đầu vào: một hình ảnh và trimap của nó. Trimap về cơ bản là một đại diện của hình ảnh ở ba cấp độ: background, foreground và một vùng mà các pixel được coi là hỗn hợp của foreground và background.



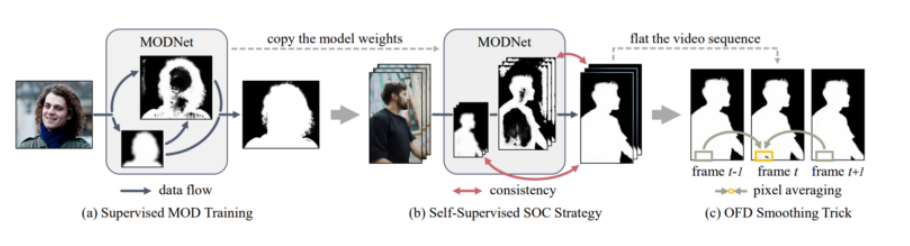
Hình 3. Hình ảnh và trimap của nó

Để xóa phông nền thành công bằng kỹ thuật Deep Image Matting, cần một mạng lưới mạnh mẽ có thể khoanh vùng đối tượng đó một cách chính xác. Sau đó, tạo ra một phân đoạn trong đó các pixel tương đương với đối tượng được đặt thành 1 và phần còn lại của hình ảnh được đặt thành 0. Tiếp theo, sử dụng các phép biến đổi thị giác máy tính cơ bản để tạo một trimap từ phân đoạn này. Bằng cách giảm kích thước của đối tượng được phân đoạn để lại một chút không gian cho vùng không xác định bằng cách làm xói mòn nó, loại bỏ lặp đi lặp lại một số pixel ở đường viền của đối tượng. Sau đó, tạo một vùng không xác định, bằng cách giãn đối tượng, thêm các pixel xung quanh đường viền.



Hình 4. Tiến trình trimap. Phân đoạn đối tượng(trái), xói mòn phân đoạn(giữa), thêm vùng không xác định giãn nở(phải).

#### 2.3. MODNet



Hình 5. Quy trình train MODNet

a) Train MODNet trên tập dữ liệu được gán nhãn để học.

b) Để thích ứng với dữ liệu trong thế giới thực, điều chỉnh MODNet trên dữ liệu không được gán nhãn bằng cách sử dụng tính nhất quán giữa các sub-objective.

c) Thủ thuật OFD giúp làm mượt các mặt đệm alpha được dự đoán của chuỗi video.

MODNet là mạng phân rã vật kính làm mờ trọng lượng nhẹ (MODNet), có thể xử lý thảm dọc từ một hình ảnh đầu vào duy nhất trong thời gian thực.

Đặc điểm của MODNet:

* Nhanh hơn nhiều so với các phương pháp chiếu cùng thời và chạy ở tốc độ 63 khung hình / giây.
* Đạt được kết quả đáng kể trong ảnh và video.
* Dễ dàng được huấn luyện theo quy trình end-to-end. Việc nhận dạng người và xóa background chứa người đó không phải là một công việc dễ dàng. Nhiều kỹ thuật đang sử dụng các thuật toán thị giác máy tính cho nhiệm vụ này một cách nhanh chóng nhưng không chính xác.
* MODNet đơn giản, nhanh chóng và hiệu quả, tránh sử dụng green screen (màn hình xành).

### Đối tượng và phạm vi nghiên cứu

* Đối tượng nghiên cứu: các đoạn video có sự hoạt động của một đối tượng cần tách khỏi phông nền.
* Phạm vi nghiên cứu: ghép hai đoạn video vào chung một background (nền) tùy ý với mỗi video chỉ có một đối tượng làm trọng tâm.

