

TRƯỜNG ĐẠI HỌC DUY TÂN

KHOA CÔNG NGHỆ THÔNG TIN

****

**BÁO CÁO ĐỒ ÁN NHÓM**

**MÔN: ĐỒ ÁN CDIO**

**Tên đề tài**

**XÂY DỰNG MODEL VÀ TRANG WEB KHÔI PHỤC**

**HÌNH ẢNH SỬ DỤNG THUẬT TOÁN GAN(Generative Adversarial Network)**

**--🙖🙐✰🙖🙐--**

**Giáo viên hướng dẫn: T.S Đặng Việt Hùng**

**Sinh viên thực hiện: Mai Đăng Lanh 26211935482**

**Hoàng Minh Thắng 26211935124**

**Phan Quốc Đạt 26211222242**

**Hồ Văn Thanh Phương 26211936249**

**Lưu Tấn Đạt 26211935778**

**Đà Nẵng 10/2023**

**MỤC LỤC**

[**CHƯƠNG I:GIỚI THIỆU TỔNG QUAN** 9](#_Toc148263544)

[**1. Giới thiệu đề tài và mục đích** 9](#_Toc148263545)

[**1.1 Giới thiệu đề tài** 9](#_Toc148263546)

[**1.2 Mục đích:** 9](#_Toc148263547)

[**2. Ngôn ngữ** 9](#_Toc148263548)

[**2.1 Python** 9](#_Toc148263549)

[**2.2 Những đặc điểm của Python** 9](#_Toc148263550)

[**2.3 Thư viện Python** 10](#_Toc148263551)

[**2.4 Những lợi ích của Python** 11](#_Toc148263552)

[**2.5 Sử dụng Python** 11](#_Toc148263553)

[**3. Công cụ phát triển** 12](#_Toc148263554)

[**3.1 Kaggle** 12](#_Toc148263555)

[**3.2 Google Colaboratory** 13](#_Toc148263556)

[**3.3 Visual studio code** 13](#_Toc148263557)

[**CHƯƠNG II: GIỚI THIỆU THUẬT TOÁN** 14](#_Toc148263558)

[**2.1 Tổng quan về GAN** 14](#_Toc148263559)

[**2.2 Đặc Trưng cơ bản của GAN** 14](#_Toc148263560)

[**2.3 GAN hoạt động như thế nào :** 15](#_Toc148263561)

[**2.4 Các thành phần cơ bản của mạng GAN** 16](#_Toc148263562)

[**2.5 Huấn luyện GAN** 16](#_Toc148263563)

[**CHƯƠNG III: ÁP DỤNG GAN VÀO BÀI TOÁN IMAGE INPAITING** 18](#_Toc148263564)

[**3.1 Đặt vấn đề** 18](#_Toc148263565)

[**3.2 Áp dụng GAN vào bài toán image inpainting** 18](#_Toc148263566)

[**3.3 Model của GAN image inpainting** 19](#_Toc148263567)

[**3.3.1 Generator trong GAN image inpainting** 19](#_Toc148263568)

[**3.3.2 Discriminator trong GAN image inpainting** 19](#_Toc148263569)

[**3.3.3 Mask trong GAN image inpainting** 20](#_Toc148263570)

[**3.3.4 Kiến trúc của mạng GAN** 20](#_Toc148263571)

[**CHƯƠNG IV: ĐÀO TẠO MODEL VÀ DEPLOY TRANG WEB KHÔI PHỤC HÌNH ẢNH BẰNG THUẬT TOÁN GAN** 21](#_Toc148263572)

[**4.1 Dataset** 21](#_Toc148263573)

[**4.2 Quá trình đào tạo model** 21](#_Toc148263574)

[**4.3 Kết quả trên tập validation** 32](#_Toc148263575)

[**4.4 Giao diện của trang web** 33](#_Toc148263576)

[**5. Kết quả đạt được** 35](#_Toc148263577)

[**5.1 Những thành quả đạt được:** 35](#_Toc148263578)

[**5.2 Thảo luận:** 35](#_Toc148263579)

[**5.3 Hướng phát triển** 36](#_Toc148263580)

[**KẾT LUẬN** 37](#_Toc148263581)

[**Tài liệu tham khảo** 38](#_Toc148263582)

**CONCEIVE (Ý TƯỞNG)**

**1. Thành viên 1:** (Chốt ý tưởng)

- Họ và tên: Hoàng Minh Thắng

- Chuyên ngành: Big Data & Machine Learning

- Tên ý tưởng: Ứng dụng thuật toán GAN image inpainting để khôi phục hình ảnh

- Ngày/Tháng/Năm: 01/09/2023

**1.1 Mô tả ý tưởng sản phẩm**

Thuật toán GAN (Generative Adversarial Network) là một mô hình học sâu (deep learning) được giới thiệu bởi Ian Goodfellow và các đồng nghiệp vào năm 2014.Việc áp dụng thuật toán GAN trong việc khôi phục hình ảnh là một việc có tính thực tiễn rất cao .GAN image inpainting sẽ tạo ra hình ảnh giả lập có tính chân thật cao và giống nhất so với ảnh gốc để có thể khôi phục ảnh hoàn thiện nhất . Tuy nhiên, để đảm bảo độ chính xác của thuật toán, công việc cần phải có một lượng dữ liệu đủ lớn và đa dạng để đào tạo và kiểm tra thuật toán. Ứng dụng này có thể được áp dụng rộng rãi trong các ứng dụng an ninh, y tế,mĩ thuật ,tìm kiếm cứu hộ …

**1.2 Đánh giá cơ sở kiến thức của sinh viên:**

- Ngôn ngữ lập trình (Python): Đang học

- Deep Learning: GAN(Generative Adversarial Network)

**1.3 Đánh giá khả năng phát triển của sản phẩm:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ước tính số lượng người sẽ hưởng lợi từ sản phẩm của bạn: | 10000 | | | |
| Ước tính số lượng người sẽ bỏ tiền ra mua sản phẩm của bạn: | 200 | | | |
| Ước lượng số năm sản phẩm bạn sẽ ở trên thị trường cho đến khi bị thay thế: | 5 | | | |
| Ước lượng số năm cần thiết để phát triển sản phẩm của bạn: | 1 | | | |
| Liệt kê các đối tượng sẽ được hưởng lợi từ ý tưởng sản phẩm của bạn (ít nhất là 1, nhiều nhất là 5): | Khách hàng, đối tác,… | | | |
|  | Trung Bình | Khá | Cao | Rất Cao |
| Đánh giá khả năng các nhà đầu tư sẽ bỏ tiền ra cho ý tưởng sản phẩm của bạn: |  | X |  |  |
| Ước tính độ khó để phát triển thành công sản phẩm của bạn: |  | X |  |  |

**2. Thành viên 2**

- Họ và tên: Mai Đăng Lanh

- Chuyên ngành: Big Data & Machine Learning

- Tên ý tưởng: Ứng dụng các thuật toán machine learning để tạo chatbot tư vấn sức khỏe

- Ngày/Tháng/Năm: 01/09/2023

**2.1 Mô tả ý tưởng sản phẩm**

Ngày này việc tư vấn sức khỏe là việc cần thiết . để có thể tiết kiệm thời gian , tài nguyên thì việc ứng dụng chatbot được lập trình các thông số sức khỏe để tư vấn tại nhà là 1 việc hết sức cần thiết .Chatbot tư vấn sức khỏe là một chương trình máy tính được thiết kế để cung cấp thông tin và hỗ trợ về sức khỏe cho người dùng. Chatbot có thể được sử dụng để trả lời các câu hỏi về sức khỏe, cung cấp thông tin về các bệnh và tình trạng sức khỏe, và hỗ trợ người dùng trong việc quản lý sức khỏe của họ. Tuy nhiên, để đảm bảo độ chính xác của chatbot,thì chatbot cần phải có một lượng dữ liệu đủ lớn và đa dạng để đào tạo và kiểm tra . ngoài ra chat bot còn có thể ứng dụng rộng rãi trong các lĩnh vực dạy học , y tế , và cả bán hàng,…

**2.2 Đánh giá cơ sở kiến thức của sinh viên:**

- Ngôn ngữ lập trình (Python): Đang học

- Machine Learning, Deep Learning: Đang học

**2.3 Đánh giá khả năng phát triển của sản phẩm:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ước tính số lượng người sẽ hưởng lợi từ sản phẩm của bạn: | 30000 | | | |
| Ước tính số lượng người sẽ bỏ tiền ra mua sản phẩm của bạn: | 150 | | | |
| Ước lượng số năm sản phẩm bạn sẽ ở trên thị trường cho đến khi bị thay thế: | 3 | | | |
| Ước lượng số năm cần thiết để phát triển sản phẩm của bạn: | 1 | | | |
| Liệt kê các đối tượng sẽ được hưởng lợi từ ý tưởng sản phẩm của bạn (ít nhất là 1, nhiều nhất là 5): | Bệnh viện, khách hàng. … | | | |
|  | Trung Bình | Khá | Cao | Rất Cao |
| Đánh giá khả năng các nhà đầu tư sẽ bỏ tiền ra cho ý tưởng sản phẩm của bạn: |  | X |  |  |
| Ước tính độ khó để phát triển thành công sản phẩm của bạn: |  | X |  |  |

**3. Thành viên 3**

- Họ và tên (sinh viên): Phan Quốc Đạt

- Chuyên ngành (của sinh viên): Big Data & Machine Learning

- Tên ý tưởng (sản phẩm): áp dụng natural language processing để nhận diện cảm xúc qua giọng nói

