Chuyển ảnh chân dung thành tranh cổ điển sử dụng CycleGAN*

1st Văn Sỹ
Trường đại học Công nghiệp thành phố Hồ Chí Minh
MSSV: 19495751

2nd Lê Mỹ Thanh Lành
Trường đại học Công nghiệp thành phố Hồ Chí Minh
MSSV: 19525231

3rd Lê Duy Tâm
Trường đại học Công nghiệp thành phố Hồ Chí Minh
MSSV: 19445631

4th Dương Văn Sang
Trường đại học Công nghiệp thành phố Hồ Chí Minh
MSSV: 19495031

***Tóm tắt nội dung*—Sử dụng mô hình CycleGAN[1] để sinh các bức ảnh chân dung mang phong cách hội hoạ cổ điển với đầu vào là một ảnh chân dung người thật. Bức ảnh sinh ra vừa mang các đặc trưng của ảnh đầu vào cũng như phải mang phong cách hội hoạ cổ điển như màu sắc, nét vẽ. Bộ dữ liệu được sử dụng bao gồm 1000 ảnh chân dung từ tập dữ liệu Real and Fake face detection của đại học Yonsai và 1000 tranh cổ điển được tìm và sử lý trên GettyImages. Mô hình mang lại kết quả ở mức khá, ảnh sinh ra mang nhiều đặc trưng của ảnh đầu vào và thay đổi màu sắc cũng như đường nét giống với các bức tranh cổ điển.**

I. Giới thiệu

Sự cách điệu của các tranh chân dung từ lâu đã là một thách thức với cộng đồng nghiên cứu Non-photorealistic rendering (NPR). Và sự xuất hiện của các mạng GAN gần đây đã trở thành kỹ thuật được yêu thích cho các nhiệm vụ Image to Image translation. [2]Nhưng nhược điểm chính là nó đòi hỏi một lượng dữ liệu lớn các bức tranh và ảnh kỹ thuật số được ghép đôi riêng cho nhau.

Và CycleGAN đã mang lại một bước đột phá trong đạo tạo hình ảnh không được ghép đôi, nó có thể tạo ra kết quả ấn tượng trên các bộ dữ liệu tương đối nhỏ (từ 200 đến 1000 ảnh là đủ)[1]. Nhưng nó vẫn có điểm bất cập đó chính là gặp khó khăn trong việc tái tạo lại các chi tiết có tần số xuất hiện cao để duy trì độ trung thực cũng như phong cách hội hoạ mà nghệ sĩ đã sử dụng.

Đầu vào cho mô hình là một bức chân dung kỹ thuật số của một người thực, sau đó sử dụng CycleGAN để xuất ra một bức chân dung cách điệu theo phong cách tranh cổ điển.

II. Chuẩn bị dữ liệu

Mô hình CycleGAN yêu cầu hai bộ dữ liệu: một bộ ảnh kỹ thuật số về khuôn mặt người thật, một bộ các bức tranh chân dung cổ điển.

Đối với bộ dữ liệu đầu tiên, chúng tôi chọn bộ dữ liệu của đại học Yonsai là "Real and Fake detection"[4] với 1000 ảnh ở độ phân giải 600x600. Đối với bộ dữ liệu thứ hai, chúng tôi đã sử dụng công cụ quét web để thu thập 1300 ảnh trên GettyImages, sau đó xử lý thủ công để cắt ảnh theo đúng yêu cầu cũng như loại những ảnh sai phong cách(thu lại được 1000 ảnh).

Tất cả ảnh trước khi được chuyển về kích thước 256x256 trước khi đi vào quá trình huấn luyện.

III. Phương pháp

A. Sơ lược về CycleGan

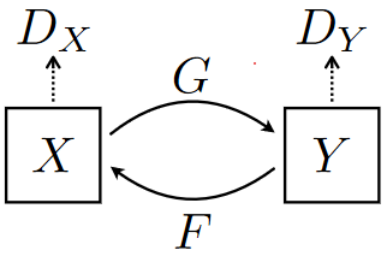
CycleGAN được thiết kế dựa trên Generative Adversarual Network (GAN). Kiến trúc GAN là một cách tiếp cận để huấn luyện một mô hình sinh ảnh bao gồm hai mạng neural: một mạng generator và mạng discriminator. Generator sử dụng một vector ngẫu nhiên lấy từ dữ liệu nén (latent space) làm đầu vào và tạo ra hình ảnh mới và sửa dụng Discrinator lấy một bức ảnh làm đầu vào và dự đoán xem nó là thật (lấy từ bộ dữ liệu) hay giả (được tạo ra bởi Generator). Cả hai mô hình sẽ thi đấu với nhâu, Generator sẽ được huấn luyện để sinh ảnh có thể đánh lừa Discriminator và Discriminator sẽ được huấn luyện để phân biệt tốt hơn hình ảnh được tạo.