### Phương pháp nghiên cứu

* Về lý thuyết:
  + Học cách viết ngôn ngữ lập trình Python trên một số trang, kênh dạy lập trình trực tuyến như Youtube, Udemy.
  + Nghiên cứu và tham khảo các bài báo khoa học liên quan đến đề tài trên mạng và từ thư viện của Khoa Công nghệ Thông tin và Truyền thông, trường Đại học Cần thơ.
  + Tìm hiểu cách xây dựng giao diện ứng dụng với thư viện PyQt5, OpenCV.
* Về thực tiễn:
  + Tiến hành thực hiện vẽ lưu đồ thuật toán cho các hàm cần thiết, kiểm tra lại lưu đồ tiến hành trao đổi với giáo viên để hoàn thiện.
  + Áp dụng các kiến thức đã học và nghiên cứu về các ngôn ngữ lập trình Python cùng với thư viện PyQt5 để tiến hành xây dựng giao diện ứng dụng, sau đó lập trình các chức năng đã đề ra từ mục tiêu ban đầu.
  + Kiểm tra lại các quá trình lập trình và thử nghiệm các chức năng để ứng dụng có thể vận hành trơn tru. Trao đổi với giáo viên về các chức năng và nhận đánh giá sau đó cập nhật ứng dụng để được tối ưu tốt hơn.

### Kết quả đạt được

* Xây dựng thành công ứng dụng tách nền và ghép video bằng Mediapipe.
* Ứng dụng có các chức năng của một ứng dụng ghép video cơ bản đáp ứng đầy đủ so với mục tiêu đề ra ban đầu: chọn video, chọn background, phát video, dừng video, lưu video và thoát ứng dụng.

### Bố cục luận văn

**Phần giới thiệu**

Giới thiệu tổng quát về đề tài.

**Phần nội dung**

**Chương 1**: Mô tả bài toán, vấn đề và giải pháp.

**Chương 2**: Thiết kế và cài đặt ứng dụng.

**Chương 3**: Kết quả đạt được.

**Phần kết luận**

Trình bày ưu điểm, khuyết điểm, và hướng phát triển hệ thống, quá trình tích lũy kinh nghiệm cho bản thân

# PHẦN NỘI DUNG

## CHƯƠNG 1

## MÔ TẢ BÀI TOÁN

### Mô tả chi tiết bài toán

Ứng dụng tách nền và ghép video của đề tài này sẽ nhận vào hai video, đọc từng frame của hai video đó sau đó thay đổi kích thước và thêm các pixel trắng vào frame sao cho kích thước của frame bằng với kích thước của background đưa vào. Tiếp đến, sau khi đã hoàn tất quá trình chuẩn bị frame, các frame sẽ lần lượt được đưa vào tiến trình Mediapipe để thực hiện nhận diện dáng người và tách frame thành hai phần: đối tượng và background. Cuối cùng, ứng dụng sẽ thay đổi background đã được tách ra của frame thành background đưa vào. Vì ghép hai video nên sẽ có chênh lệch về mặt thời lượng video, ứng dụng sẽ cho phép video ngắn sẽ kết thúc sớm, video dài hơn sẽ tiếp tục phát cho đến hết.

### Vấn đề và giải pháp liên quan đến bài toán

#### 2.1. Vấn đề liên quan đến bài toán

Các giải pháp đã nêu ở phần lịch sử giải quyết vấn đề và một số giải pháp khác có chung một vài đặc điểm đó là cài đặt phức tạp, độ chính xác không quá cao. Với nhu cầu hiện nay thì việc tách nền cần đến độ chính xác cao, xử lý nhanh và trong thời gian thực để mang đến trải nghiệm tốt nhất cho người dùng. Vì lý do đó Google tập trung phát triển và công khai các giải pháp liên quan đến nhận diện sử dụng Trí Tuệ Nhân Tạo từ đó có thể tách nền ra khỏi vật thể.

**2.2. Giải pháp liên quan đến bài toán**

**Giới thiệu Mediapipe**

Mediapipe là một tập hợp các giải pháp cho việc xây dựng các ứng dụng, hệ thống máy học (Machine Learning) chuyện xử lý các dạng dữ liệu loại hình ảnh, video, âm thanh… được Google phát triển.

