Chào tất cả các bạn! Hôm nay nhóm mình xin trình bày về Pix2pix. Nó là một biến thể của mạng Generative Adversarial Network(GAN).

Phần 1: Trình Bày Ý Tưởng và Ứng Dụng

-Ý tưởng chính của pix2pix: sử dụng mô hình GAN (Generative Adversarial Networks) để thực hiện chuyển đổi hình ảnh từ một miền dữ liệu sang miền dữ liệu khác thông qua quá trình huấn luyện trên cặp dữ liệu đào tạo gồm hình ảnh đầu vào và hình ảnh mục tiêu tương ứng.

- Ứng dụng của Pix2pix rất đa dạng:

* Chuyển đổi hình ảnh từ đen trắng sang màu sắc: Pix2pix có thể tự động thêm màu sắc vào hình ảnh đen trắng, tạo ra những bức tranh sắc nét và sống động hơn.
* Chuyển đổi từ sketch thành hình ảnh đầy đủ: Nó có thể chuyển đổi các bản vẽ tay đơn giản thành hình ảnh thực tế, hữu ích trong lĩnh vực thiết kế và nghệ thuật số.
* Chuyển đổi hình ảnh ngày thành hình ảnh đêm: Điều này có thể hữu ích trong việc tạo ra dữ liệu mô phỏng hoặc trong việc hiển thị các biến đổi không gian thời gian trong ảnh.
* Chuyển đổi kiểu vật liệu hoặc cấu trúc: Ví dụ, từ hình ảnh vật liệu gỗ sang hình ảnh vật liệu đá, hoặc từ kiến trúc cổ điển sang kiến trúc hiện đại.
* Ứng dụng trong y học và khoa học: Pix2pix cũng có thể được áp dụng trong việc xử lý hình ảnh y khoa, chẳng hạn như chuyển đổi hình ảnh từ MRI sang hình ảnh CT để hỗ trợ chẩn đoán và phân tích y học.

Thuật toán này đã mang lại những tiến bộ đáng kể trong việc tạo ra hình ảnh chất lượng cao từ dữ liệu nguồn thô và đã được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau.

Phần 2: Trình Bày Model

Pix2pix là 1 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 những mạng GAN bình thường khi input của generator là noise, thì trong pix2pix input của generator là ảnh vẽ. Và output của generator là ảnh đủ màu sắc.

Input của discriminator là ảnh x (input của generator) và G(x) (output của generator). 2 ảnh này cùng kích thước được xếp lên nhau rồi cho vào discriminator.

Mọi người nhìn ảnh trên rất rõ, discriminator học bằng cách phân biệt x và G(x) là ảnh fake, x và y là ảnh thật. Ngược lại generator sẽ học bằng cách cho x và G(x) là ảnh thật.

Phần 3 Generator

Generator input là ảnh vẽ output là ảnh màu nên kiến trúc cũng gần gần giống với U-Net, dạng encoder-decoder.

Generator pix2pix, nguồn.

Ngoài ra còn có thể sử dụng ý tưởng của resnet (densenet) để cộng (nối) các layer tương ứng ở encoder với decode.

Phần 4 Discriminator

Discriminator sẽ có input là 2 ảnh: 1 là input của generator và 2 là ảnh output của generator hoặc ảnh output thật, (input của generator , output của generator), hoặc ( input của generator, ảnh output thật ). Điểm khác ở đây là input của discriminator còn có cả ảnh input đầu vào nữa. Vì mạng GAN thông thường input là noise, còn pix2pix input là ảnh vẽ nên còn được gọi là condition GAN (cGAN), tức là ảnh output ra phụ thuộc vào ảnh input đầu vào. Việc cho cả ảnh input vào giúp discriminator dễ phân biệt hơn ảnh nào là generator sinh ra và ảnh nào trong dataset.

Mô hình discriminator, nguồn.

2 ảnh đầu vào sẽ được nối với nhau sau đó thực hiện các lớp trong ConvNet và ra output 30\*30\*1, khác với discriminator trong GAN mình biết output 1 node và là bài toán binary classification nhỉ? Cách thiết kế mạng discrimintor như vậy gọi là PatchGAN

Phần 5: PatchGAN

PatchGAN là một loại kiến trúc mạng thường được sử dụng trong các mô hình GAN (Generative Adversarial Networks). Đặc điểm chính của PatchGAN là nó không đánh giá toàn bộ hình ảnh mà thay vào đó tập trung vào từng "patch" (miếng) nhỏ trong hình ảnh.

Thay vì đưa ra một đánh giá toàn cục về việc hình ảnh được tạo ra là thực tế hay giả mạo, PatchGAN chia hình ảnh thành các patch nhỏ hơn và đánh giá mỗi patch riêng lẻ. Điều này giúp mô hình tập trung vào việc tạo ra các chi tiết cục bộ chân thực hơn trong quá trình huấn luyện.

Phần 6: Trình Bày Về Cách Tính Hàm Loss

Loss function của pix2pix giống với GAN bình thường. Trong paper gốc tác giả sử dụng LSGAN. Bên cạnh đó tác giả còn sử dụng L1 giữa output của Generator và otuput thật trong dataset.

Pix2pix loss function, nguồn.

Ta có thể thấy là GAN loss học được các chi tiết khung ảnh nhiều hơn còn L1 sẽ học được các chi tiết nhỏ (style) của ảnh. 2 loss kết hợp với nhau sẽ cho ra kết quả tốt hơn.

Phần 7: Code

- Dataset: Facades(400 image train, 100 image val, 106 image test)

- Mổi ảnh có kích thước 512\*256\*3 tách thành ảnh thật và ảnh đầu vào của Gennerator 256\*256\*3. Chúng được:

+ resize: (256\*256\*3) -> (286\*286\*3)

+ random\_crop: (286\*286\*3) -> (256\*256\*3) (cắt cùng 1 vùng)

+ flip\_left\_right với sắc xuất là 50%

- Training: steps = 40000, batch\_size = 1

=> epochs = 100

- Pix2pix tính loss tương tự như mạng Gan nhưng khác ở:

- generator\_loss = gan\_loss + (LAMBDA \* l1\_loss) với LAMBDA = 100