- Ngày/Tháng/Năm: 26/02/2023

**3.1 Mô tả ý tưởng sản phẩm**

Một trong những ứng dụng thú vị của máy học là việc nhận diện giọng nói và cảm xúc của người nói . việc nhận diện cảm xúc thông qua giọng nói là một việc rất thú vị, tương tự như ngôn từ, ngữ điệu cũng là một thứ rất quan trọng khi chúng ta nói chuyện , việc có thể hiểu được ngữ điệu của nhau giúp chúng ta có thể hiểu rõ về nhau hơn.Ngoài ra khả năng suy luận xem liệu chủ nhân đang cảm thấy tức giận, hạnh phúc hay buồn rầu ngay từ ngữ điệu lời nói của họ có thể trở nên rất hữu ích cho người dùng thiết bị nhà ở thông minh. Tuy nhiên, để đảm bảo độ chính xác của thuật toán, công việc cần phải có một lượng dữ liệu đủ lớn và đa dạng để đào tạo và kiểm tra thuật toán. Ứng dụng này có thể được áp dụng rộng rãi trong các ứng dụng nhà thông minh , xe thông minh,…

**3.2 Đánh giá cơ sở kiến thức của sinh viên**

- Ngôn ngữ lập trình ( Python): Đã học

- Machine Learning: Đang học

**3.3 Đánh giá khả năng phát triển:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ước tính số lượng người sẽ hưởng lợi từ sản phẩm của bạn: | 10000 | | | |
| Ước tính số lượng người sẽ bỏ tiền ra mua sản phẩm của bạn: | 200 | | | |
| Ước lượng số năm sản phẩm bạn sẽ ở trên thị trường cho đến khi bị thay thế: | 2 | | | |
| Ước lượng số năm cần thiết để phát triển sản phẩm của bạn: | 1 | | | |
| Liệt kê các đối tượng sẽ được hưởng lợi từ ý tưởng sản phẩm của bạn (ít nhất là 1, nhiều nhất là 5): | Nhà đầu tư, doanh nghiệp, … | | | |
|  | Trung Bình | Khá | Cao | Rất Cao |
| Đánh giá khả năng các nhà đầu tư sẽ bỏ tiền ra cho ý tưởng sản phẩm của bạn: |  | X |  |  |
| Ước tính độ khó để phát triển thành công sản phẩm của bạn: |  | X |  |  |

# **CHƯƠNG I:GIỚI THIỆU TỔNG QUAN**

## **1. Giới thiệu đề tài và mục đích**

### **1.1 Giới thiệu đề tài :**

Ứng dụng thuật toán GAN (Generative Adversarial Network) để khôi phục hình ảnh là một trong số những chủ đề được nghiên cứu và bàn luận sâu trong khoảng thời gian gần đây. Việc khôi phục lại những hình ảnh đã bị hư hại hoặc mất đi một phần để có thể bảo tồn và ghi nhớ là một việc rất quan trọng trong thời đại số hiện nay.

Thuật toán GAN là một trong những thuật toán Deep Learning phổ biến nhất trong việc xử lý khôi phục hình ảnh. Với ứng dụng này, mô hình sẽ học cách nhận dạng Hình ảnh và tạo ra hình ảnh giả lập bằng cách học theo các chi tiết của hình ảnh được đưa vào. Đầu vào của mô hình là một bức ảnh bị khuất 1 phần và đầu ra của mô hình là một hình ảnh hoàn chỉnh với các phần bị khuất đã được tái tạo.

Với việc khôi phục hình ảnh , thì thuật toán có thể được ứng dụng rộng rãi trong rất nhiều lĩnh vực , với hội họa thì chúng ta có thể khôi phục các tác phẩm , với việc an ninh thì chúng ta có thể khôi phục được hình ảnh đã bị nhiễu . Ngoài ra đối với nước ta , viêc khôi phục các hình ảnh của các chiến sĩ đã ngã xuống trong các cuộc chiến bảo vệ tổ quốc là 1 vấn đề rất cần thiết.

### **1.2 Mục đích:**

Việc khôi phục hình ảnh đã bị mất hay bị hư hại là việc cần thiết để ta có thể xác minh vấn đề hoặc đơn giản chỉ là nhớ lại hình ảnh những người thân đã mất. Ngoài ra nó còn có tác dụng bảo quản và lưu giữ nhưng tác phẩm đã đi qua của thời đại. Việc khôi phục ảnh tốn rất nhiều thời gian và công sức, nhưng mà thành quả mang lại rất to lớn và rất cần thiết.

## **2. Ngôn ngữ**

### **2.1 Python**

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

### **2.2 Những đặc điểm của Python**

Python là một ngôn ngữ thông dịch, linh hoạt, dễ sử dụng, là một ngôn ngữ cao cấp và cũng là một ngôn ngữ lập trình hướng đối tượng.

### **2.3 Thư viện Python**

Thư viện là một tập hợp các mã thường xuyên được sử dụng mà các nhà phát triển có thể bao gồm trong những chương trình Python của họ để không phải lập trình từ đầu. Theo mặc định, Python đi kèm với Thư viện chuẩn, chứa rất nhiều các hàm có thể tái sử dụng. Ngoài ra, hơn 137.000 thư viện Python có sẵn cho các ứng dụng khác nhau, bao gồm phát triển web, khoa học dữ liệu và máy học (ML).

**2.3.1 Image Completion**

Các nhà phát triển sử dụng Matplotlib để hiển thị dữ liệu dưới dạng đồ họa hai và ba chiều (2D và 3D) chất lượng cao. Thư viện này thường được sử dụng trong các ứng dụng khoa học. Với Matplotlib, bạn có thể trực quan hóa dữ liệu bằng cách hiển thị dữ liệu dưới dạng các biểu đồ khác nhau, chẳng hạn như biểu đồ cột và biểu đồ đường. Bạn cũng có thể hiển thị nhiều biểu đồ cùng lúc và các chi tiết đồ họa có thể được di chuyển qua mọi nền tảng.

**2.3.2 Pandas**

Pandas cung cấp cấu trúc dữ liệu được tối ưu hóa và linh hoạt mà bạn có thể sử dụng để thao tác với dữ liệu chuỗi thời gian và dữ liệu có cấu trúc, chẳng hạn như bảng và nhóm. Ví dụ, bạn có thể sử dụng Pandas để đọc, ghi, hợp nhất, lọc và nhóm dữ liệu. Thư viện này được nhiều người sử dụng cho các tác vụ khoa học dữ liệu, phân tích dữ liệu và ML.

**2.3.3 NumPy**

NumPy là một thư viện phổ biến mà các nhà phát triển sử dụng để dễ dàng tạo và quản lý nhóm, thao tác với các hình dạng logic và thực hiện các phép toán đại số tuyến tính. NumPy hỗ trợ tích hợp với nhiều ngôn ngữ như C và C ++.

**2.3.4 Requests**

Thư viện Requests cung cấp nhiều hàm hữu ích cần thiết để phát triển web. Bạn có thể sử dụng thư viện này để gửi các yêu cầu HTTP, bổ sung tiêu đề, thêm tham số URL, bổ sung dữ liệu và thực hiện nhiều tác vụ khác khi giao tiếp với các ứng dụng web.

**2.3.5 OpenCV-Python**

OpenCV-Python là một thư viện mà các nhà phát triển sử dụng để xử lý hình ảnh cho các ứng dụng thị giác máy tính. Thư viện này cung cấp nhiều hàm cho các tác vụ xử lý hình ảnh như đọc và ghi hình ảnh cùng lúc, xây dựng môi trường 3D từ môi trường 2D cũng như chụp và phân tích hình ảnh từ video.

**2.3.6 Keras**

Keras là thư viện mạng nơ-ron chuyên sâu của Python với khả năng hỗ trợ tuyệt vời cho việc xử lý dữ liệu, trực quan hóa và hơn thế nữa. Keras hỗ trợ nhiều mạng nơ-ron. Thư viện này có cấu trúc mô-đun mang lại sự linh hoạt cho việc lập trình các ứng dụng sáng tạo.

**2.3.7 Tensorflow**

Với sự bùng nổ của lĩnh vực trí tuệ nhân tạo - AI trong thập kỷ vừa qua, machine learning và deep learning rõ ràng cũng phát triển theo cùng. Và ở thời điểm hiện tại, [TensorFlow](https://www.tensorflow.org/) chính là thư viện mã nguồn mở cho machine learning nổi tiếng nhất thế giới, được phát triển bởi các nhà nghiên cứu từ Google. Việc hỗ trợ mạnh mẽ các phép toán học để tính toán trong machine learning và deep learning đã giúp việc tiếp cận các bài toán trở nên đơn giản, nhanh chóng và tiện lợi hơn nhiều.

Các hàm được dựng sẵn trong thư viện cho từng bài toán cho phép TensorFlow xây dựng được nhiều neural network. Nó còn cho phép bạn tính toán song song trên nhiều máy tính khác nhau, thậm chí trên nhiều CPU, GPU trong cùng 1 máy hay tạo ra các dataflow graph – đồ thị luồng dữ liệu để dựng nên các model. Nếu bạn muốn chọn con đường sự nghiệp trong lĩnh vực A.I. này, nắm rõ những điều cơ bản của TensorFlow thực sự rất quan trọng.