Một số ưu điểm có thể kể đến của giải pháp này bao gồm:

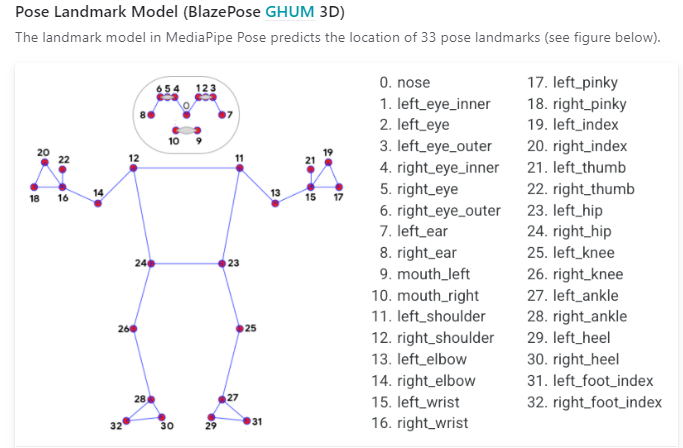
* Cung cấp một giải pháp inference nhanh chóng: Google khẳng định rằng bộ công cụ này có thể chạy ổn định trên hầu hết các cấu hình phần cứng thông dụng.
* Dễ dàng cài đặt và triển khai: Việc cài đặt cực kỳ dễ dàng và tiện lợi, có thể triển khai trên nhiều nền tảng khác nhau như Mobile (Android/iOS), Desktop/Cloud, Web và IoT devices.
* Mã nguồn mở và miễn phí: Toàn bộ source code được công khai trên website Mediapipe, người dùng hoàn toàn có thể sử dụng và tùy chỉnh trực tiếp để phù hợp với bài toán của mình.

****

Hình 6. Các giải pháp được cung cấp và tính khả dụng trên các nền tảng.

Hầu hết các bài toán nổi bật trong lĩnh vực Computer Vision - Thị giác máy tính, đều được Google cài đặt trong Mediapipe, trong đề tài niên luận này em tập trung và sử dụng Mediapipe Pose để nhận dạng dáng người.

Mediapipe Pose là một giải pháp Machine Learning để theo dõi, nhận dạng tư thế cơ thể con người có độ trung thực cao, cung cấp 33 điểm mốc (Landmarks) 3D được kết nối tạo nên một khung xương người và mặt nạ phân đoạn nền (Background Segmentation Mask) trên toàn bộ cơ thể từ frame RGB của video. Giải pháp này đạt được hiệu suất thời gian thực trên hầu hết các điện thoại di động, máy tính để bàn / máy tính xách tay, trong Python và thậm chí trên nền web.



Hình 7. Danh sách 33 điểm mốc (Landmarks) và các kết nối giữa chúng.

### 3. Giới thiệu về ngôn ngữ lập trình, các thư viện

#### 3.1. Ngôn ngữ lập trình Python

**Giới thiệu về Python**

Python là ngôn ngữ lập trình hướng đối tượng bậc cao, dùng để phát triển website và nhiều ứng dụng khác nhau. Python được tạo ra bởi Guido van Rossum và được phát triển trong một dự án mã mở (open source).

Với cú pháp cực kì đơn giản và thanh lịch, Python là lựa chọn hoàn hảo cho những ai lần đầu tiên học lập trình. Tuy nhiên, đây cũng là ngôn ngữ nổi tiếng về sự chặt chẽ, nhanh, mạnh, và có mặt ở mọi hệ điều hành.

Trong thực tế, Python được sử dụng trong nhiều ngành nghề khác nhau, như:

* Machine Learning và AI.
* Data Science và Data Visualization: Python được coi là một trong những ngôn ngữ lập trình phổ biến nhất trong lĩnh vực Data Science trên toàn thế giới.
* Phát triển game.
* Phát triển phần mềm.
* Phát triển website.

Python là một ngôn ngữ lập trình thông dịch (interpreted), hướng đối tượng (object-oriented), và là một ngôn ngữ bậc cao (high-level) ngữ nghĩa động (dynamic semantics). Python hỗ trợ các module và gói (packages), khuyến khích chương trình module hóa và tái sử dụng mã. Trình thông dịch Python và thư viện chuẩn mở rộng có sẵn dưới dạng mã nguồn hoặc dạng nhị phân miễn phí cho tất cả các nền tảng chính và có thể được phân phối tự do.