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 và biến đổi nó sang domain Y. Generator còn lại gọi là F, có nhiệm vụ biến đổi ảnh từ domain Y sang X. Mỗi mạng Generator có 1 Discriminator tương ứng với nó. (Hình 1)

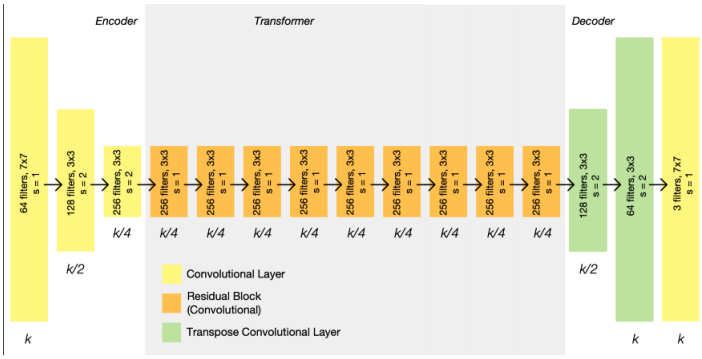
- D_Y : phân biệt ảnh lấy từ domain Y và ảnh được translate G(x).
- D_X : phân biệt ảnh lấy từ domain X và ảnh được translate F(y)

1) Generator: Generator trong bài được chia làm ba phần: encoder , transformer và decoder.

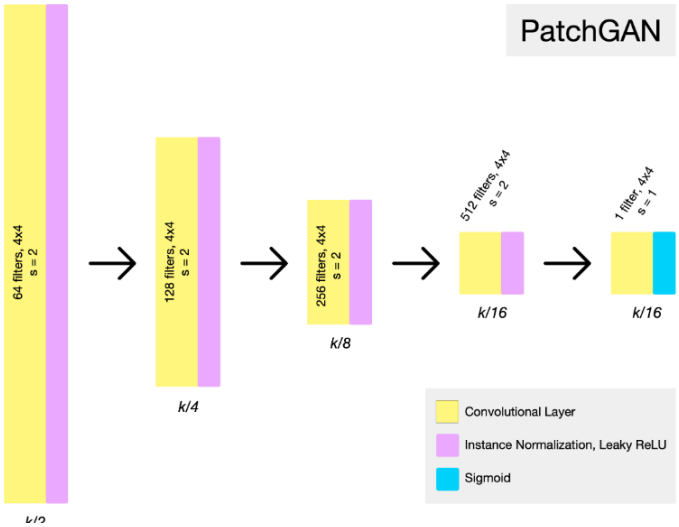
Phần encoder bao gồm 3 lớp tích chập, 2 lớp sau có stride = 2 để làm giảm kích thước đầu vào của ảnh và tăng số channel. Output của encoder được sử dụng làm đầu vào cho transformer bao gồm 9 khối residual như trong resnet.