### **2.4 Những lợi ích của Python**

Python có những lợi ích như sau:

* Python cải thiện năng suất làm việc của các nhà phát triển vì họ có thể sử dụng ít dòng mã hơn để viết một chương trình.
* Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ.
* Cộng đồng Python lớn.
* Trên internet có rất nhiều hữu ích nếu bạn muốn học Python.
* Python có thể được sử dụng trên nhiều hệ điều hành khác nhau như: Windows, macOS, Linux và Unix.

### **2.5 Sử dụng Python**

**2.5.1 Phát triển web phía máy chủ**

Python hữu ích trong việc lập trình mã phía máy chủ bởi vì ngôn ngữ này cung cấp nhiều thư viện bao gồm mã viết sẵn cho các hàm backend phức tạp. Các nhà phát triển cũng sử dụng một loạt các khung Python cung cấp tất cả những công cụ cần thiết để xây dựng ứng dụng web một cách nhanh chóng và dễ dàng hơn.

**2.5.2 Tự động hoá bằng các tập lệnh Python**

Ngôn ngữ tập lệnh là một ngôn ngữ lập trình tự động hóa các tác vụ mà thường được con người thực hiện. Các lập trình viên thường xuyên sử dụng các tập lệnh Python để tự động hóa nhiều tác vụ hàng ngày như:

* Đổi tên một số lượng lớn tệp cùng lúc.
* Chuyển đổi một tệp sang một loại tệp khác.
* Loại bỏ các từ trùng lặp trong tệp văn bản.
* Thực hiện các phép toán cơ bản.
* Gửi email.
* Tải xuống nội dung.
* Thực hiện phân tích nhật ký cơ bản.
* Tìm kiếm lỗi trong nhiều tệp.

**2.5.3 Khoa học dữ liệu và máy học**

Khoa học dữ liệu trích xuất thông tin quý giá từ dữ liệu và máy học (ML) dạy máy tính tự động học hỏi từ dữ liệu và đưa ra các dự đoán chính xác. Các nhà khoa học dữ liệu sử dụng Python cho các tác vụ khoa học dữ liệu sau:

* Sửa và loại bỏ dữ liệu không chính xác, hay còn được gọi là làm sạch dữ liệu.
* Trích xuất và chọn lọc các đặc điểm đa dạng của dữ liệu.
* Gắn nhãn dữ liệu, gán tên cho ý nghĩa dữ liệu.
* Tìm các số liệu thống kê khác nhau từ dữ liệu.
* Trực quan hoá dữ liệu bằng cách sử dụng các biểu đồ và đồ thị, chẳng hạn như biểu đồ đường, biểu đồ cột, biểu đồ tần suất và biểu đồ tròn.

Các nhà khoa học dữ liệu sử dụng những thư viện ML của Python để đào tạo các mô hình ML và xây dựng các công cụ phân loại giúp phân loại dữ liệu một cách chính xác. Các chuyên gia từ nhiều lĩnh vực sử dụng những công cụ phân loại dựa trên Python để thực hiện các tác vụ phân loại, chẳng hạn như phân loại hình ảnh, văn bản cũng như lưu lượng truy cập mạng, nhận dạng giọng nói và nhận diện khuôn mặt. Các nhà khoa học dữ liệu cũng sử dụng Python cho deep learning, một kỹ thuật ML nâng cao.

## **3. Công cụ phát triển**

### **3.1 Kaggle**

Kaggle là một nền tảng trực tuyến cho cộng đồng Machine Learning (ML) và Khoa học dữ liệu. Kaggle cho phép người dùng chia sẻ, tìm kiếm các bộ dữ liệu; tìm hiểu và xây dựng models, tương tác với những nhà khoa học và kỹ sư ML trên toàn thế giới; tham gia các cuộc thi để có cơ hội chiến thắng những giải thưởng giá trị. Người dùng Kaggle sẽ được hỗ trợ Graphic Processing Unit (GPU) và gần đây có thêm Tensor Processing Unit (TPU) để tăng tốc độ tính toán trong quá trình training cũng như inference.

### **3.2 Google Colaboratory**

Google Colaboratory, thường được gọi là Colab, là một dịch vụ cung cấp môi trường Jupyter Notebook miễn phí do Google cung cấp. Nó cho phép bạn chạy mã Python trong trình duyệt mà không cần cài đặt bất kỳ phần mềm nào trên máy tính cá nhân. Colab cung cấp khả năng sử dụng miễn phí các tài nguyên tính toán mạnh mẽ, bao gồm cả GPU và TPU, để thực hiện các tác vụ như học máy, học sâu và nghiên cứu khoa học dữ liệu.

### **3.3 Visual studio code**

Visual Studio Code hay VS Code là một **trình biên tập, soạn thảo đoạn code** giúp hỗ trợ quá trình xây dựng, thiết kế website nhanh chóng hơn. Ứng dụng **hỗ trợ đa nền tảng** và vận hành một cách mượt mà trên Windows, macOS và Linux. Không chỉ vậy, VS Code còn tương thích với nhiều máy tính có cấu hình tầm trung. Qua đó giúp người dùng dễ dàng sử dụng phần mềm này hơn.

# **CHƯƠNG II: GIỚI THIỆU THUẬT TOÁN**

## **2.1 Tổng quan về GAN**

GAN thuộc nhóm generative model. Generative là tính từ nghĩa là khả năng sinh ra, model nghĩa là mô hình. Vậy hiểu đơn giản generative model nghĩa là mô hình có khả năng sinh ra dữ liệu. Hay nói cách khác, GAN là mô hình có khả năng sinh ra dữ liệu mới. Ví dụ như những ảnh mặt người ở dưới bạn thấy là do GAN sinh ra, không phải mặt người thật. Dữ liệu sinh ra nhìn như thật nhưng không phải thật.

Mạng GAN đã được đề xuất bởi Goodfellow vá cộng sự(2014). Trong những năm gần đây, GAN đã được nghiên cứu rộng rãi kết hợp với các thuật toán học máy khác trong một số ứng dụng cụ thể, chẳng hạn như học bán giám sát (Odena, 2016), học chuyển giao (Cho và cộng sự, 2017) và học tăng cường (Wang và cộng sự, 2020), và được sử dụng rộng rãi trong việc vẽ tranh bằng hình ảnh. GAN đã tạo ra bước đột phá đáng kể trong việc vẽ hình ảnh bằng cách tạo ra những hình ảnh chân thực. Ý tưởng cốt lõi của GAN xuất phát từ “trò chơi hai người chơi”, trong đó các mạng được tối ưu hóa bằng cách lừa dối lẫn nhau giữa n generators và discriminators, dẫn đến trạng thái cân bằng Nash. GAN bao gồm mạng tạo G và mạng phân biệt D và cấu trúc của nó được hiển thị như sau.

Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động  
Sơ đồ cơ bản của mạng GAN

Mô hình gồm có đầu vào là một vecto ngẫu nhiên sau đó được đi qua các lớp đặc trưng của Generator, sau khi hình ảnh giả được tạo ra thì đi qua lớp Discriminator và phân biệt và cập nhập trọng số để sinh ra ảnh tốt hơn.

**2.2 Đặc trưng cơ bản của GAN**

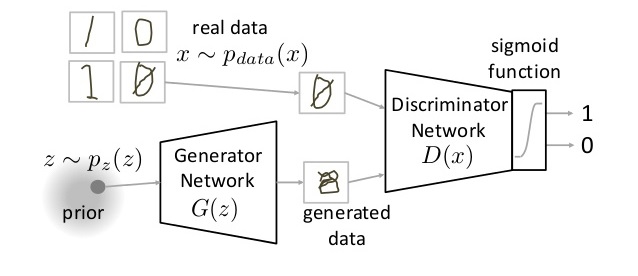
Mạng GAN (Generative Adversarial Networks) là một mô hình học máy có hai mô hình đối lập cạnh tranh với nhau để tạo ra dữ liệu giống thật. Hai mô hình này là:

* Mô hình tổng quát (Generator): Mô hình này có nhiệm vụ tạo ra dữ liệu mới, giống với dữ liệu thật.
* Mô hình phân biệt (Discriminator): Mô hình này có nhiệm vụ phân biệt dữ liệu thật và dữ liệu giả.

Hai mô hình này được đào tạo cùng nhau trong một quá trình cạnh tranh. Mô hình tổng quát cố gắng tạo ra dữ liệu giả sao cho giống với dữ liệu thật, còn mô hình phân biệt cố gắng phân biệt dữ liệu thật và dữ liệu giả. Quá trình này sẽ tiếp tục cho đến khi mô hình tổng quát tạo ra được dữ liệu giả mà mô hình phân biệt không thể phân biệt được.

Bằng cách học ánh xạ phân bố xác suất Pdata của dữ liệu thực, mạng sinh G dự kiến sẽ xuất ra nội dung G(z) gần với dữ liệu thực. Mạng phân biệt D cần xác định nguồn dữ liệu đầu vào càng nhiều càng tốt, tức là phân loại x và G(z). Khi mạng phân biệt D không thể phân biệt các nguồn dữ liệu, hiệu suất mạng là tối ưu. Hàm mục tiêu như sau:

**2.3 GAN hoạt động như thế nào :**



Hoạt động của mạng GAN

* phân phối của dữ liệu thực tế
* X mẫu từ
* phân phối của bộ tạo
* Z mẫu từ
* G(z) Mạng Tạo (Generator Network)
* D(x) Mạng Phân biệt (Discriminator Network)

Mạng sinh (Generator Network) nhận vào một đầu vào ngẫu nhiên và cố gắng tạo ra một mẫu dữ liệu. Trong hình ảnh trên, chúng ta có thể thấy rằng bộ tạo G(z) nhận vào một đầu vào z từ p(z), trong đó z là một mẫu từ phân phối xác suất p(z). Sau đó, nó tạo ra một tập dữ liệu sau đó được đưa vào một mạng phân biệt D(x).