Sau đây là các đặc điểm của Python:

* Ngữ pháp đơn giản, dễ đọc.
* Vừa hướng thủ tục (procedural-oriented), vừa hướng đối tượng (object-oriented)
* Hỗ trợ module và hỗ trợ gói (package)
* Xử lý lỗi bằng ngoại lệ (Exception)
* Kiểu dữ liệu động ở mức cao.
* Có các bộ thư viện chuẩn và các module ngoài, đáp ứng tất cả các nhu cầu lập trình.
* Có khả năng tương tác với các module khác viết trên C/C+ (Hoặc Java cho Jython, hoặc .Net cho IronPython).
* Có thể nhúng vào ứng dụng như một giao tiếp kịch bản (scripting interface).

#### 3.2. OpenCV

**Giới thiệu về OpenCV**

Dự án OpenCV được bắt đầu từ Intel năm 1999 bởi Gary Bradsky. OpenCV viết tắt cho Open Source Computer Vision Library. OpenCV là thư viện nguồn mở hàng đầu cho Computer Vision và Machine Learning, và hiện có thêm tính năng tăng tốc GPU cho các hoạt động theo real-time.Vì thế hiện nay mọi người hầu hết đều sử dụng thư viện OpenCV qua ngôn ngữ lập trình Python để xây dựng các ứng dụng về thị giác máy tính nhờ sự tiện lợi trong việc cài đặt và sửa lỗi.

OpenCV được phát hành theo *giấy phép BSD*, do đó nó miễn phí cho cả học tập và sử dụng với mục đích thương mại. Nó có trên các giao diện C++, C, Python và Java và hỗ trợ Windows, Linux, Mac OS, iOS và Android. OpenCV được thiết kế để hỗ trợ hiệu quả về tính toán và chuyên dùng cho các ứng dụng real-time (thời gian thực). Nếu được viết trên C/C++ tối ưu, thư viện này có thể tận dụng được bộ xử lý đa lõi (multi-core processing).

Một số ứng dụng nổi bật của OpenCV được sử dụng cho đa dạng nhiều mục đích khác nhau bao gồm:

* Kiểm tra và giám sát tự động.
* Robot và xe hơi tự lái.
* Phân tích hình ảnh y học.
* Tìm kiếm và phục hồi hình ảnh/video.
* Phim – cấu trúc 3D từ chuyển động.

**3.3. PyQt5**

**Giới thiệu về PyQt5**

Qt là một Application framework đa nền tảng viết trên ngôn ngữ C++ , được dùng để phát triển các ứng dụng trên desktop, hệ thống nhúng và mobile. Hỗ trợ cho các platform bao gồm: Linux, OS X, Windows, VxWorks, QNX, Android, iOS, BlackBerry, Sailfish OS và một số platform khác. PyQt là Python interface của Qt, kết hợp của ngôn ngữ lập trình Python và thư viện Qt, là một thư viện bao gồm các thành phần giao diện điều khiển (widgets, graphical control elements).

PyQt được phát triển bởi Riverbank Computing Limited. PyQt API bao gồm các module bao gồm số lượng lớn với các classes và functions hỗ trợ cho việc thiết kế ra các giao diện giao tiếp với người dùng của các phần mềm chức năng. Hỗ trợ với Python 2.x và 3.x.