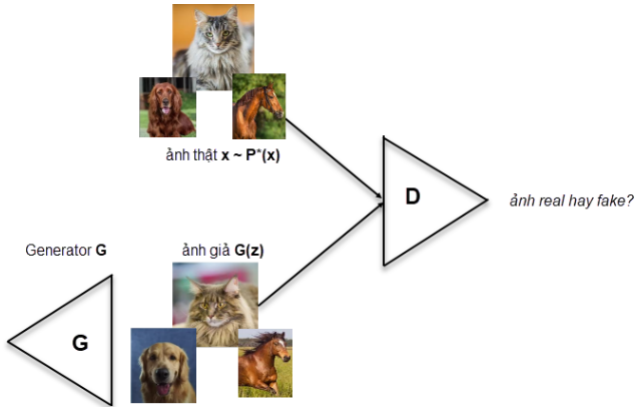
Hình 1. CycleGAN



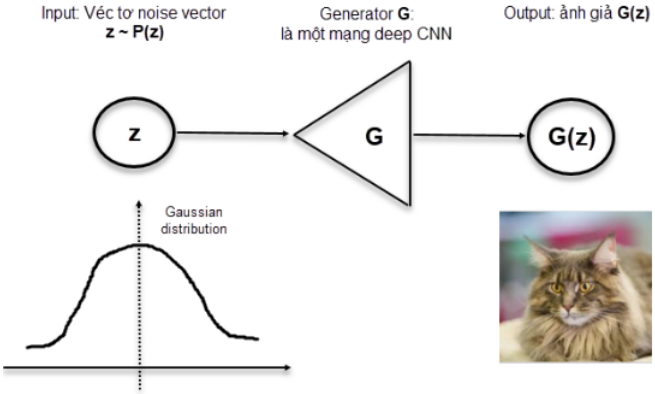
Hình 2. Kiến trúc của Generator



Hình 3. Kiến trúc của Discriminator



Lớp batch normalization trong khối residual được thay bằng instance normalization. Cuối cùng phần decoder bao gồm 3 lớp transposed convolution sẽ biến đổi ảnh về kích thước ban đầu và số channel phụ thuộc vào domain đầu ra.



2) Discriminator: Discriminator sử dụng kiến trúc PatchGAN. Thông thường trong bài toán classification, output của mạng sẽ là một giá trị scalar - xác suất thuộc class nào đó. Trong mô hình CycleGAN, tác giả thiết kế Discriminator sao cho output của nó là một feature map NxNx1. Có thể xem là Discriminator sẽ chia ảnh đầu vào thành 1 lưới N×N và giá trị tại mỗi vùng trên lưới sẽ là xác suất để vùng tương ứng trên ảnh là thật hay giả.

3) Phân biệt Generator Discriminator: Các mô hình Machine Learning có thể được phân chia thành Discriminative và Generative. Đây chỉ là một cách phân chia trong vô số các cách phân chia khác như: mô hình học có giám sát (supervised learning)/học không giám sát (unsupervised learning), mô hình tham số (parametric)/mô hình phi tham số (non parametric), mô hình đồ thị (graphic)/mô hình phi đồ thị (non-graphic)

