Chuyển Đổi Ảnh-phác họa Khuôn Mặt Sử Dụng GAN

1st Nguyễn Thị Thanh Hòa
Khoa Công Nghệ Thông Tin
Bộ môn Khoa học dữ liệu
thanhhoanguyen.iuh@gmail.com
MSSV: 19429041

2nd Ngô Quốc Hoàng
Khoa Công Nghệ Thông Tin
Bộ môn Khoa học dữ liệu
nqh29092001@gmail.com
MSSV: 19477071

3rd Quách Trọng Nghĩa
Khoa Công Nghệ Thông Tin
Bộ môn Khoa học dữ liệu
trongnghia110401@gmail.com
MSSV: 19502841

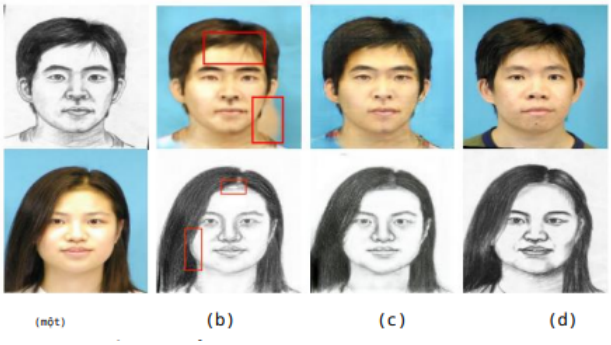
***Tóm tắt nội dung*—Chuyển đổi từ phác họa sang hình ảnh, và từ hình ảnh sang phác họa có nhiều ứng dụng. Đây vẫn là vấn đề khó khăn do ảnh và phác họa có những đặc điểm khác nhau. Trong dự án này, chúng tôi coi đây là bài toán chuyển đổi hình ảnh sang hình ảnh và sử dụng các mô hình GAN để tạo những bức ảnh thực tế có chất lượng cao từ các bản phác họa và ngược lại. Các phương pháp dựa trên GAN gần đây đã cho kết quả đầy hứa hẹn về các vấn đề dịch từ hình ảnh sang hình ảnh và đặc biệt là tổng hợp từ hình ảnh sang bản phác họa, tuy nhiên, chúng vẫn có hạn chế trong việc tạo ra hình ảnh thực tế có độ phân giải cao. Để đạt được mục tiêu này,chúng tôi sử dụng Autoencoder, Pix2Pix và CycleGAN tiến hành đào tạo trên bộ data CUHK.**

Từ khóa: Autoencoder, Pix2Pix, CycleGAN, CUHK.

I. Giới thiệu

Nghiên cứu về sinh trắc học đã được những tiến bộ đáng kể trong vài thập kỷ qua và khuôn mặt vẫn luôn là sinh trắc học được nghiên cứu phổ biến nhất vì sự thuận tiện của việc thu thập dữ liệu. Ngoài các ứng dụng trong bảo mật, tổng hợp ảnh phác họa khuôn mặt còn có một số ứng dụng trong giải trí kỹ thuật số. Ảnh phác họa cũng ngày càng trở nên phổ biến đối với người dùng điện thoại thông minh và mạng xã hội, nơi ảnh phác họa được sử dụng làm ảnh đại diện hoặc ảnh đại diện. Vì vậy, tổng hợp và ghép ảnh phác họa là những bài toán quan trọng và thiết thực. Một số công trình đã khai thác thành công CNN để thực hiện các tác vụ dịch từ hình ảnh sang hình ảnh khác nhau. Gần đây, các mô hình tạo ra như GAN và VAE, đã thành công hơn trong các nhiệm vụ như vậy do khả năng tạo ra mạnh mẽ của chúng. Đặc biệt, GAN đã đạt được kết quả ấn tượng trong việc tạo ảnh, chỉnh sửa ảnh và học biểu diễn. Khi đánh giá các phương pháp này một cách chi tiết cho nhiệm vụ của chúng tôi, nó đã phát hiện ra rằng chúng có những hạn chế trong việc tạo ra hình ảnh có độ phân giải cao hơn (như trong Hình.1).

Để tổng hợp từ ảnh thành bản phác họa và tổng hợp từ bản phác họa thành ảnh đều có các ứng dụng thực tế, chúng tôi áp dụng CycleGAN. Phương pháp được đề xuất có hai bộ tạo GA và GB lần lượt tạo bản phác họa từ ảnh và ảnh từ bản phác họa. lưu ý hai điểm khác biệt chính:



Hình 1: Mẫu kết quả tổng hợp ảnh và sketch.

1) Để giải quyết vấn đề tạo tác trong quá trình tạo ảnh có độ phân giải cao, chúng tôi đề xuất sử dụng discriminator network.

2) Trong khi CycleGAN chỉ sử dụng tổn thất tính nhất quán theo chu kỳ, chúng tôi cũng sử dụng lỗi tái tạo L1 giữa đầu ra được tạo và hình ảnh đích. Việc sử dụng các hàm mất mát bổ sung hoạt động như một quy trình chuẩn hóa trong quá trình học tập.