Mạng Phân biệt (Discriminator Network) là nhận đầu vào từ dữ liệu thực tế hoặc từ bộ tạo và cố gắng dự đoán liệu đầu vào đó có phải là thực tế hay được tạo ra. Nó nhận đầu vào x từ trong đó là phân phối dữ liệu thực tế của chúng ta. D(x) sau đó giải một vấn đề phân loại nhị phân bằng cách sử dụng hàm sigmoid để đưa ra đầu ra trong khoảng từ 0 đến 1.

**2.4 Các thành phần cơ bản của mạng GAN**

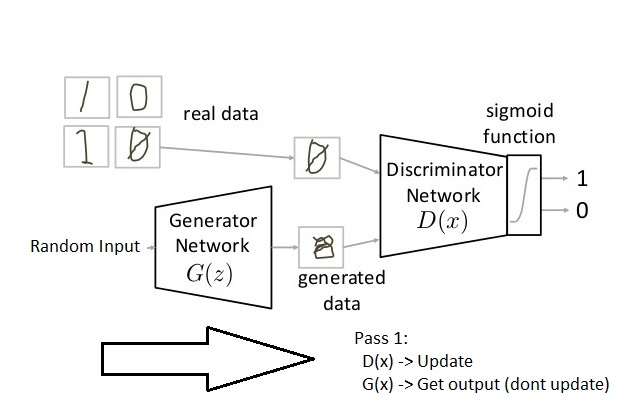
Mạng GAN gồm hai thành phần cơ bản là Mạng Sinh và Mạng phân biệt :

* Mạng Sinh(Generator) trong GAN là một mô hình học máy có nhiệm vụ tạo ra dữ liệu mới, giống với dữ liệu thật. Mạng sinh được đào tạo cùng với mạng phân biệt (Discriminator) trong một quá trình cạnh tranh. Mô hình sinh cố gắng tạo ra dữ liệu giả sao cho giống với dữ liệu thật, còn mô hình phân biệt cố gắng phân biệt dữ liệu thật và dữ liệu giả. Quá trình này sẽ tiếp tục cho đến khi mô hình sinh tạo ra được dữ liệu giả mà mô hình phân biệt không thể phân biệt được.
* Mạng phân biệt (Discriminator) trong GAN là một mô hình học máy có nhiệm vụ phân biệt dữ liệu thật và dữ liệu giả. Mạng phân biệt được đào tạo cùng với mạng sinh (Generator) trong một quá trình cạnh tranh. Mô hình sinh cố gắng tạo ra dữ liệu giả sao cho giống với dữ liệu thật, còn mô hình phân biệt cố gắng phân biệt dữ liệu thật và dữ liệu giả. Quá trình này sẽ tiếp tục cho đến khi mô hình sinh tạo ra được dữ liệu giả mà mô hình phân biệt không thể phân biệt được.

## **2.5 Huấn luyện GAN**

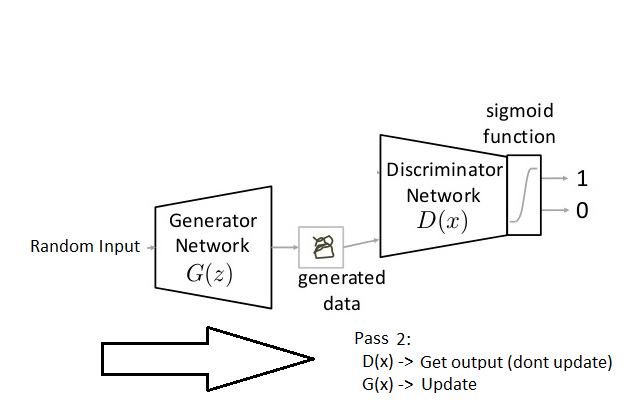
Về cơ bản, giai đoạn đào tạo có hai phần chính và chúng được thực hiện theo thứ tự:

**Bước 1:** Training Generator và đóng Discriminator (đóng băng có nghĩa là đặt đào tạo thành sai. Mạng chỉ thực hiện chuyển tiếp và không áp dụng backpropagation)



Phase 1 của Generator

**Bước 2:** Training Discrimator và đóng Generator



Phase 2 của mạng Discriminator

# **CHƯƠNG III: ÁP DỤNG GAN VÀO BÀI TOÁN IMAGE INPAITING**

## **3.1 Đặt vấn đề**

Với sự phát triển nhanh chóng của công nghệ thị giác máy tính, hình ảnh kỹ thuật số đã trở thành xu hướng chủ đạo trong việc thu nhận hình ảnh khuôn mặt. Thông thường, mọi người thường dựa vào các thiết bị điện tử để thu được hình ảnh khuôn mặt; tuy nhiên, hình mờ bị tắc, vết nhòe, một phần vùng bị thiếu và các vấn đề khác thường xuất hiện trong quá trình truyền hình ảnh kỹ thuật số, làm hỏng chất lượng hình ảnh khuôn mặt và dẫn đến cảm giác thị giác kém. Để giải quyết các vấn đề nêu trên, các học giả liên quan đã bắt đầu nghiên cứu các loại vấn đề này và đề xuất một loạt các phương pháp inpainteing.

Image Inpainting là một nhiệm vụ rất khó khăn trong xử lý hình ảnh, mục đích của nó là khôi phục và hoàn thiện phần hình ảnh bị thiếu hoặc bị biến dạng. Một hình ảnh mới cần được suy luận và xây dựng dựa trên thông tin ngữ cảnh của hình ảnh bị hỏng và cấu trúc hình ảnh tổng thể. Hình ảnh được khôi phục phải có kết cấu rõ ràng và các pixel ranh giới tự nhiên, đồng thời phù hợp với nhận thức thị giác của con người. Ví dụ rất khó để suy ra hình ảnh mũi hợp lý dựa trên các khu vực xung quanh khi thiếu phần mũi, điều này có thể dẫn đến tỷ lệ hình ảnh khuôn mặt mất cân đối. Đối với vấn đề này, cần phải tái tạo lại hình ảnh thỏa mãn nhận thức thị giác của con người theo một lượng lớn thông tin trước đó.

Các phương pháp deep learning gần đây đã có thể đưa ra một kết quả hứa hện với thử thách vẽ tranh vào các vùng bị thiếu hụt hình ảnh.những phương pháp này cho ra các cấu trúc và kết hợp hình ảnh một các hợp lý nhưng lại khiến chi tiết không phù hợp với các vị trí xung quanh. Điều này chủ yếu là do các mạng nơ ron tích chập được chọn chưa được phù hợp cho việc hồi phục hình ảnh. Với sự pháp triển không ngừng của các thuật toán hiện nay, chúng tôi xin được phép giới thiệu về mạng GAN cụ thể hơn là GAN image inpainting.

## **3.2 Áp dụng GAN vào bài toán image inpainting**

Trước khi các phương pháp deep learning được đề xuất, có hai loại nghiên cứu lý thuyết về inpaint trong hình ảnh, bao gồm partial differential equations(phương trình vi phân từng phần) và texture-based methods(phương pháp dựa trên kết cấu).

Facial inpainting dựa trên deep learning phù hợp hơn với nhiều tình huống khôi phục khác nhau so với các phương pháp khôi phục hình ảnh truyền thống. Tập dữ liệu phân phối đặc trưng được mạng nơ-ron học phù hợp hơn cho việc khôi phục hình ảnh khuôn mặt với diện tích bị thiếu lớn và hư hỏng ngẫu nhiên. Các chi tiết kết cấu không chỉ chính xác mà các đường nét cũng hài hòa và hình ảnh khuôn mặt phù hợp với ngữ nghĩa ngữ cảnh. Sau khi các học giả liên quan liên tục nghiên cứu chuyên sâu, các phương pháp sửa chữa hình ảnh dựa trên deep learning đã mang lại một số kết quả tích cực.

Pathak và cộng sự(2016) đã sử dụng context encoder(bộ mã hóa ngữ cảnh) để hoàn thành nhiệm vụ image inpainting, đây là phương pháp in hình ảnh đầu tiên dựa trên mạng đối thủ tổng quát(GAN).

## **3.3 Model của GAN image inpainting**

### **3.3.1 Generator trong GAN image inpainting**

Bộ generator được chia thành bộ mã hóa(encoder) và bộ giải mã(decoder) (Sun và cộng sự 2018). Bộ mã hóa chịu trách nhiệm nén và trích xuất thông tin đặc điểm từ một hình ảnh không đầy đủ và bộ giải mã chịu trách nhiệm khôi phục đặc điểm được nén đầu vào cho hình ảnh. Trong phương pháp này, bộ mã hóa bối cảnh có thể đạt được hiệu quả sửa chữa tốt, nhưng tổn thất đối kháng thế hệ được bộ mã hóa bối cảnh chấp nhận chỉ xem xét thông tin cục bộ của một vùng không hoàn chỉnh chứ không phải sự gắn kết ngữ nghĩa tổng thể của hình ảnh.

