**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**BÁO CÁO MÔN DEEP LEARING**

**CONDITION GANS**

**GVHD : TS. NGUYỄN THIÊN BẢO**

**SVTH : Lỷ Say Hội**

**Phạm Tuấn Anh**

**Lê Minh Cảnh**

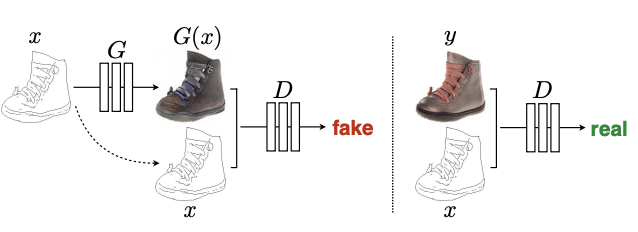
**Lê Văn Quang**

**Đặng Phan Hoàng Phúc**

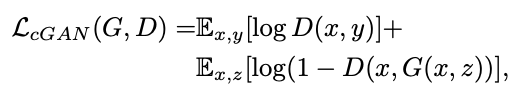
**TP. Hồ Chí Minh, tháng 06 năm 2019**

1. Cơ sở lý thuyết
   1. Phương pháp

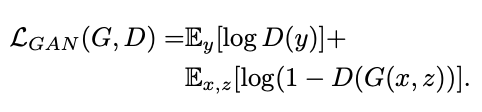
GANs là một mô hình sinh mẫu có thể học cách ánh xạ từ random noise vector z thành một hình ảnh đầu ra y, G: z -> y. conditional GANs học cách ánh xạ từ một hình ảnh x và random noise vector z thành vector y. G: { x, z } -> y. Hàm khởi tạo G được train để có thể tạo ra hình ảnh mà hàm phân biệt D không thể phân biệt được với những hình ảnh thật. Hàm phân biệt D cũng được train để nhận diện những dữ liệu giả.



* 1. Mục tiêu
* Hàm mục tiêu của condition GAN có thể được giải thích như sau:



* Trong khi G cố gắng giảm thiểu hàm này thì D lại cố gắng tối đa nó.
* Để kiểm tra tính quan trọng của việc sử dụng điều kiện trong discriminator, chúng tôi so sánh một biến thể không có điều kiện trong đó discriminator không quan tâm tới x.



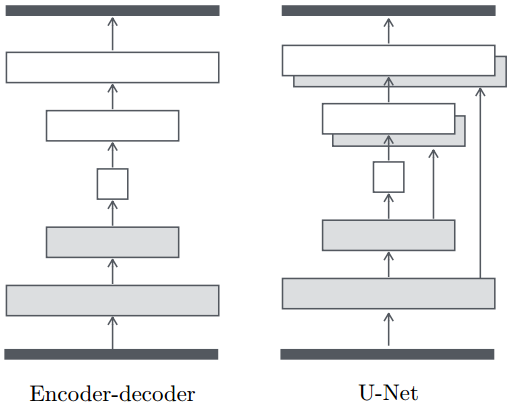
* Ưu điểm của cách trên là có thể kết hợp mục tiêu của GAN với thêm một hàm mất mát như hàm khoảng cách Euclid. Công việc của discriminator vẫn không thay đổi còn generator không chỉ phải đánh lừa discriminator mà còn phải tạo ra các output trong khoảng cách Euclid. Trong trường hợp này, chúng tôi nhận thấy sử dụng khoảng cách Manhattan cho kết quả chính xác hơn:



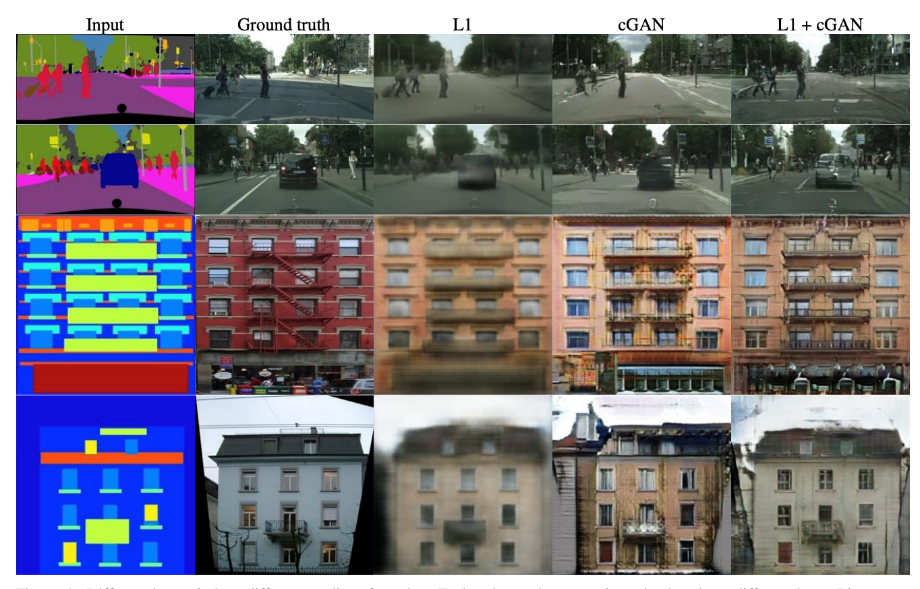
* Hàm mục tiêu cuối cùng:



* 1. Kiến trúc mạng
     1. Generator with skips
* Bài toán chuyển đổi hình ảnh sang hình ảnh sẽ ánh xạ một mạng lưới hình ảnh chất lượng cao sang một mạng lưới hình ảnh chất lượng cao khác. Ngoài ra, để giải quyết bài toán trên, chúng tôi còn cân nhắc về việc input và output sẽ khác nhau về bề mặt nhưng vẫn cùng một kết cấu. Vì thế, cấu trúc của input được sắp xếp giống với output.
* Trong một mạng lưới. Input được đưa qua các layer nhằm xử lý và đưa ra output và điều này có thể gây ra nghẽn cổ chai. Tuy nhiên trong một số trường hợp. Một số thông tin của input có thể trực tiếp đưa qua output mà không cần phải thông qua các layer (ví dụ về bài toán chỉnh sửa màu hình ảnh, thông tin về các nét của hình ảnh vẫn giữ nguyên).
* Encoder-decoder: Input được đưa qua một chuỗi các layer, cho tới khi bị nghẽn cổ chai, là điểm quá trình này bị đảo ngược.
* Để generator có khả năng tránh được việc nghẽn cổ chai. Chúng tôi sử dụng U-Net: Mạng kết hợp cho việc phân đoạn hình ảnh. Chúng tôi bỏ qua các kết nối giữa layer i và layer n-i với n là tổng số layer



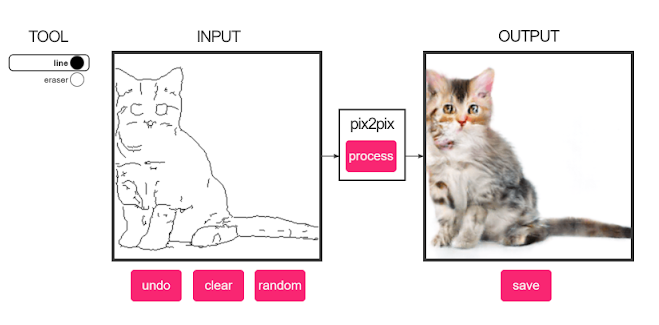
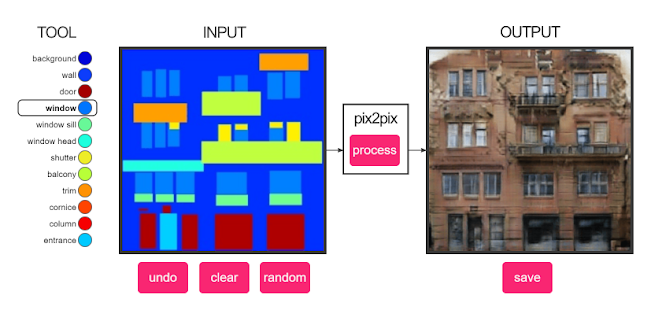
* + 1. Markovian discriminator (PatchGAN)
* Chúng ta đều biết là nếu sử dụng L2 thì sẽ có tình trạng mất hình ảnh, còn L1 thì ảnh sẽ bị mờ