II. Nghiên cứu liên quan

A. Face photo-sketch synthesis

Các phương pháp dựa trên học tập không gian con liên quan đến việc sử dụng các phương pháp không gian con tuyến tính và phi tuyến tính như Principal Component Analysis (PCA) và Local Linear Embedding (LLE). Tuy nhiên giả định về ánh xạ tuyến tính là không hợp lý, [8] đề xuất một phương pháp phi tuyến tính dựa trên LLE nơi họ thực hiện tổng hợp bản phác họa dựa trên patch-based sketch synthesis. Hình ảnh ảnh đầu vào được chia thành các mảng chồng lên nhau và được chuyển đổi thành các mảng phác họa tương ứng bằng phương pháp LLE. Toàn bộ hình ảnh phác họa sau đó thu được bằng cách lấy trung bình các vùng chồng lấp giữa các mảng phác họa lân cận. Tuy nhiên, nó dẫn đến hiệu ứng mờ và bỏ qua các mối quan hệ lân cận giữa các patches và do đó không thể tận dụng cấu trúc toàn cầu. Theo một cách tiếp cận khác, một số phương pháp đã được phát triển bằng kỹ thuật phân

tích Bayes. [9], [10] Mô hình Hidden Markov (HMMs) để mô hình hóa mối quan hệ phi tuyến tính giữa bản phác họa và ảnh. [11] đề xuất kỹ thuật dựa trên Markov Random Field (MRF) để kết hợp mối quan hệ giữa các bản vá lân cận. Để cải thiện hơn, [12] phương pháp mới được đề xuất mô hình Markov weight fields (MWF) có khả năng tổng hợp các bản vá mục tiêu mới không tồn tại trong tập huấn luyện. Gần đây, [13] một phương pháp tổng hợp ảnh phác họa khuôn mặt dựa trên nhiều biểu diễn kết hợp một cách thích ứng nhiều biểu diễn để tái tạo một bản vá hình ảnh đã gửi bằng cách kết hợp nhiều tính năng từ hình ảnh khuôn mặt được xử lý bằng nhiều bộ lọc và ngoài ra Markov networks là mô hình được sử dụng để mô hình hóa mối quan hệ giữa các bản vá lân cận. Các bản vá hình ảnh ứng cử viên từ bản phác họa ước tính ban đầu và bản phác họa mẫu sau đó được chọn bằng các tính năng đa tỷ lệ. Những bản vá lỗi này có thể được tinh chỉnh và lắp ráp để có được bản phác họa cuối cùng được cải tiến hơn nữa bằng cách sử dụng cascaded regression strategy. [14] Đề xuất việc sử dụng Bayesian framework bao gồm mô hình lựa chọn hàng xóm và mô hình tính toán trọng số. Họ xem xét ràng buộc lân cận không gian giữa các mảng hình ảnh liền kề cho cả hai mô hình trái ngược với các phương pháp hiện có trong đó ràng buộc kề cận chỉ được xem xét cho một trong các mô hình. Phương pháp dựa trên CNN [15], [16] đã được đề xuất gần đây cho thấy kết quả đầy hứa hẹn, ngoài ra còn có một công trình gần đây về tổng hợp khuôn mặt từ thuộc tính khuôn mặt [17] áp dụng phác họa để tổng hợp ảnh như một giai đoạn thứ hai trong cách tiếp cận của họ.

B. Image-to-image translation

Trái ngược với các phương pháp truyền thống để tổng hợp ảnh phác họa, một số nhà nghiên cứu đã khai thác thành công của CNN để tổng hợp và nhận dạng ảnh phác họa trên nhiều miền. Tổng hợp ảnh phác họa khuôn mặt được coi là một vấn đề image-to-image translation. [18] Một phương pháp tổng hợp ảnh phác họa dựa trên mạng tích chập hoàn toàn từ đầu đến cuối được đề xuất.Một số phương pháp đã được phát triển cho các nhiệm vụ liên quan như tổng hợp phác họa chung [19], dịch ảnh biếm họa [20] và tạo hình đại diện được tham số hóa [21]. Trong công việc này, chúng tôi khám phá các kỹ thuật lập mô hình chung đã rất thành công đối với một số tác vụ dịch từ hình ảnh sang hình ảnh. GANs [22], [23] và VAEs [24], [25] là hai lớp kỹ thuật tổng quát phổ biến gần đây. GAN [26] được sử dụng để tổng hợp các hình ảnh thực tế bằng cách học phân phối các hình ảnh đào tạo. Gần đây, một số biến thể dựa trên GAN ban đầu đã được đề xuất cho các tác vụ dịch từ hình ảnh sang hình ảnh. [27] GAN có điều kiện được đề xuất cho một số tác vụ như nhãn cho cảnh đường phố, nhãn cho mặt tiền, tô màu hình ảnh, v.v. [23] CycleGAN đã đề xuất học cách dịch từ hình ảnh sang hình ảnh theo cách không giám sát. Tương tự như cách tiếp cận trên, [28] đề xuất một phương pháp không giám sát để thực hiện các tác vụ dịch thuật dựa trên dữ liệu chưa ghép nối.