Các class của PyQt5 được chia thành các module, bao gồm:

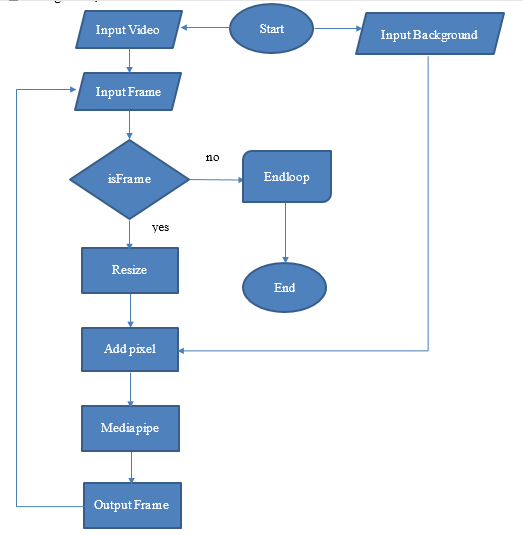
* QtCore: là module bao gồm phần lõi không thuộc chức năng GUI, ví dụ dùng để làm việc với thời gian, file và thư mục, các loại dữ liệu, streams, URLs, mime type, threads hoặc processes.
* QtGui: bao gồm các class dùng cho việc lập trình giao diện (windowing system integration), event handling, 2D graphics, basic imaging, fonts và text.
* QtWidgets: bao gồm các class cho widget, ví dụ: button, hộp thoại, … được sử dụng để tạo nên giao diện người dùng cơ bản nhất.
* QtMultimedia: thư viện cho việc sử dụng âm thanh, hình ảnh, camera…
* QtBluetooth: bao gồm các class giúp tìm kiếm và kết nối với các thiết bị có giao tiếp với phần mềm.
* QtNetwork: bao gồm các class dùng cho việc lập trình mạng, hỗ trợ lập trình TCP/IP và UDP client, server hỗ trợ việc lập trình mạng.
* QtPositioning: bao gồm các class giúp việc hỗ trợ xác định vị.
* Enginio: module giúp các client truy cập các Cloud Services của Qt.
* QtWebSockets: cung cấp các công cụ cho WebSocket protocol.
* QtWebKit: cung cấp các class dùng cho làm việc với các trình duyệt Web, dựa trên thư viện WebKit2.
* QtWebKitWidgets: các widget cho WebKit.
* QtXml: các class dùng cho làm việc với XML file.
* QtSvg: dùng cho hiển thị các thành phần của SVG file.
* QtSql: cung cấp các class dùng cho việc làm việc với dữ liệu.
* QtTest: cung cấp các công cụ cho phép test các đơn vị của ứng dụng với PyQt5.

## CHƯƠNG 2

## THIẾT KẾ VÀ CÀI ĐẶT

### Thiết kế ứng dụng

Lưu đồ giải thuật:



Hình 8. Lưu đồ giải thuật về cách hoạt động của ứng dụng.

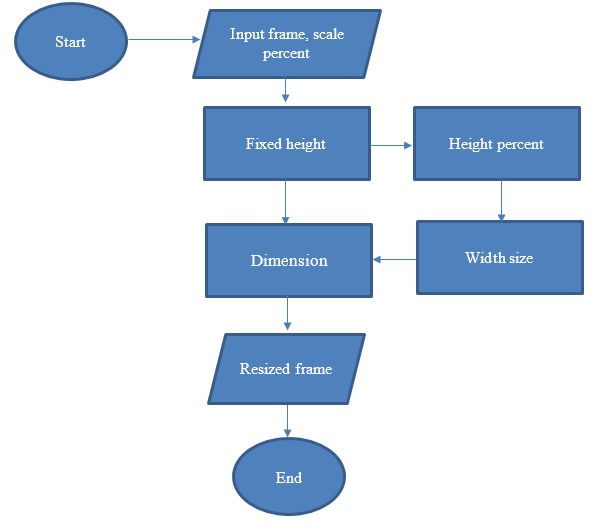
Ứng dụng cho phép người dùng chọn hai video đầu vào sau đó thực hiện vòng lặp duyệt qua từng frame của từng video, kiểm tra frame có tồn tại hay bị lỗi. Nếu có lỗi xảy ra, dừng vòng lặp, ngược lại không có lỗi thì frame sẽ được resize và thêm các pixel trắng vào sao cho bằng với kích thước của background input. Tiếp đến, frame sẽ được đưa vào tiến trình Mediapipe tiến hành nhận diện dáng người sau đó tách frame thành hai phần đối tượng và background rồi thay đổi phần background thành background input.