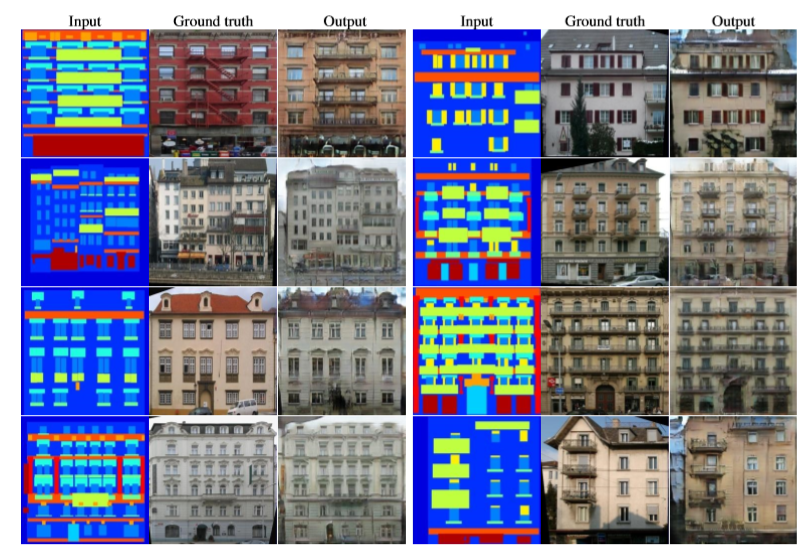
* Việc này xảy ra khi hình ảnh có độ chi tiết cao nên chúng tôi cần một framework khác để xử lí hình ảnh ở tần số thấp. L1 sẽ giúp chúng tôi làm điều này.
* Chúng tôi sử dụng PatchGAN để xử lý vấn đề này. Trong PatchGAN, discriminator sẽ cố gắng phân loại từng patch nhỏ có kích thước NxN trong một tấm ảnh. Sau đó kết hợp kết quả của từng patch để cho ra kết quả cuối cùng bằng cách tính Trung bình.
* Đối với PatchGAN, số N (kích thước nhỏ của patch) phải nhỏ hơn kích thước của hình ảnh gốc nhưng vẫn cho ra kết quả chính xác. Đây là lợi thế của PatchGAN khi nó chỉ phải chạy trên hình ảnh nhỏ hơn, và hoàn toàn có thể áp dụng cho những hình ảnh lớn.

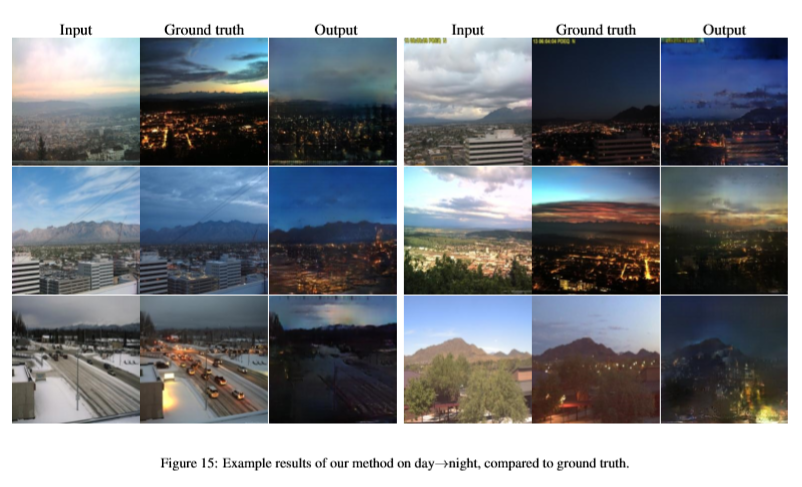
1. Thực nghiệm

* Gans có rất nhiều lợi ích trong cuộc sống:
  + Làm tranh giả, tiền giả, …
  + Dịch văn bản thành hình ảnh
  + Tăng độ phân giải của ảnh
  + Sinh ra ảnh từ một ảnh khác. Ví dụ như dưới đấy: từ hình ảnh khung của toà nhà, sinh ra hỉnh ảnh giống thực tế của ngôi nhà với cấu trúc tương ứng. Sử dụng Conditional GANs.

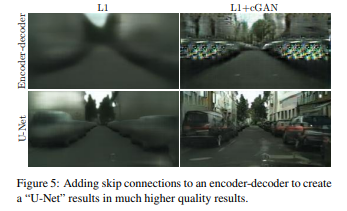


* Yêu cầu dữ liệu và tốc độ: kết quả tốt thường có thể thu được ngay cả trên các bộ dữ liệu nhỏ. Bộ huấn luyện chỉ bao gồm 400 hình ảnh (xem kết quả trong Hình 14) và bộ huấn luyện tối nay chỉ bao gồm 91 webcam duy nhất (xem kết quả trong Hình 15). Trên các tập dữ liệu có kích thước này, việc đào tạo có thể rất nhanh: ví dụ, kết quả trong Hình 14 chỉ mất chưa đầy hai giờ đào tạo trên một Pascal Titan XGPU. Thời gian thử nghiệm, tất cả các mô hình đều chạy tốt trong một giây trên GPU này.