III. Phương pháp tiếp cận

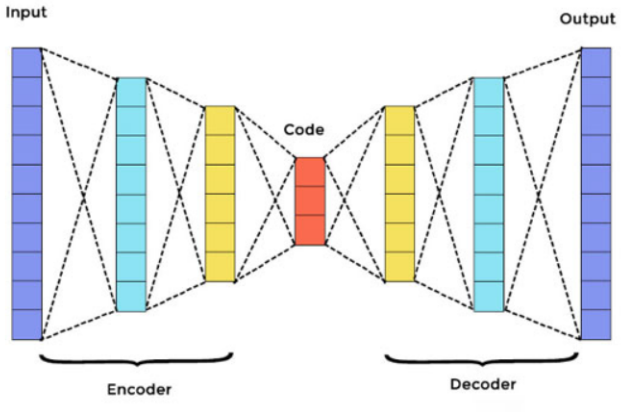
Trong dự án này, chúng tôi tiến hành đạo tạo bộ dữ liệu từ ảnh phác họa ra ảnh và từ ảnh ra phác họa, sử dụng các kiến trúc mô hình Autoencoder, pix2pix và cycleGAN.

A. Autoencoder

Autoencoder là một loại mạng neural nhân tạo được sử dụng để học các loại mã hóa dữ liệu không giám sát. Mục đích của Autoencoder là học cách biểu diễn chiều nhỏ hơn (mã hóa) cho dữ liệu có chiều cao hơn. Đây cũng là lý do mà autoencoder thường được dùng cho các bài toán giảm chiều dữ liệu hay trích xuất đặc trưng.

Ngoài ra, Autoencoder còn có thể được sử dụng với chức năng tạo ra các mô hình học tập trung (Generative learning models). Ở đây, chúng tôi sử dụng để tạo ra ảnh photo từ bản phác họa.

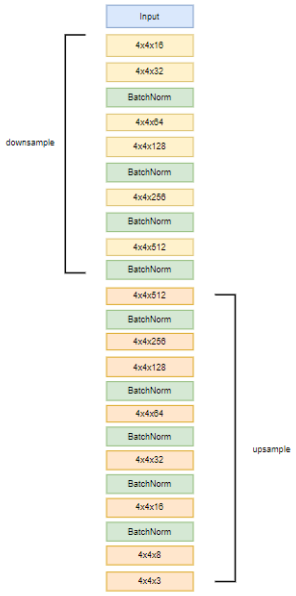
Kiến trúc của Autoencoder bao gồm 3 thành phần chính: Encoder, Bottleneck, Decoder.



Hình 2: Kiến trúc mô hình autoencoder

- Encoder: một module có chức năng nén dữ liệu đầu vào của bộ kiểm tra xác thực thành một biểu diễn được mã hóa. Thông thường nó sẽ nhỏ hơn một bài bậc so với dữ liệu đầu vào.
- Bottleneck: Một module chứa các biểu diễn tri thức đã được nén hay còn gọi là output của Encoder, đây là phần quan trọng nhất trong mạng vì nó mang đặc trưng của dữ liệu đầu vào, có thể sử dụng để lấy đặc trưng của ảnh, tái tạo hình ảnh,...
- Decoder: Module hỗ trợ mạng giải nén các biểu diễn tri thức và tái tạo lại cấu trúc dữ liệu từ dạng mã hóa của nó. Mô hình học dựa theo việc so sánh đầu ra của Decoder với đầu vào ban đầu của nó.

Mô hình này chúng tôi xây dựng bao gồm 14 lớp. Trong đó có 6 lớp Conv2d và 8 lớp Conv2dTranspose, kết hợp với batchnorm và hàm kích hoạt LeakyReLU.



Hình 3: Kiến trúc mô hình autoencoder trong bài toán

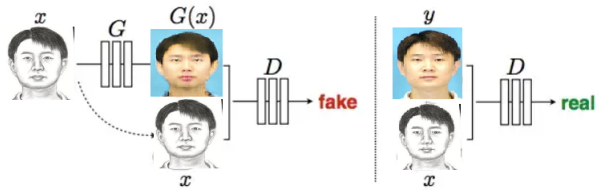
Chúng tôi sử dụng Mean Absolute Error (MAE) làm hàm lỗi và đánh giá dựa vào thang đo Accuracy và SSIM. Công thức MAE:

$$MAE = \frac{abs(\hat{y} - y)}{n} \quad (1)$$

Trong đó:

- $\hat{y}$: là kết quả dự đoán
- y: là kết quả thực
- n: số lượng mẫu

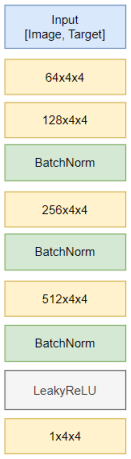
B. Pix2pix



Hình 4: Pix2Pix

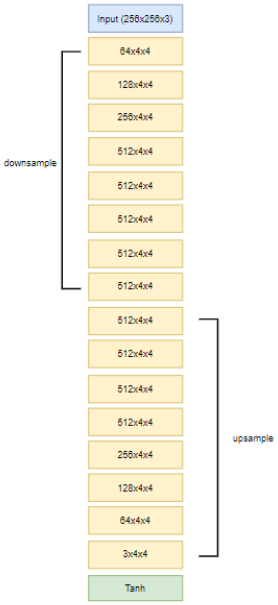
Pix2pix là một mạng GAN nên cũng có 2 phần Generator (G) để sinh ảnh fake và Discriminator(D) để phân biệt ảnh thật và ảnh fake. Tuy nhiên khác với GAN bình thường khi input của Generator là nhiễu, thì trong pix2pix của generator là ảnh. Và output cũng là ảnh.

Input của disciminator là ảnh x (input của gennerator) và G(x) (output của generator). Hai ảnh này cũng kích thước được xếp lên nhau rồi cho vào discriminator. Discriminator học bằng cách phân biệt x và G(x) là ảnh fake, x và y là ảnh thật.



Hình 5: Cấu trúc Discriminator của Pix2Pix

Ngược lại generator sẽ học bằng cách cho x và G(x) là ảnh thật.



Hình 6: Cấu trúc Generator của Pix2Pix

Generator input là ảnh và output cũng là ảnh nên kiến trúc cũng gần giống mạng U-net, dạng encoder-decoder. Kiến trúc generator chúng tôi xây dựng gồm 8 lớp Conv2d với hàm kích hoạt LeakyReLU, 8 lớp Conv2dTranspose với hàm kích hoạt ReLU. Từ layer thứ 2 trở đi chúng tôi xây dựng kết hợp với BatchNorm chuẩn hóa dữ liệu.

Generator loss sử dụng binary cross entropy kết hợp với L1(MAE) bằng tham số lambda.

Công thức Binary cross entropy:

$$BCE\_Loss = -\frac{1}{m}\sum_{i=1}^{m} y_i.log\hat{y}_i + (1 - y_i).log(1 - \hat{y}_i) \quad (2)$$

$$G\_loss = BCE\_Loss + \lambda * MAE \quad (3)$$

Trong đó:

- λ: là hệ số, thường là 100

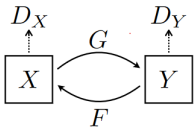
Mô hình này cũng sử dụng SSIM làm thông số đánh giá.

C. Cycle GAN

CycleGAN được thiết kế dựa trên Generative Adversarial Network (GAN). CycleGAN là một mở rộng của kiến trúc GAN cổ điển bao gồm 2 Generator và 2 Discriminator. Generator đầu tiên gọi là G, nhận đầu vào là ảnh từ domain X (ảnh đầu vào) và convert nó sang domain Y (ảnh mục tiêu. Generator còn lại gọi là Y, có nhiệm vụ convert ảnh từ domain Y sang X. Mỗi mạng Generator có 1 Discriminator tương ứng với nó:

D_Y: phân biệt ảnh lấy từ domain Y và ảnh được dịch qua G(x).

D_X: phân biệt ảnh lấy từ domain X và ảnh được dịch qua F(y).



Hình 7: Mô hình Discriminator và Generator

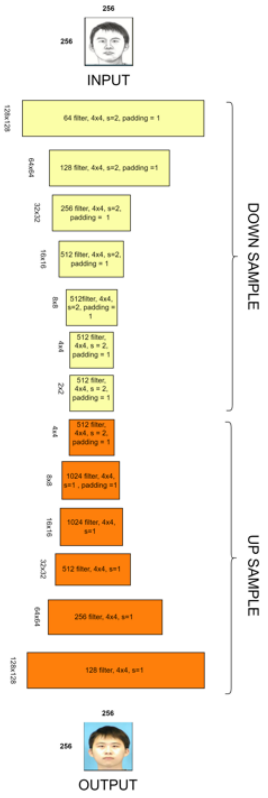
1) Generator: Mạng CycleGAN sử dụng để giải quyết các bài toán unsupervised uni-model image to image translation, để giải quyết bài toán này, nhóm chúng tôi đã xây dựng mô hình với kiến trúc CycleGAN như sau:

- Mỗi generator của chúng tôi là một mạng U-net, gồm có 2 phần là encoder và decoder. Phần encoder bao gồm 7 khối downsample với covolution có kernel 4x4,stride là 2, padding là 1, kiến trúc có dạng C64 - C128 - C256 - C512. Trong đó C là một khối downsample bao gồm một lớp Convolution, theo sau là một lớp Instance Norm và sử dụng hàm kích hoạt là Leaky ReLU.

Với stride là 2, kernel (4,4) và padding là 1 thì sau mỗi lần downsample, kích thước ảnh sẽ giảm đi 1 nửa.