### Cài đặt ứng dụng

#### 2.1. Hàm thay đổi kích thước frame: resize()

Do video input có kích thước không phù hợp (lớn hơn chiều dài hoặc lớn hơn chiều rộng, quá nhỏ) so với background nên cần resize các frame của nó sao cho bằng với kích thước của background.

Lưu đồ giải thuật:



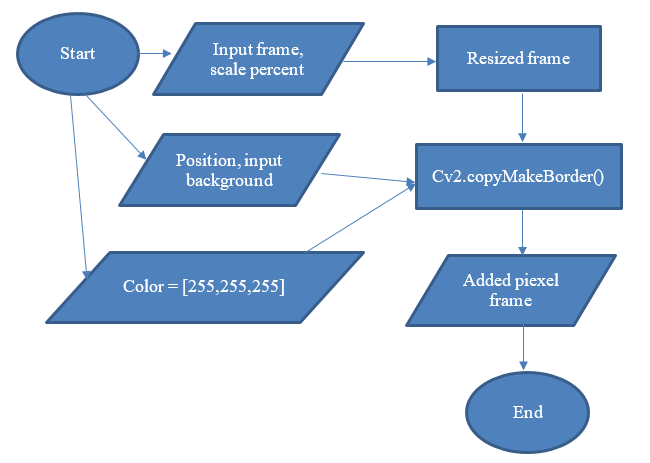
Hình 9. Lưu đồ giải thuật cho hàm resize.

Truyền vào frame và số phần trăm scale, thực hiện các phép tính toán để ra được kích thước của ảnh sao cho phù hợp với mong muốn.

#### 2.2. Hàm thêm pixel vào frame: add\_pixel()

Vì Mediapipe chỉ thực hiện nhận diện và tách nền của video nên khi chèn các video vào background (nền) khác, chúng ta cần thêm các pixel vào tất cả các frame của video sao cho toàn thể video input có kích thước bằng với background.

Lưu đồ giải thuật:



Hình 10. Lưu đồ giải thuật cho hàm add\_pixel.

Truyền vào vị trí, background và một mảng chứa giá trị màu tượng trưng cho màu trắng cùng với frame kết quả từ hàm resize() vào hàm cv2.copyMakeBorder() để thêm các pixel trắng vào frame đã được resize.

#### 2.3. Các class được sử dụng trong ứng dụng

**2.3.1. Class Capture:**

* Hàm init(): khởi tạo các biến sẽ lưu các giá trị sẽ sử dụng trong class:

|  |  |  |
| --- | --- | --- |
| Tên biến | Giá trị khởi tạo | Chức năng |
| capturing | False | Trạng thái phát/dừng video |
| saved\_frame\_array | Mảng rỗng | Chứa các frame kết quả đã lưu |
| video\_capture\_1 | None | Ghi video thứ nhất |
| video\_capture\_2 | None | Ghi video thứ hai |
| size | Tuple rỗng | Chứa kích thước của frame đầu ra |
| background | None | Đọc file background được truyền vào |

Bảng 1. Bảng các biến khởi tạo trong class Capture.

* Hàm setVideoFile\_1() và setVideoFile\_2() có chức năng tương tự nhau, nhận vào đường dẫn chứa video và ghi video vào biến video\_capture\_1, video\_capture\_2 tương ứng.
* Hàm open\_background() đảm nhiệm việc mở file background và lưu vào biến background đã khởi tạo.
* Hàm startCapture(), đây là hàm xử lý chính chức năng của ứng dụng bằng cách nhận giá trị từ các biến nêu trên và áp dụng các đặc trưng của vòng lặp While, xử lý ngoại lệ try – except của ngôn ngữ lập trình Python cùng với đó là giải pháp Mediapipe được đưa vào để xử lý các frame và từ đó tách chúng thành hai phần đối tượng và background. Cuối cùng thay đổi background đã tách thành background input.
* Hàm saveCapture() lưu từng frame kết quả từ hàm startCapture() thành video output ở thư mục hiện hành.
* Hàm pauseCapture() thay đổi trạng thái của biến capturing từ true thành false để tạm dừng video.
* Hàm quitCapture() kết thúc quá trình phát video.