Mạng sinh trong GAN image inpainting được sử dụng tự động bằng các phép tính tích chập .Hình ảnh train được nạp vào kích thước ngẫu nhiên , sau đó hình ảnh sẽ được scale size thành 32x32 pixel sau đó cho qua các thư viện và để học các đặc trưng cơ bản của hình ảnh (ở đây sẽ là hình ảnh mặt người) ở đây sử dụng mạng CNN để nhận diện đặc trưng của hình ảnh , sau đó vẽ lên chỗ khuất bằng các pixel đặc trưng , sau đó lại scale size về lại kích thước ban đầu và xuất ra hình ảnh

### **3.3.2 Discriminator trong GAN image inpainting**

Iizuka và cộng sự. (2017) đã áp dụng bộ phân biệt kép toàn global–local để cải thiện bộ mã hóa bối cảnh. Bộ phân biệt đối xử cục bộ được áp dụng cho kết quả sửa chữa của một khu vực chưa hoàn thiện và bộ phân biệt toàn cục được áp dụng cho kết quả sửa chữa tổng thể. Thiết kế này không chỉ đảm bảo tính chính xác của khu vực sửa chữa mà còn đảm bảo tính toàn vẹn của kết quả cuối cùng. Tuy nhiên, kết quả dự đoán của phương pháp này vẫn chưa chính xác khi thiếu hình ảnh khuôn mặt diện rộng.

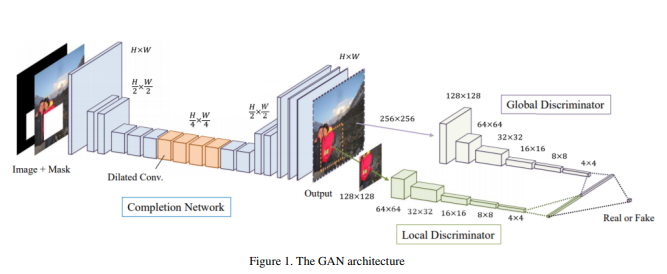
Mạng Global Discriminator nhận toàn bộ hình ảnh làm đầu vào và áp dụng các phép tích chập liên tiếp với bước nhảy (stride) bằng 2. Cuối cùng, nó có một tầng tuyến tính với kích thước 1024 làm tầng cuối cùng.

Local Discriminator cơ bản giống với Global Discriminator, ngoại trừ việc nó nhận đầu vào là một vùng nhỏ hơn chứa và được tập trung vào vùng bị mất mà Generator đã điền vào. Vùng này luôn có kích thước cố định, trong dự án này là 32x32 pixel.

### **3.3.3 Mask trong GAN image inpainting**

Masks được sử dụng để biểu diễn các vùng trống là các mảng gồm toàn số không, với các số một ở các vị trí tạo thành vùng trống. Đối với quá trình huấn luyện, masks được tạo ngẫu nhiên. Chúng có hình dạng hình chữ nhật với chiều cao và chiều rộng được lấy mẫu ngẫu nhiên theo phân phối đều, độc lập với nhau, trong khoảng từ 5 đến 12 điểm ảnh.

### **3.3.4 Kiến trúc của mạng GAN**



Đầu vào sẽ là hình ảnh và lớp Mask bị khuất sau đó hình ảnh sẽ được scale thành 32x32 pixel sau đó sẽ tiếp tục qua lớp Completion Network để học các đặc trưng hình ảnh và bắt đầu lắp đầy hình ảnh sau đó xuất hình ảnh ra ngoài.

**3.3.5 Loss function của mạng GAN**

Hàm Loss function được gộp lại từ loss function của Discriminator và loss function của Generator.

Ở phase của Generator thì hàm loss function của Discriminator phải là min để cho Generator có kết quả bằng 1.

Ở phase của Discriminator thì hàm loss funcion của Generator phải là min để Generator có kết quả bằng 0.

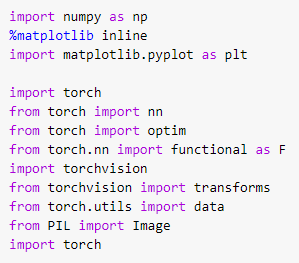
# **CHƯƠNG IV: ĐÀO TẠO MODEL VÀ DEPLOY TRANG WEB KHÔI PHỤC HÌNH ẢNH BẰNG THUẬT TOÁN GAN**

## **4.1 Dataset**

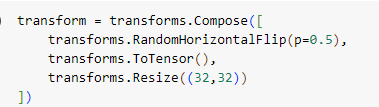
Bộ dữ liệu dataset là 100 bước hình ảnh mặt người được thu thập từ trang web Kaggle gồm có nam nữ trẻ em và người gia với độ phân giải 32x32 pixel.



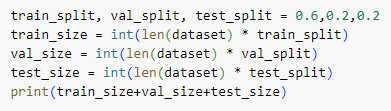
## **4.2 Quá trình đào tạo model**

****

Khai báo thư viện



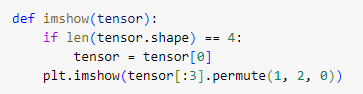
Thay đổi kích thước ảnh đầu vào thành 32pixel x 32 pixel



Chuẩn bị dữ liệu để train



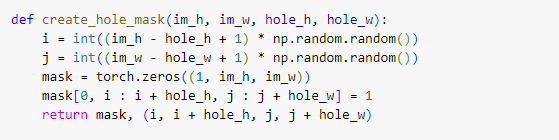
Chia dữ liệu thành các tập train, test và validation dựa trên các kích thước đã được tính toán trước đó



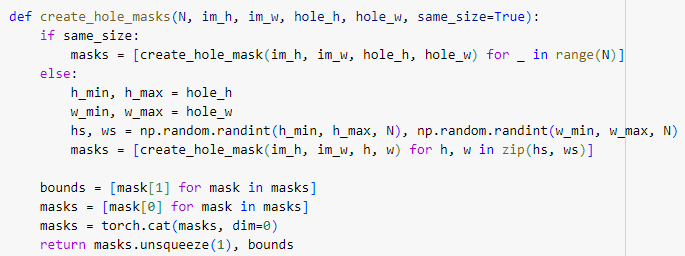
Hàm imshow để hiển thị một tensor dưới dạng hình ảnh

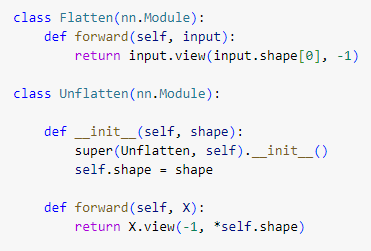


Hình ảnh sau khi được thay đổi kích thước đầu vào thành 32pixel x 32 pixel



Hàm create\_hole\_mask dùng để tạo một lớp mask có kích thước tương ứng với hình ảnh là một vùng trắng và một hole trong hình ảnh

Hàm create\_hole\_masks dùng để tạo một batch với các mask và vị trí khác nhau



Lớp Flatten biến đổi tensor từ dạng có kích thước nhiều chiều thành dạng phẳng và lớp Unflatten biến đổi lại tensor từ dạng phẳng thành dạng ban đầu theo kích thước được xác định trong quá trình khởi tạo.



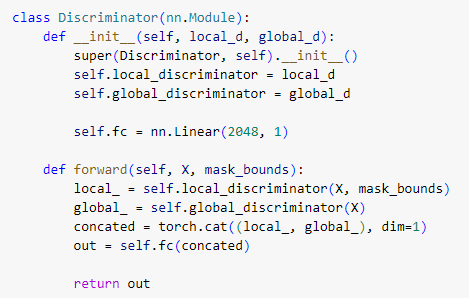
Lớp Generator gồm các hàm Conv2d , leakyRelu,hàm BatchNorm2d, Hàm Conv2d thực hiện phép biến đổi tích chập 2 chiều,hàm leakyRelu có tác dụng học các đặc trưng của hình ảnh,lớp chuẩn hóa theo batch giúp cải thiện tốc độ hội tụ của quá trình huấn luyện bằng cách chuẩn hóa các giá trị đầu vào của các lớp mạng.



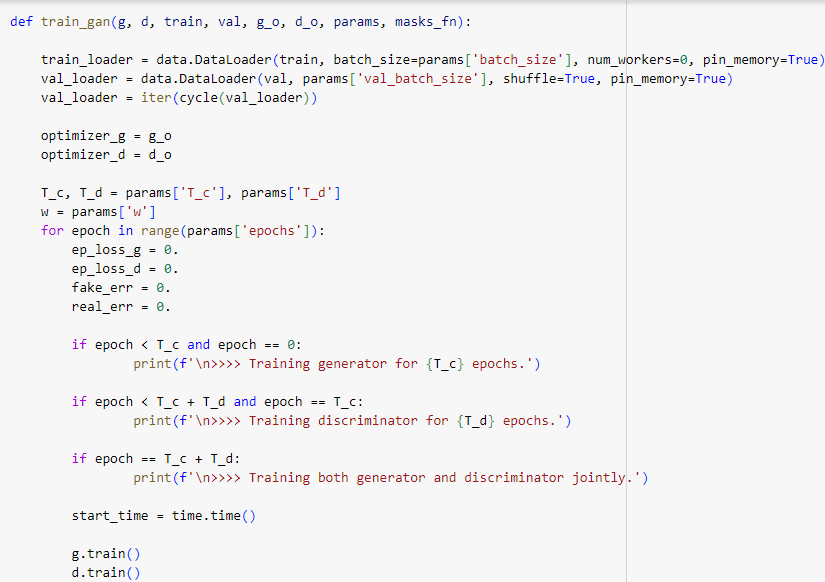
LocalDiscriminator có chức năng đánh giá các vùng cục bộ của hình ảnh