Việc lựa chọn Batch norm áp dụng lên mỗi lớp có nghĩa là ta sẽ chuẩn hóa dữ liệu theo từng batch. Tuy nhiên,phương pháp này có nhược điểm nếu batchsize quá nhỏ thì sẽ dẫn tới đào tạo không hiệu quả, còn nếu batch size lớn thì có thể sẽ dẫn đến hiện tượng tràn bộ nhớ. Do đó, chúng ta có một lựa chọn khác vẫn có thể đẩy nhanh quá trình đào tạo và hội tụ của mô hình đó chính là sử dụng Instance Norm. Đối với Instance Norm, nó sẽ chuẩn hóa trên từng mẫu dữ liệu của một batch,vì vậy sẽ không phụ thuộc vào batchsize mà vẫn cho kết quả hiệu quả tương tự.

Kiến trúc một generator chi tiết như hình 8:

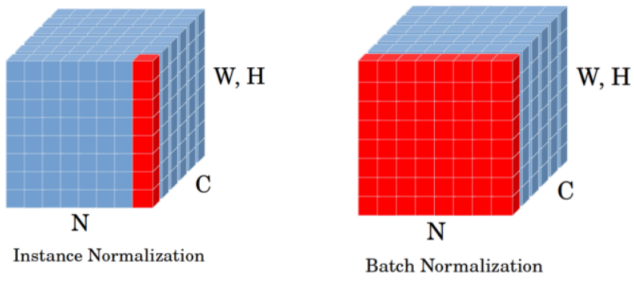


Hình 8: Cấu tạo của mỗi generator

Để chuẩn hóa cho 1 x_{ni}, ta cần tính giá trị trung bình và phương sai:

$$\mu_{ni} = \frac{1}{HW}\sum_{l=1}^{W}\sum_{m=1}^{H} x_{nilm} \quad (4)$$

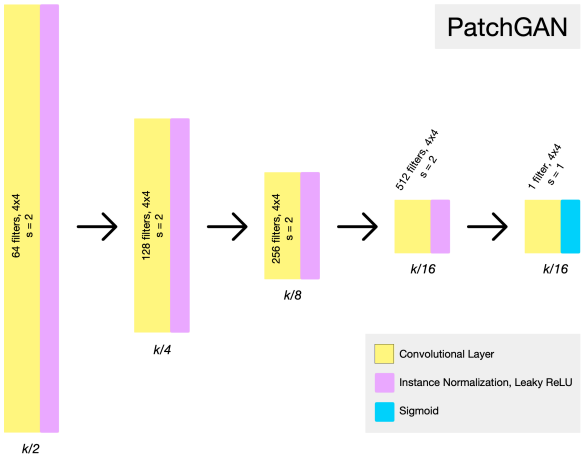
$$\sigma_{ni}^2 = \frac{1}{HW}\sum_{l=1}^{W}\sum_{m=1}^{H} (x_{nilm} - \mu_{ni})^2 \quad (5)$$



Hình 9: Sự khác nhau giữa Batch normalization và Instance normalization

Phần decoder gồm 6 khối upsamlpe với kernel 4x4 và sử dụng transpose convolution thay cho convolution, kiến trúc có dạng Cr512-Cr1024-Cr1024-Cr512-Cr128 , trong đó Cr là một khối upsamle. Chi tiết đã được mô tả ở hình 8.

2) Discriminator: Discriminator sẽ được huấn luyện để phân biệt tốt hơn hình ảnh được tạo. Trong kiến trúc này Discriminator của chúng tôi sử dụng kiến trúc PatchGAN. Kiến trúc được mô tả ở hình dưới đây:



Hình 10: Cấu tạo của Patch GAN

3) Hàm mất mát: a. Adversarial loss Trong quá trình huấn luyện, generator G cố gắng tối thiểu hóa hàm adversarial loss bằng cách translate ra ảnh $G(x)$ (với x là ảnh lấy từ domain X) sao cho giống với ảnh từ domain Y nhất, ngược lại Discriminator D_Y cố gắng cực đại hàm adversarial loss bằng cách phân biệt ảnh $G(x)$ và ảnh thật y từ domain.

$$L_{adv}(G, D_Y, X, Y) = \frac{1}{n}[logD_Y(y)] + \frac{1}{n}[log(1 - D_Y(G(x))] \tag{6}$$

Adversarial loss được áp dụng tương tự đối với generator F và Discriminator:

$$L_{adv}(F, D_X, Y, X) = \frac{1}{n}[logD_X(x)] + \frac{1}{n}[log(1 - D_X(F(y))] \tag{7}$$

b. Cycle consistency loss

Chỉ riêng adversarial loss là không đủ để mô hình cho ra kết quả tốt. Nó sẽ lai generator theo hướng tạo ra được ảnh output bất kỳ trong domain mục tiêu chứ không phải output mong muốn. Để giải quyết vấn đề này, cycle consistency loss được giới thiệu:

$$L_{cycle}(G, F) = \frac{1}{n}\sum |F(G(x_i)) - x_i| + |G(F(y + i)) - y_i| \tag{8}$$

Công thức tổng quát:

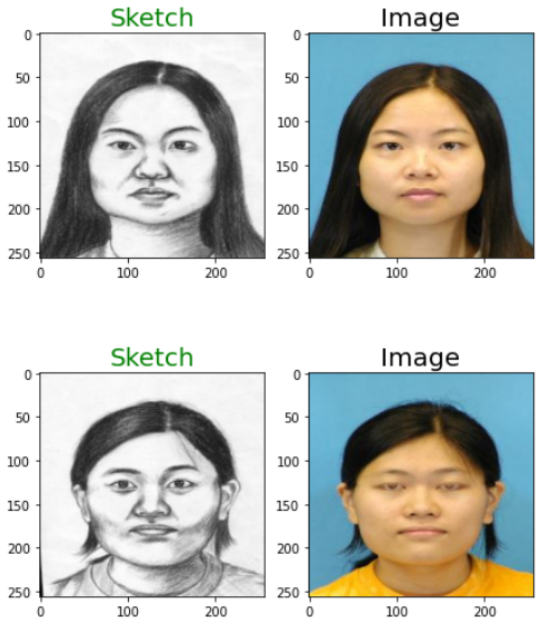
$$L = L_{adv}(G, D_Y, X, Y) + L_{adv}(F, D_X, Y, X) + \lambda L_{cycle}(G, F) \tag{9}$$

IV. Dữ liệu và đặc trưng

Phương pháp được đánh giá trên bộ dữ liệu bản phác họa đã có. Bộ dữ liệu phác họa khuôn mặt CUHK bao gồm 188 cặp phác họa - ảnh khuôn mặt từ cơ sở dữ liệu sinh viên Đại học Hồng Kông Trung Quốc có kích thước 256x256x3.

Với mỗi khuôn mặt, chúng tôi sử dụng phương pháp lật và xoay 90 độ. Với 188 khuôn mặt, sinh ra được 1504 cặp phác họa - ảnh.

Chúng tôi chia tập dữ liệu với tỉ lệ 8:2 cho tập đào tạo và tập kiểm thử để tiến hành đưa vào mô hình.



Hình 11: Hình ảnh từ bộ dữ liệu

V. Thực nghiệm

A. Autoencoder

Với mô hình Autoencoder, chúng tôi huấn luyện với các tham số epoch: 150, learning rate: 0.001

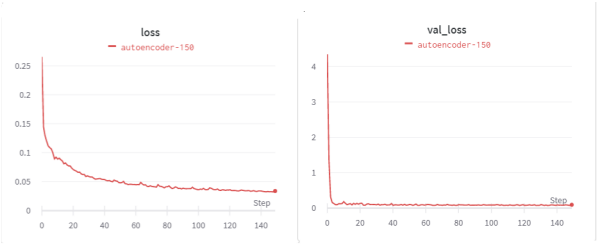
Kết quả thu được:

- Accuracy train: 0.92
- Accuracy test: 0.89
- Loss train: 0.0339
- Loss test: 0.0941

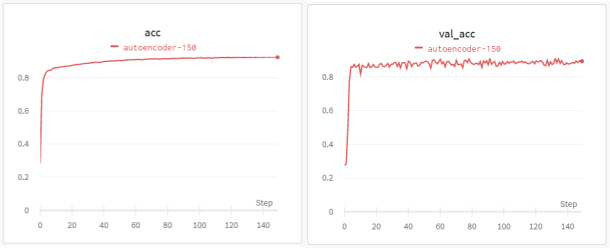
B. Pix2Pix

Mô hình này chúng tôi huấn luyện với 50 epoch và huấn luyện trên từng batch và thu được kết quả như hình 16:

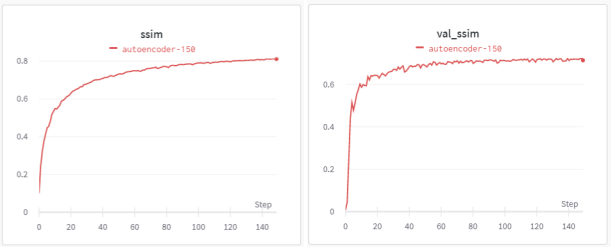
Hình ảnh hàm mất mát của G và D trong quá trình huấn luyện được mô tả ở hình 17 và 18