**2.3.2. Class Window:**

Class Window nhận nhiệm vụ tạo ra giao diện cho ứng dụng, tương tác với người dùng thông qua các đặc trưng của thư viện PyQt5.

* Hàm init(): khởi tạo các nút chức năng và layout của ứng dụng:

Các nút chức năng được liên kết với các hàm trong Class Capture và trong chính class Window

|  |  |  |
| --- | --- | --- |
| Tên nút | Liên kết với hàm | Chức năng |
| Open Video 1 | open\_vid\_1() | Mở video thứ nhất |
| Open Video 2 | open\_vid\_2() | Mở video thứ hai |
| Open Background Image | open\_background() | Mở background |
| Start Video | startCapture() | Phát video kết quả |
| Pause Video | pauseCapture() | Tạm dừng video |
| Quit | quitCapture() | Thoát khỏi ứng dụng |

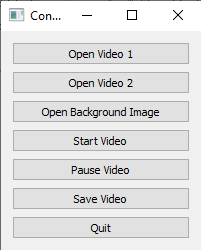
Bảng 2. Bảng các nút chức năng trong class Window.

* Hàm open\_vid\_1() và open\_vid\_2() nhận vào đường dẫn của video và truyền vào hàm setVideoFile\_1, setVideoFile\_2 tương ứng.

## CHƯƠNG 3

## KẾT QUẢ THỰC NGHIỆM

### Giao diện ứng dụng



Hình 11. Giao diện ứng dụng.

Ứng dụng được thiết kế đơn giản phục vụ mục đích học tập bao gồm các nút tương ứng với các chức năng:

* Nhấn vào nút Open Video 1 và Open Video 2: sẽ mở các cửa sổ để người dùng chọn video muốn ghép.
* Nhấn vào nút Open Background Image: sẽ mở ra cửa sổ để người dùng chọn background phù hợp với mục đích sử dụng.
* Nhấn vào nút Start Video: sẽ mở lên cửa sổ phát video kết quả đã ghép hai video input vào background đã chọn.
* Nhấn vào nút Pause Video: sẽ tạm dừng video kết quả cho đến khi người dùng nhấn vào nút Start Video để tiếp tục phát video kết quả.
* Nhấn vào nút Save Video: sẽ lưu video kết quả tại thư mục hiện hành.
* Nhấn vào nút Quit để dừng phát video kết quả đồng thời đóng ứng dụng.

1. **Kết quả thực nghiệm**

Cùng truyền vào một background có độ phân giải: 1920x1080 và kiểm thử với hai trường hợp:

**Trường hợp 1:**

Truyền vào hai video như hình:



Hình 12. Hình 12a: Cô gái áo đen, hình 12b: Cô gái áo trắng.

|  |  |  |
| --- | --- | --- |
| Video | Thời lượng | Độ phân giải |
| Thứ nhất (hình 12a) | 21 giây | 622x1064 |
| Thứ hai (hình 12b) | 31 giây | 1920x1080 |

Bảng 3. Bảng thông tin video kiểm thử cho trường hợp 1.

Ta được kết quả như hình dưới:



Hình 13a. Khung hình được cắt ra từ video kết quả của trường hợp 1.

Do thời lượng của video thứ hai dài hơn nên sẽ tiếp tục phát cho đến hết:



Hình 13b. Khung hình được cắt ra từ video kết quả của trường hợp 1.

**Trường hợp 2:**

Truyền vào hai video như hình:



Hình 14. Hình 14a: Cô gái áo đen, hình 14b: Hai cô gái đeo khẩu trang.

|  |  |  |
| --- | --- | --- |
| Video | Thời lượng | Độ phân giải |
| Thứ nhất (hình 14a) | 21 giây | 622x1064 |
| Thứ hai (hình 14b) | 21 giây | 1920x1080 |

Bảng 4. Bảng thông tin video kiểm thử cho trường hợp 2.

Ta được kết quả như hình dưới:



Hình 15. Khung hình được cắt ra từ video kết quả của trường hợp 2.