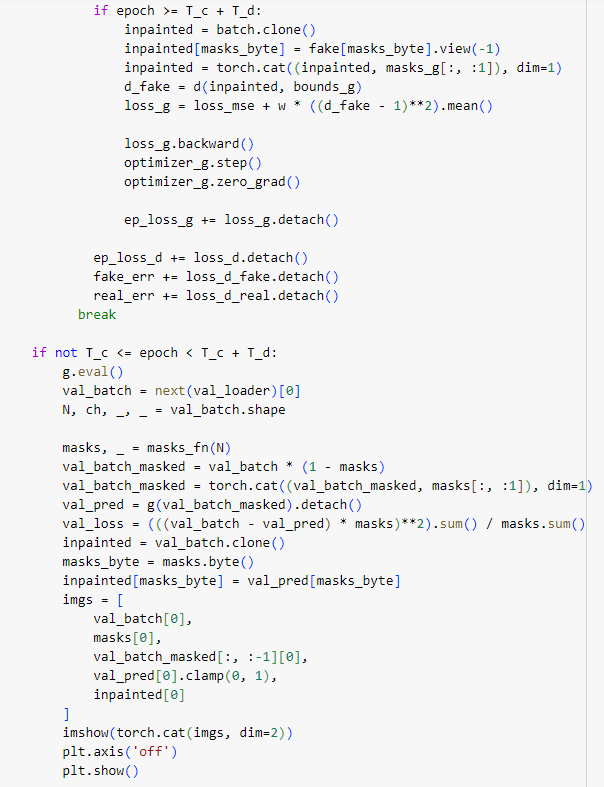
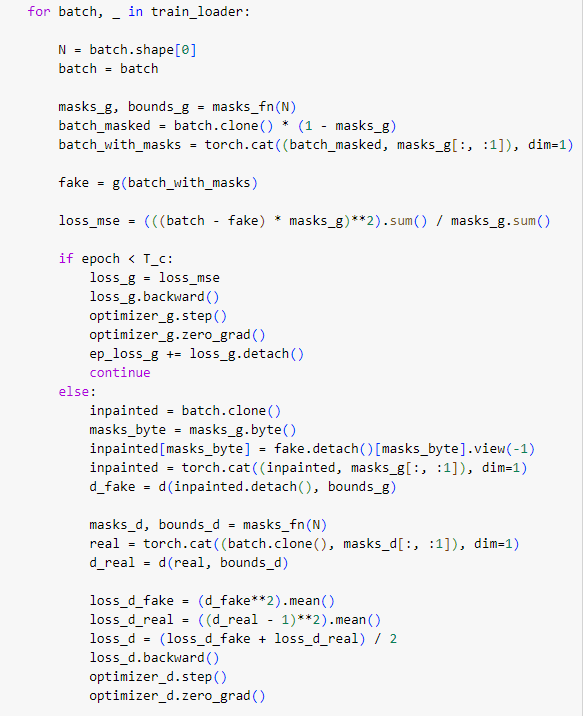


Lớp GlobalDiscriminator định nghĩa một mô hình mạng neural dùng để đánh giá toàn bộ hình ảnh.



Lớp Discriminator kết hợp cả hai mô hình local\_discriminator và global\_discriminator để tạo thành một mô hình phân biệt hoàn chỉnh.

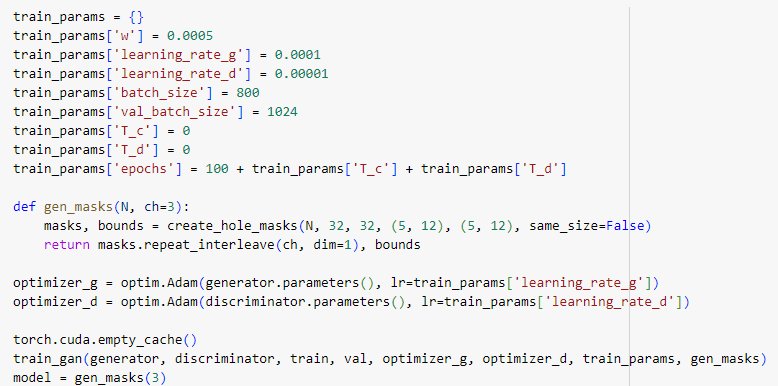




Hàm train GAN có nhiệm vụ là huấn luyện mô hình GAN

Giải thích về train GAN:

* Đầu tiên GAN nhận vào các thông số
* Sau đó tạo các DataLoader cho dữ liệu huấn luyện Validation
* Sau đó thiết lập các optimizer cho generator và discriminator.
* Sau đó Lặp qua các epochs:
* Huấn luyện generator trong T\_c epochs đầu tiên.
* Huấn luyện discriminator trong T\_d epochs tiếp theo.
* Sau khi huấn luyện xong discriminator, cả hai mô hình generator và discriminator sẽ được huấn luyện cùng một lúc.
* Tính toán loss và cập nhật các tham số của generator và discriminator.
* Theo dõi các thông số liên quan đến loss.
* Hiển thị hình ảnh validation để theo dõi quá trình huấn luyện.

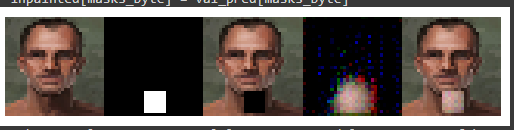


Quá trình train model

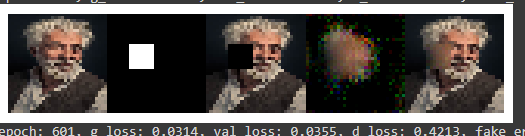
## **4.3 Kết quả trên tập validation**

Sau khi training 4000 epochs trong thời gian 6 tiếng bằng Google Colab với GPU T4 chúng em thu được kết quả là:

Sau 100 lần train ảnh thu được kết quả như sau:

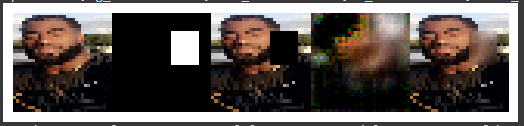
  
Quá trình train ở giai đoạn này thì các mật độ pixel có sự xuất hiện , nhưng mật độ không đồng đều và vẫn còn bị nhiễu bởi các điểm pixel RGB

Sau 500 lần train ảnh thu được kết quả như sau:



Quá trình train ở giai đoạn này , các mật độ điểm ảnh đã xuất hiện nhiều hơn , đã có phân vùng sáng tối , nhưng vẫn còn bị nhiễu bởi các pixel RGB

Sau 1000 lần train ảnh thu được kết quả như sau:



Quá trình train ở giai đoạn này , các mặt độ điểm ảnh đã xuất hiện khá nhiều , đã có phân vùng sáng tối rõ hơn , song song với đó độ nhiễu của các pixel RGB đã ít đi

Sau 1500 lần train ảnh thu được kết quả như sau:****

Ở giai đoạn này thì mật độ pixel đã có thể hoàn chỉnh cả tấm hình và hình ảnh đã xuất hiện rõ nét các ví trị hơn , vùng sáng tối ổn định hơn và các mật độ pixel RGB hầu như xuất hiện rất ít

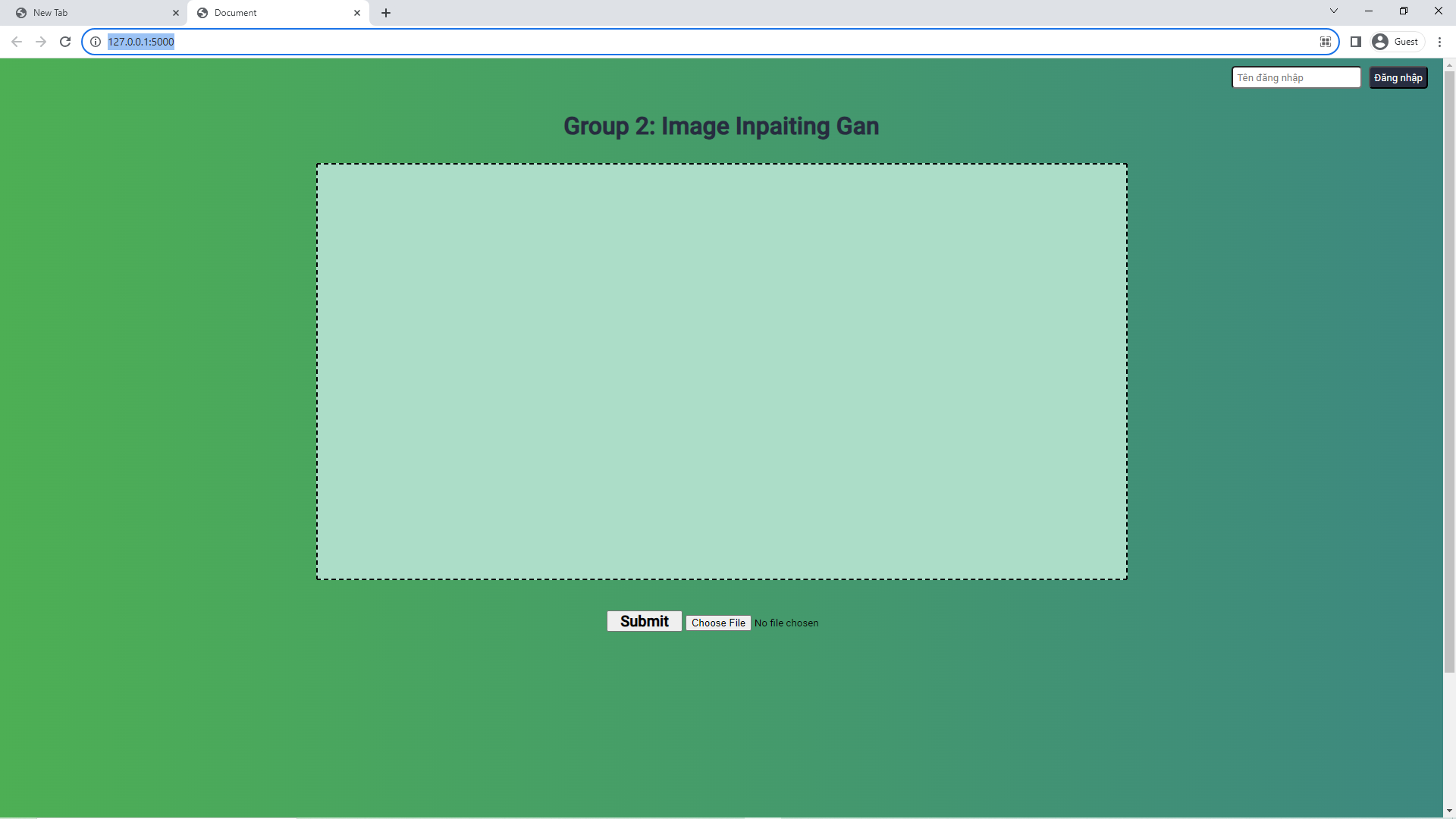
Sau 4000 lần train ảnh thu được kết quả như sau:



Sau hơn 4000 lần , thì các mật độ điểm ảnh đã có thể phân chia các vùng trên khuôn mặt rõ ràng hơn , độ sáng tối gần như hoàn thiện , và ảnh không còn bị nhiễu bởi các pixel RGB nữa

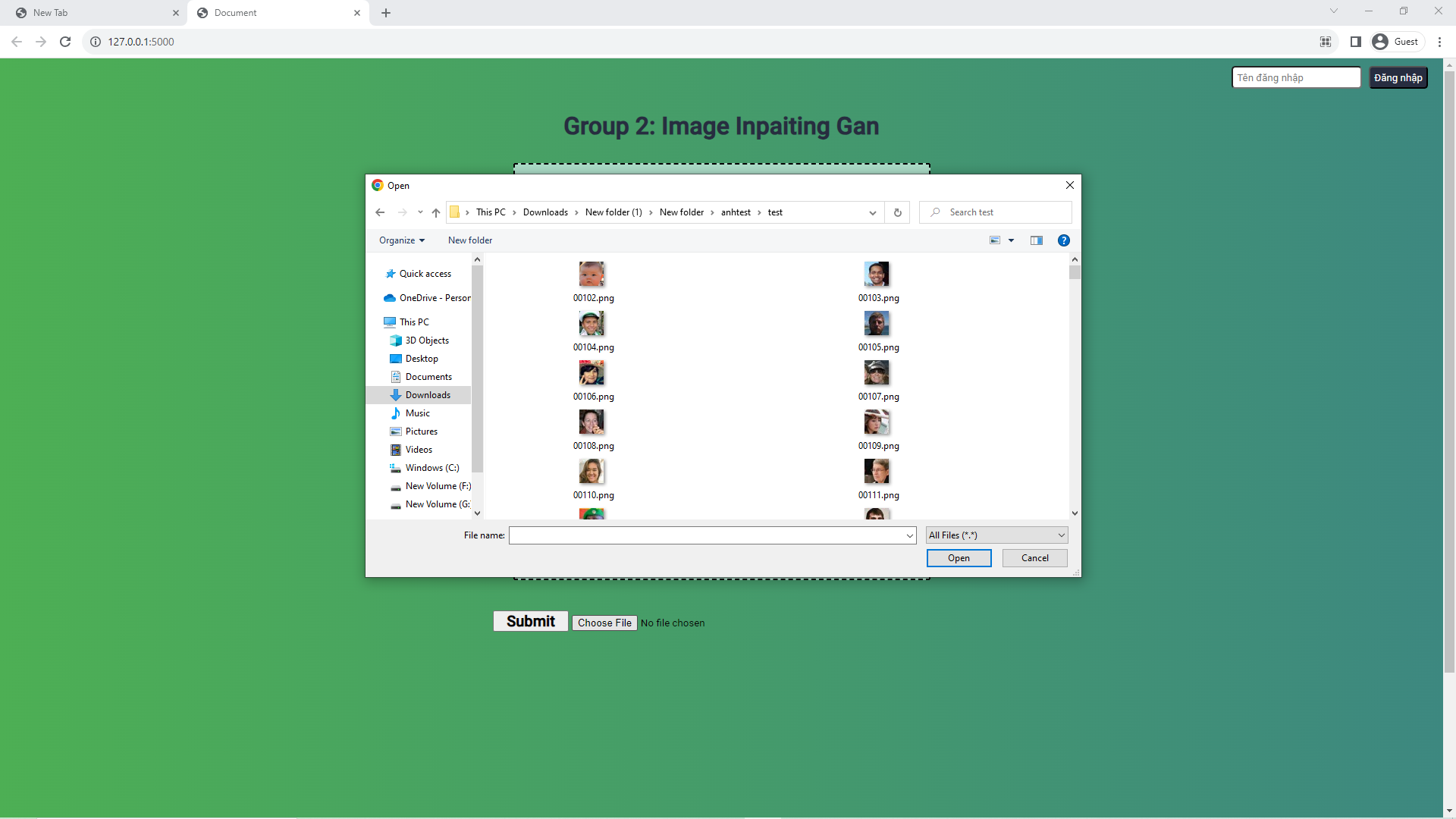
## **4.4 Giao diện của trang web**

Áp dụng flask vào việc deploy trang web.



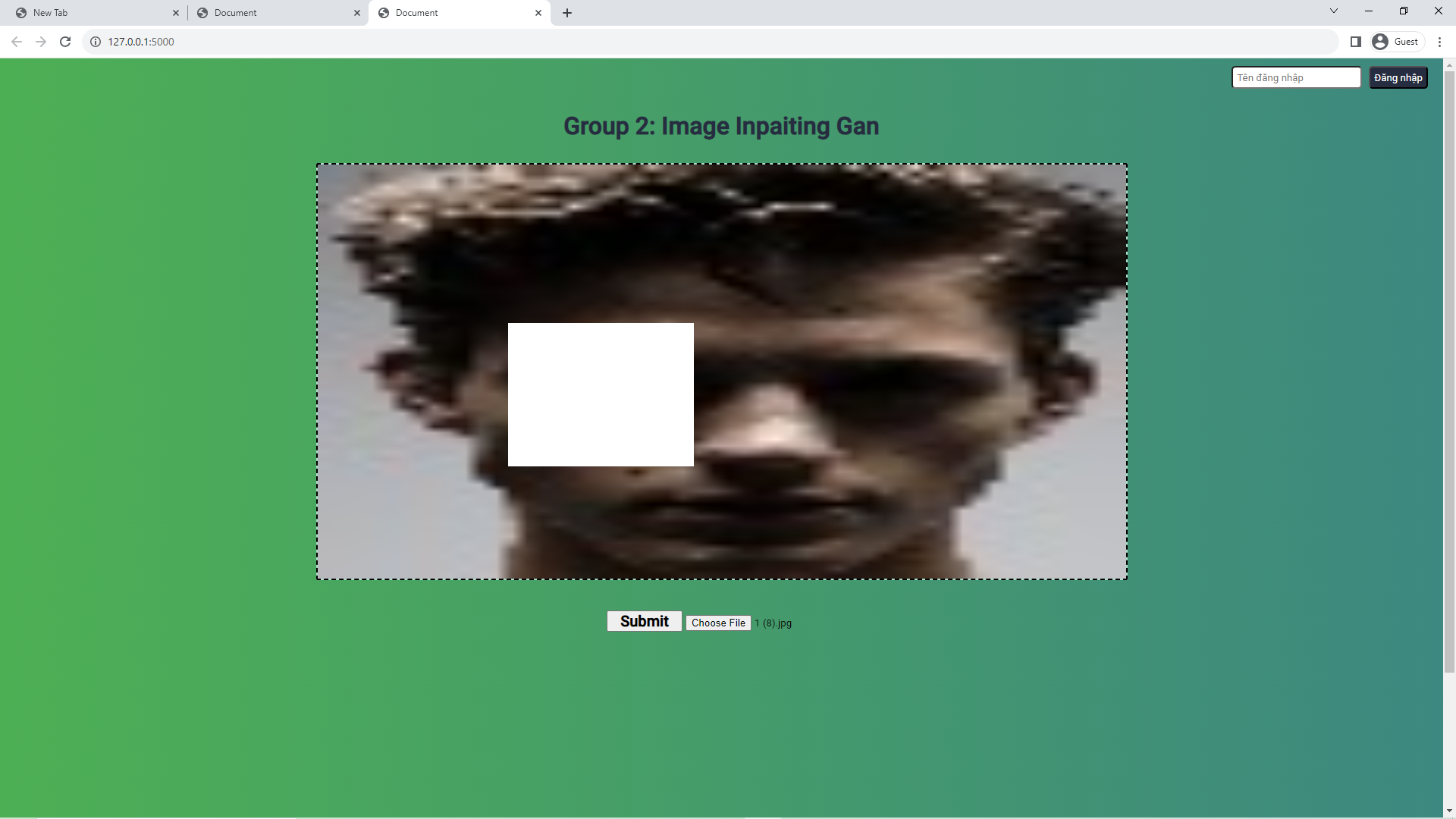
Trang web khôi phục hình ảnh

Chọn hình ảnh bằng cách click vào nút choose file.