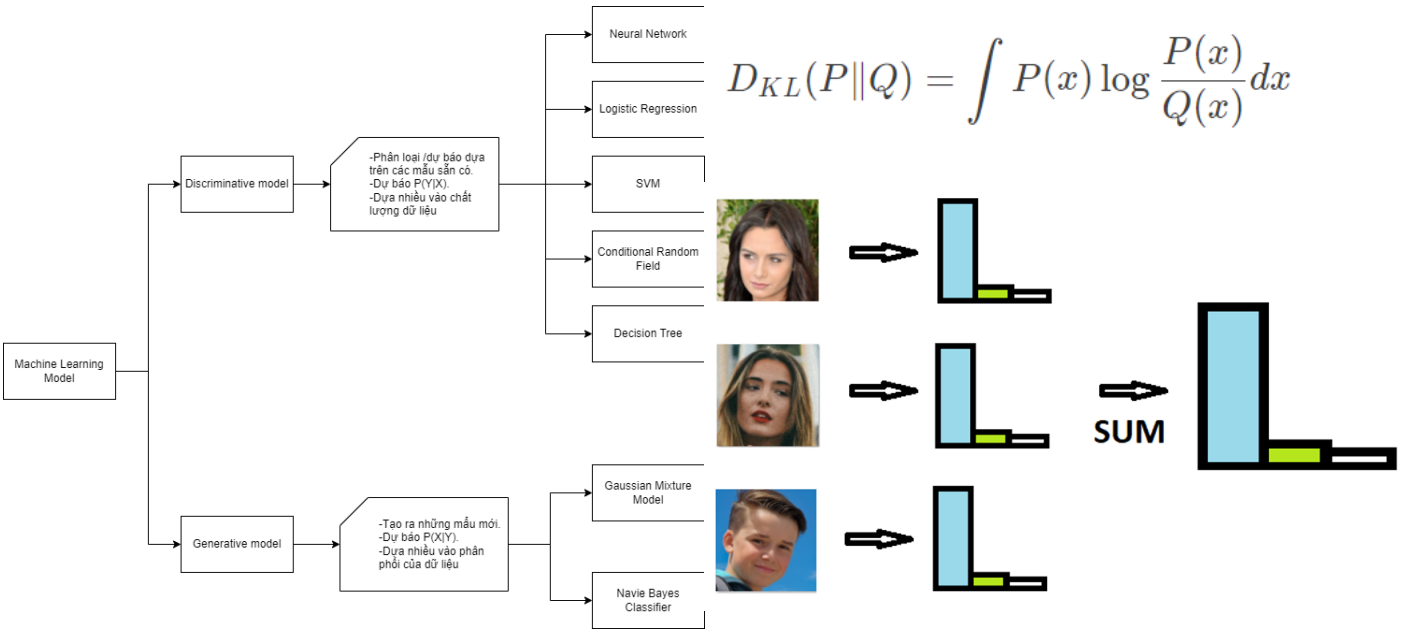
B. Hàm tổn thất

1) 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 Dy cố gắng cực đại hàm adversarial loss bằng cách phân biệt ảnh G(x) và ảnh thật y từ domain

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

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

$$L_{adv}(F, D_y, X, Y) = \frac{1}{n}[\log D_x(x)] + \frac{1}{n}[\log(1 - D_x(F(y)))]$$



Hình 4. Ví dụ về Label distribution và Marginal distribution.

2) 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. Ví dụ với bài toán biến ngựa vằn thành ngựa thường, generator có thể biến con ngựa vằn thành 1 con ngựa thường rất đẹp nhưng lại không có đặc điểm nào liên quan tới con ngựa vằn ban đầu. Để giải quyết vấn đề này, cycle consistency loss được giới thiệu. Trong paper, tác giả cho rằng nếu ảnh x từ domain X được translate sang domain Y và sau đó translate ngược lại về domain Y lần lượt bằng 2 generator G, F thì ta sẽ được ảnh x ban đầu: x$\rightarrow G(x) \rightarrow F(G(x)) \approx x$

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

3) Full loss: .

$L = L_{adv}(G, D_y, x, y) + L_{adv}(F, D_x, X, Y) + \lambda L_{cycle}(G, F)$

Trong đó λ là siêu tham số và được chọn là 10

IV. Thí nghiệm

A. Phương pháp đánh giá

Chúng tôi sử dụng hai chỉ số định lượng khác nhau và đánh giá của con người để đánh giá các mô hình của chúng tôi

1) IS: Để đánh giá ảnh sinh ra bởi Generator ta có thể dùng Inception Score[5]:

- Chất lượng ảnh: thông qua phân phối xác suất của một ảnh (Label distribution).
- Độ đa dạng: thông qua tổng xác suất của tất cả các ảnh (Marginal distribution).

Để đưa về dạng con số, ta dùng Kullback–Leibler divergence (KLD), nó nhận 2 phân phối làm đầu vào và trả về độ tương đồng giữa hai phân phối đó. Nếu 2 phân phối càng giống thì KL sẽ tiến về 0, ngược lại thì KL càng lớn.

Vì generator bài toán này chỉnh sinh ra một lớp nên phân phối của từng ảnh và phân phối tổng có dạng giống nhau nên IS sẽ thấp và chỉ mang ý nghĩa đánh giá về chất lượng ảnh được sinh ra.

2) FID: Chúng tôi sử dụng Fréchet Inception Distance[6] để đánh giá ảnh sinh ra bởi GAN và ảnh thực bằng cách tính khoảng cách giữa 2 multivariate gaussian distribution từ mạng Inception-V3 pool3. Mỗi ảnh trong tập ảnh thực(r) và tập ảnh sinh ra(g) sau khi qua mạng Inception-V3 pool3 ta sẽ được một 1 vector 2048*1,ta tìm được một multivariate gaussian distribution phù hợp nhất với các vector trong mỗi bộ ảnh với mean = $\mu(r)$,std = $\sum(r)$ mean = $\mu(g)$,std = $\sum(g)$.
Khoảng cách giữa 2 multivariate gaussian distribution:

$$\mathbf{FID} = ||\mu_r - \mu_g||^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2})$$

Trong đó:

- Tr(A) là tổng của các phần tử dọc theo đường chéo chính của ma trận vuông A.