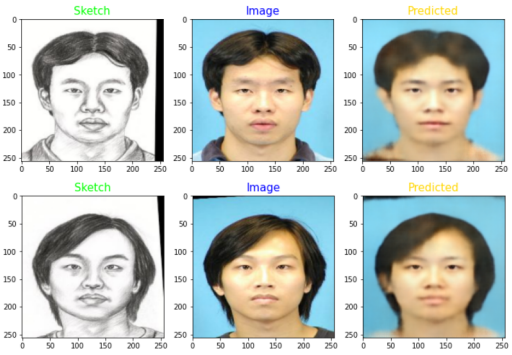
Hình 12: Loss autoencoder



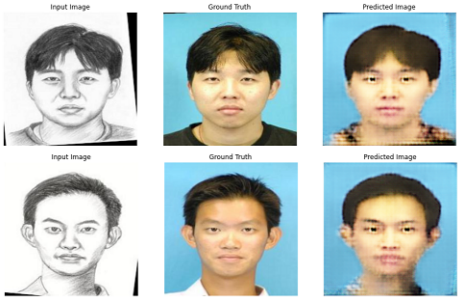
Hình 13: Accuracy autoencoder



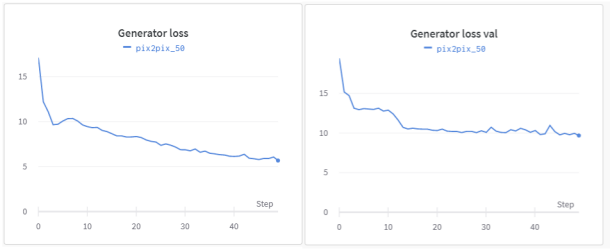
Hình 14: SSIM autoencoder



Hình 15: Kết quả của mô hình autoencoder



Hình 16: Kết quả predict Pix2Pix



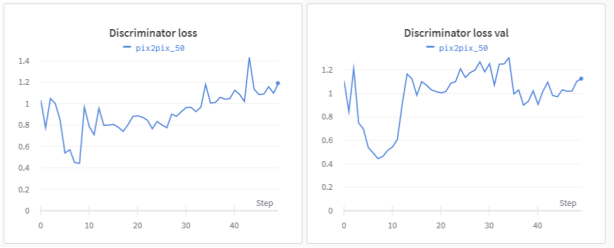
Hình 17: Generator Loss của Pix2Pix

C. *CycleGAN*

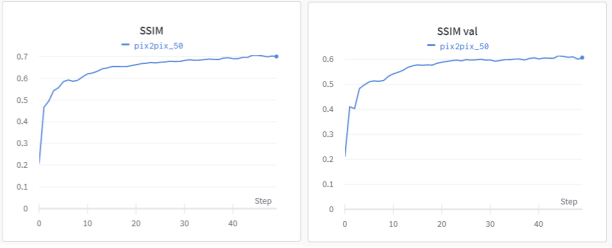
Đối với mô hình này chúng tôi thực hiện huấn luyện với 50 epoch và batch size bằng 1, learning rate của G và D đều bằng 0.0002, kết quả thu được được trình bày ở hình 20:

Biểu đồ thể hiện loss của G và D của mô hình CycleGAN được trình bày ở hình 21

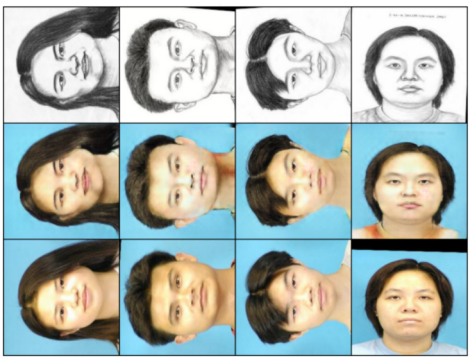
Biểu đồ thể hiện đánh giá ảnh theo thang đo SSIM của train và validate được thể hiện ở hình 22



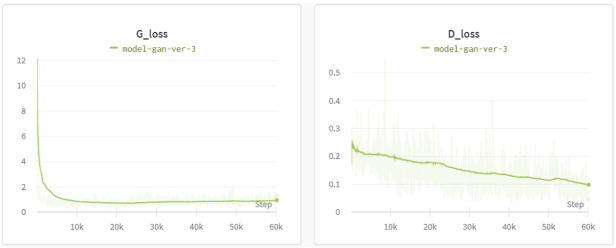
Hình 18: Discriminator Loss của Pix2Pix



Hình 19: SSIM của Pix2Pix



Hình 20: Kết quả dự đoán ảnh sử dụng CycleGAN



Hình 21: loss G và D theo thời gian của CycleGAN trong quá trình huấn luyện

D. Bảng so sánh kết quả giữa 3 mô hình

Thông số	Autoencoder	Pix2Pix	CycleGAN
D_loss		1.1917	0.046
D_loss val		1.1264	
G_loss		5.6730	0.9478
G_loss val		9.6808	
SSIM	0.8118	0.7002	0.1698
SSIM val	0.7135	0.6064	0.1624

Bảng I: Kết quả độ của các mô hình

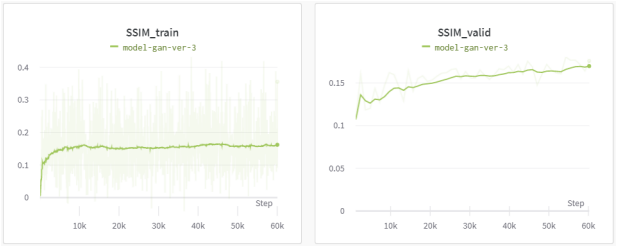
Bảng kết quả của bài báo đối chiếu:

Thông số	Pix2Pix	CycleGAN
SSIM	0.6606	0.7626

Bảng II: Kết quả SSIM trên bài báo gốc

VI. Kết luận

Việc chuyển đổi phác họa - ảnh khuôn mặt giờ đây đã là một vấn đề giúp ích trong việc nhận dạng các đối tượng tình nghi, việc chuyển đổi từ ảnh qua phác họa cũng góp phần vào việc giải trí khi nhiều người muốn biết khuôn mặt mình khi chuyển qua ảnh phác họa sẽ như thế nào. Trong đề tài này, chúng tôi sử dụng các mô hình như Autoencoder, Pix2Pix, CycleGAN và cho kết quả khả quan. Bằng việc đánh giá và so sánh giữa các mô hình, chúng tôi nhận thấy mô hình CycleGAN cho kết quả tạo ảnh tốt nhất.