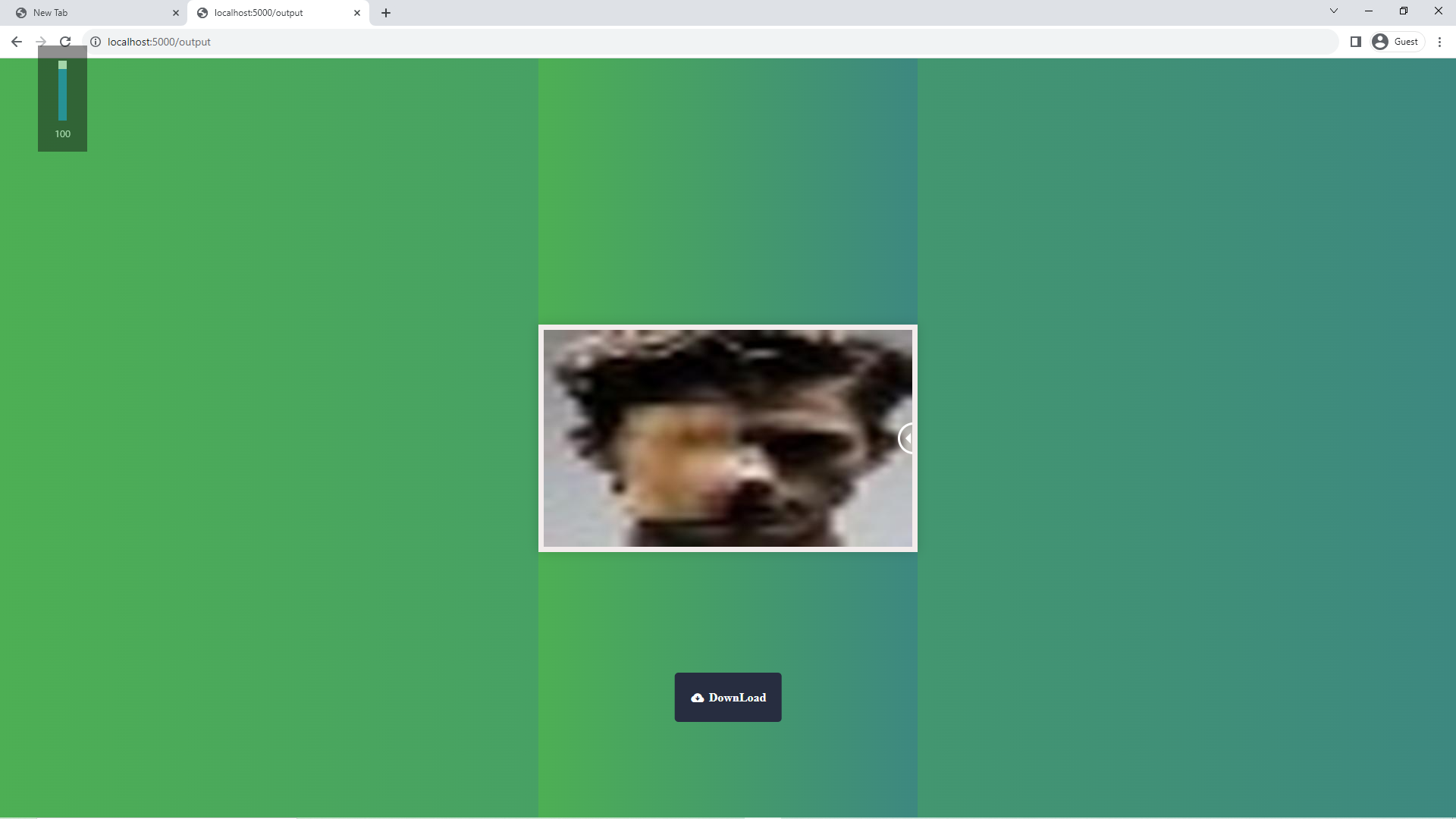
Chọn file hình ảnh cần khôi phục

Nhấp chuột trái vào hình ảnh để chọn vùng cần khôi phục.



Vùng chọn hình chữ nhật

Sau đó nhấn submit để khôi phục phần được chọn bị khuyết.



Phần bị khuyết đã được khôi phục

Kéo thanh để nhận biệt thay đổi của hình ảnh.

Và nút download để tải hình ảnh về.

## **5. Kết quả đạt được**

## **5.1 Những thành quả đạt được:**

Thông qua đồ án GAN khôi phục hình ảnh:

Nhóm em đã tìm hiểu và học được kiến trúc của mạng GAN và quá trình GAN áp dụng vào bài toán khôi phục hình ảnh bị che lấp (image inpainting) cũng như củng cố về mạng neural network.

Nhóm đã học được cách deploy trang web bằng Flask , tự tạo được giao diện hình ảnh và các chức năng quan trọng để tạo trải nghiệm cơ bản , ngoài ra còn có tập SQL đơn giản để lưu trữ các thông tin đăng nhập.

## **5.2 Thảo luận:**

**Ưu điểm:**

Kết quả là chúng em đã hoàn thiện 1 trang web có chức năng image inpainting với độ phân giải hình ảnh là 32x32 pixel với đầu vào hình ảnh là hình ảnh mặt người.

Qua quá trình học tập và tìm hiểu thì chúng em đã có thể biết thêm về kiến trúc mạng GAN và quá trình GAN có thể tạo ra hình ảnh bằng cách nào , ngoài ra chúng em còn biết thêm về quá trình mà mạng GAN được sử dụng để có thể tạo ra hình ảnh phù hợp với hình ảnh đã bị khuất.

Mô hình có thể áp dụng được cho nhiều loại tệp hình ảnh khác nhau và tình huống khác nhau.

**Nhược điểm:**

Mô hình nhạy cảm với dữ liệu bị nhiễu : nên việc nhiễu hình ảnh sau khi khôi phục là điều không thể tranh khỏi.

Mô hình cần nhiều dữ liệu huấn luyện hơn và thời gian nhiều hơn để có thể chính xác hơn ,Đồ án chỉ thực hiện với 1 bộ dữ liệu nhỏ , và ảnh có độ phân giải thấp , nên khả năng khôi phục hình ảnh chỉ được khôi phục với chất lượng ảnh 32x32 pixel.

Vì kết hợp 2 mạng Local Discriminator và Global Discriminator nhóm vẫn chưa thử nghiệm việc tắt một trong hai lớp Local Discriminator và Global Discriminator thì model sẽ hoạt động như nào.Đó là thiếu xót của nhóm chúng em.

Trang web chỉ có thể khôi phục hình ảnh bị khuất theo dạng hình chữ nhật tốc độ phản hồi chưa cao ,giao diện trang web còn đơn giản.

## **5.3 Hướng phát triển**

Để cải thiện thì cần có nhiều tài nguyên hơn, cần bộ dữ liệu lớn hơn, ít bị nhiễu, cần thời gian và vật chất hiện đại hơn để quá trình huấn luyện đạt kết quả tốt hơn. Song song với đó cần có nhiều sự đầu tư hơn về mặt kĩ thuật để có thể tối ưu được kết quả tốt hơn so với hiện tại. Cần cải thiện về thuật toán cà kiến trúc mạng để có thể nâng cao được độ phân giải của hình ảnh được khôi phục, tiến tới việc hoàn thiện được quá trình khôi phục hình ảnh tốt nhất,cần kết hợp nhiều kĩ thực phân vùng hình ảnh mới để có thể tạo ra kết quả tốt nhất.

Hoàn thành thêm các chức năng web để mang tới trải nghiệm tốt cho người dùng.

# **KẾT LUẬN**

Trong đồ án này, chúng tôi đã đề xuất một phương pháp khôi phục hình ảnh bị che khuất bằng GAN image inpainting. Phương pháp này sử dụng một mạng GAN gồm hai mạng, một mạng generator và một mạng discriminator. Mạng generator được sử dụng để tạo ra các pixel mới để thay thế cho các pixel bị che khuất, trong khi mạng discriminator được sử dụng để phân biệt giữa các pixel thật và các pixel được tạo ra bởi mạng generator.

Kết quả thử nghiệm cho thấy phương pháp này có thể khôi phục hình ảnh bị che khuất một cách hiệu quả. Trong tương lai, chúng tôi sẽ tiếp tục nghiên cứu và cải thiện phương pháp này. Chúng tôi sẽ thử nghiệm với các mạng GAN khác nhau và các kỹ thuật huấn luyện khác nhau để cải thiện chất lượng của hình ảnh được khôi phục.

Phương pháp này chỉ có thể khôi phục các hình ảnh bị che khuất một phần. Phương pháp này có thể bị ảnh hưởng bởi các đặc tính của hình ảnh, chẳng hạn như độ tương phản và độ phức tạp của hình ảnh.

# **Tài liệu tham khảo**

* Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image inpainting with contextual attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5980-5988).
* Zhang, Y., Isola, P., & Efros, A. A. (2020). Self-supervised learning of image inpainting with partial convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2400-2409).
* Yu, J., Zhang, X., Wang, J., & Liu, X. (2021). Context-aware image inpainting with adaptive mask propagation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 11531-11539).