Ta có thể thấy ở trường hợp hai, khi truyền vào một video có nhiều người thì giải pháp Mediapipe chỉ nhận diện được một người duy nhất sau đó tiến hành ghép hai video vào background.

# PHẦN KẾT LUẬN

1. Kết quả đạt được

* Xây dựng thành công ứng dụng tách nền và ghép video bằng Mediapipe.
* Thành công trong việc tích hợp giải pháp Mediapipe vào giao diện ứng dụng được thiết kế bằng thư viện PyQt5.

1. Hạn chế

Ứng dụng còn nhiều tính năng thiếu sót:

* Chưa cho phép người dùng điều chỉnh vị trí và phần trăm scale của video so với background.
* Giao diện chưa bắt mắt, thu hút người dùng.
* Chưa cho phép người dùng chọn nhiều video cùng lúc.
* Chưa nhận diện được nhiều người trong một video.

1. Thuận lợi

Trong quá trình thực hiện em đã có những thuận lợi:

* Nhận được sự quan tâm, hỗ trợ và góp ý nhiệt tình từ giáo viên hướng dẫn.
* Học được nhiều điều mới lạ và kinh nghiệm làm thực tế.
* Rút ra được nhiều bài học từ đề tài niên luận và trong quá trình thực hiện đề tài về lỗi, sửa lỗi, kiểm thử và nhiều quá trình khác.
* Tiếp thu được nhiều kinh nghiệm quý báu từ giáo viên hướng dẫn và các bạn trong quá trình thực hiện đề tài.

1. Khó khăn

Bên cạnh những thuận lợi, em cũng gặp phải rất nhiều khó khăn:

* Khả năng chuyên môn về phân tích hệ thống.
* Chưa có kinh nghiệm xây dựng ứng dụng cũng như tối ưu về phần cứng.

1. Hướng phát triển

* Thiết kế giao diện, xây dựng bố cục trực quan và chuyên nghiệp hơn.
* Phát triển thêm các chức năng sẵn có và hoàn thiện hơn.
* Bổ sung các tính năng như: chọn nhiều video, cho phép người dùng điều chỉnh vị trí và phần trăm scale so với background.
* Nhận diện được nhiều người cũng như cải thiện độ chính xác và tốc độ xử lý.

# TÀI LIỆU THAM KHẢO

[1] **Phạm Nguyên Khang,** Giáo trình Trí Tuệ Nhân Tạo, Khoa Công Nghệ Thông tin và Truyền Thông, Trường Đại học Cần Thơ.

[2] **Carsten Rother, Vladimir Kolmogorov, Andrew Blake**, Bài báo "GrabCut": interactive foreground extraction using iterated graph cuts.

[3] **Ning Xu, Brian Price, Scott Cohen, Thomas Huang**,Bài báo Deep Image Matting.

[4] **Zhanghan Ke,**[**Jiayu Sun**](https://arxiv.org/search/cs?searchtype=author&query=Sun%2C+J)**,**[**Kaican Li**](https://arxiv.org/search/cs?searchtype=author&query=Li%2C+K)**,**[**Qiong Yan**](https://arxiv.org/search/cs?searchtype=author&query=Yan%2C+Q)**,**[**Rynson W.H. Lau**](https://arxiv.org/search/cs?searchtype=author&query=Lau%2C+R+W)**,** Bài báoMODNet:Real-Time Trimap-Free Portrait Matting via Objective Decomposition

[4] **google.github.io/mediapipe/** là trang giới thiệu, công bố những dự án, nghiên cứu, các giải pháp liên quan đến Trị Tuệ Nhân Tạo, Thị Giác Máy Tính… do Google phát triển. Truy cập vào 14h ngày 16/4/2022.

[5] **opencv.org** là trang chủ của thư viện OpenCV, nơi cung cấp các hàm, các giải pháp phục vụ cho ngành Trí Tuệ Nhân Tạo, Thị Giác Máy Tính, Xử Lý Ảnh... Truy cập vào 14h ngày 16/4/2022.