Nhận xét: FID không âm và càng thấp thì càng tốt vì nó càng thấp thì nó biểu hiệu ảnh sinh ra càng giống ảnh thực.

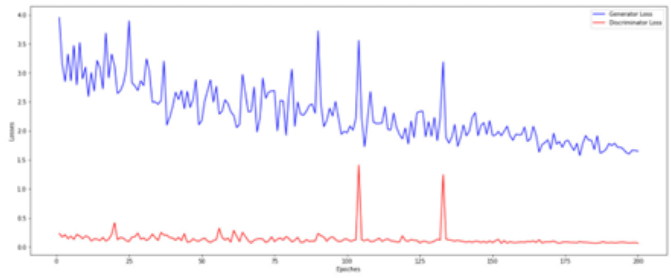
B. Thông số chi tiết huấn luyện

Tiến hành training với 200 epochs, tối ưu hoá với Adam optimizer, và sử dụng linear learning rate decays. Learning rate sẽ giữa nguyên trong 100 epochs đầu và tiến dần về 0 ở 100 epochs cuối. Các tham số được khai báo trước:

- $lr = 0.0002$
- $b1 = 0.5$
- $b2 = 0.996$

C. Kết quả

Learning rate	No. of layers	FID	FID (paper)	IS	IS (paper)
0.0002	5	54.032	121.469	2.86012	3.02852



Hình 5. Losses

Hình 6. Một số kết quả.

Việc IS thấp là vì các bức tranh có độ đa dạng các lớp thấp (so với 1000 lớp của mạng Inception). Nhưng nó vẫn không quá gần giá trị 0, nghĩa là chất lượng hình ảnh vẫn chưa tốt vì các phân phối xác suất trong việc phân loại hình ảnh không ở một lớp. Kết quả cho ra IS thấp hơn bài báo tham khảo -> chất lượng ảnh tốt hơn (vì resize ảnh ở bài làm là 256 sẽ giữ được nhiều thông tin hơn so với 240 của bài báo tham khảo và 256x256 cũng là input yêu cầu của CycleGAN).

FID thấp hơn nhiều so với bài tham khảo ở cùng các tham số huấn luyện, cho thấy các bức ảnh được sinh ra có nhiều đặc trưng giống với dữ liệu chân dung gốc -> Không được tốt vì mong muốn ảnh sinh ra phải mang cả 2 đặc trưng từ 2 tập dữ liệu. Ảnh sinh ra mang màu sắc cổ điển và mang số ít các đặc trưng của các bức tranh hội hoạ, nhưng nhìn chung vẫn giống ảnh chân dung cũ. -> Đây là vấn đề cần cải thiện.

V. Kết luận

Mục tiêu tạo ra các bức ảnh mang phong các tranh cổ điển từ ảnh chân dung thực tế. Model CycleGAN đã được sử dụng, với các tham số được tinh chỉnh và đặt lịch trình giảm learning rate. Bộ dữ liệu gồm hai phần: các bức ảnh chân dung thực tế của Đại học Yonsai, các bức tranh cổ điển được nhóm thu thập và xử lý. Mặc dù có kết quả khả quan nhưng vẫn chưa được như kì vọng, trong tương lai nhóm đề xuất cải thiện hiệu quả mô hình bằng cách tinh chỉnh các tham số, xem xét hai phương pháp bổ sung là FaceNet và SSIM, hy vọng mô hình có thể cải thiện cũng như đem lại kết quả tích cực.

Tài liệu

[1] junyanz/pytorch-CycleGAN-and-pix2pix. https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix.

[2] D. Futschik, M. Chai, C. Cao, C. Ma, A. Stoliar, S. Korolev, S. Tulyakov, M. Kucera, and D. Sýkora. Real-time-patch-based stylization of portraits using generative adversarial network. In Proceedings of the 8th ACM/Eurographics Expressive Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering, Expressive '19, page 33–42, Goslar, DEU, 2019. Eurographics Association.

[3] Xiaohan Jin, Ye Qi, and Shangxuan Wu. Cyclegan face-off. CoRR, abs/1712.03451, 2017.

[4] https://www.kaggle.com/ciplab/real-and-fake-face-detection.

[5] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans, 2016.

[6] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2017.

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980,2014.