Hình 22: Kết quả đánh giá tập train và validate sử dụng thang đo SSIM

VII. Lời cảm ơn

Chúng tôi chân thành cảm ơn quý thầy bộ môn đã tận tình giúp đỡ và hướng dẫn chúng tôi hoàn thành đề tài. Cảm ơn các thành viên đã cùng nhau đóng góp và hoàn thiện đề tài. Do kinh nghiệm của các thành viên nhóm còn hạn chế nên không thể tránh khỏi sai sót, rất mong sẽ nhận được những lời nhận xét và góp ý đến từ các thầy và các bạn để bài có thể hoàn thiện hơn. Xin chân thành cảm ơn!

Tài liệu

[1] S. Bell and K. Bala. Learning visual similarity for product design with convolutional neural networks. ACM Transactions on Graphics (TOG), 34(4):98, 2015.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[3] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[4] G. W. Taylor, I. Spiro, C. Bregler, and R. Fergus. Learning invariance through imitation. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 2729–2736. IEEE, 2011

[5] G. Wang, D. Hoiem, and D. Forsyth. Learning image similarity from flickr groups using stochastic intersection kernel machines. In Computer Vision, 2009 IEEE 12th International Conference on, pages 428–435. IEEE, 2009.

[6] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1386–1393, 2014.

[7] https://www.baeldung.com/cs/instance-vs-batch-normalization

[8] Q. Liu, X. Tang, H. Jin, H. Lu, and S. Ma. A nonlinear approach for face sketch synthesis and recognition. In CVPR 2005, volume 1, pages 1005–1010. IEEE, 2005.

[9] X. Gao, J. Zhong, J. Li, and C. Tian. Face sketch synthesis algorithm based on e-hmm and selective ensemble. IEEE CSVT, 18(4):487–496, 2008.

[10] B. Xiao, X. Gao, D. Tao, and X. Li. A new approach for face recognition by sketches in photos. Signal Processing, 89(8):1576–1588, 2009.

[11] X. Wang and X. Tang. Face photo-sketch synthesis and recognition. TPAMI, 31(11):1955–1967, 2009.

[12] H. Zhou, Z. Kuang, and K.-Y. K. Wong. Markov weight fields for face sketch synthesis. In CVPR, pages 1091–1097. IEEE, 2012.

[13] C. Peng, X. Gao, N. Wang, D. Tao, X.Li, and J. Li.Multiple representations-based face sketch-photo synthesis. IEEE NNLS, 27(11):2201–2215, 2016.

[14] N. Wang, X. Gao, L. Sun, and J. Li. Bayesian face sketch synthesis. IEEE TIP, 26(3):1264–1274, 2017.

[15] F. Gao, S. Shi, J. Yu, and Q. Huang. Composition-aided sketch-realistic portrait generation. arXiv:1712.00899, 2017.

[16] C. Chen, X. Tax, and K. Wong. Face sketch synthesis with style transfer using pyramid column feature. In IEEE WACV, 2018.

[17] X. Di and V. M. Patel. Face synthesis from visual attributes via sketch using conditional vaes and gans. arXiv:1801.00077, 2017.

[18] L. Zhang, L. Lin, X. Wu, S. Ding, and L. Zhang. End-to end photo-sketch generation via fully convolutional representation learning. In ACM ICMR, pages 627–634. ACM, 2015.
[19] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays. Scribbler: Controlling deep image synthesis with sketch and color. arXiv:1612.00835, 2016.
[20] Z. Zheng, H. Zheng, Z. Yu, Z. Gu, and B. Zheng. Photo-to-caricature translation on faces in the wild. arXiv:1711.10735, 2017.
[21] L. Wolf, Y. Taigman, and A. Polyak. Unsupervised creation of parameterized avatars. arXiv:1704.05693, 2017.
[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in NIPS, pages 2672–2680, 2014.
[23] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. IEEE ICCV, 2017.
[24] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic back-propagation and approximate inference in deep generative models. arXiv:1401.4082, 2014
[25] D. P. Kingma and M. Welling. Auto-encoding variational bayes. arXiv:1312.6114, 2013.
[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in NIPS, pages 2672–2680, 2014.
[27] . Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. arXiv:1611.07004, 2016.
[28] Z. Yi, H. Zhang, P. T. Gong, et al. Dualgan: Unsupervised dual learning for image-to-image translation. IEEE ICCV, 2017.