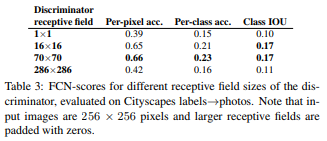


* 1. Đánh giá số liệu
* Để nâng cao chất lượng hình ảnh bốn lần, chúng tôi sử dụng hai chiến thuật. Đầu tiên, chúng tôi thực hiện các nghiên cứu về nhận thức của Real và so với giả mạo trên Amazon Mechanical Turk (AMT). Đối với các vấn đề đồ họa như màu sắc và tạo ảnh, tính hợp lý đối với người quan sát con người thường là mục tiêu cuối cùng. Do đó, chúng tôi kiểm tra việc tạo bản đồ, tạo ảnh trên không và tô màu hình ảnh bằng phương pháp này.
* Thứ hai, chúng tôi đo xem liệu cảnh quan thành phố tổng hợp của chúng tôi có đủ thực tế để hệ thống nhận dạng ngoài luồng có thể nhận ra các đối tượng trong đó hay không
  1. Analysis of the generator architecture
* Kiến trúc A U-Net cho phép thông tin cấp thấp cắt ngắn đi qua network. Điều này có dẫn đến kết quả tốt hơn? Figure 5 and Table 2 so sánh U-Net với bộ mã hóa trên thế hệ cảnh quan thành phố.
* Bộ giải mã-mã hóa được tạo đơn giản bằng cách cắt đứt các kết nối bỏ qua trong UNet
* Bộ giải mã mã hóa không thể học để tạo hình ảnh thực tế trong các thí nghiệm. Những lợi thế của U-Net dường như không dành riêng cho GAN có điều kiện: khi cả U-Net và bộ giải mã-giải mã đều được đào tạo với tổn thất L1, U-Net một lần nữa đạt được kết quả vượt trội.



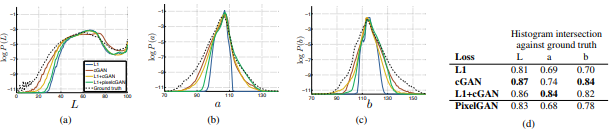
* 1. FromPixelGANstoPatchGANstoImageGANs





Hình 6: Các biến thể kích thước bản vá. Sự không chắc chắn trong đầu ra biểu hiện khác nhau đối với các hàm mất khác nhau.

Các vùng không chắc chắn trở nên mờ và không bão hòa theo L1. PixelGAN 1x1 khuyến khích sự đa dạng màu sắc lớn hơn nhưng không ảnh hưởng đến thống kê không gian. PatchGAN 16x16 tạo ra kết quả sắc nét cục bộ, nhưng cũng dẫn đến các hiện vật ốp lát vượt quá quy mô mà nó có thể quan sát được. Các PatchGAN 70 × 70 buộc các đầu ra sắc nét, ngay cả khi không chính xác, ở cả kích thước không gian và quang phổ (màu sắc). ImageGAN đầy đủ 286 × 286 tạo ra kết quả tương tự trực quan với PatchGAN 70 × 70, nhưng chất lượng thấp hơn một chút theo số liệu điểm FCN của (Bảng 3).



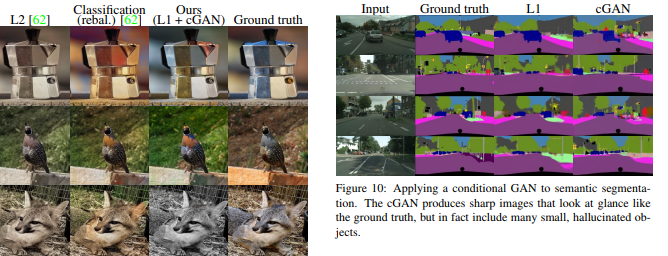
Hình 6 cho thấy định tính kết quả phân tích này và Bảng 3 định lượng các hiệu ứng bằng cách sử dụng điểm số FCN. Lưu ý rằng ở những nơi khác trong bài viết này, trừ khi được chỉ định, tất cả các thử nghiệm đều sử dụng PatchGAN 70 × 70 và cho phần này tất cả các thí nghiệm sử dụng hàm loss L1 + cGAN.

* PixelGAN không có ảnh hưởng đến độ sắc nét không gian nhưng không làm tăng màu sắc của kết quả (định lượng trong Hình 7). Ví dụ, xe buýt trong Hình 6 được sơn màu xám khi mạng được huấn luyện với tổn thất L1, nhưng chuyển sang màu đỏ với sự mất mát của PixelGAN. Kết hợp biểu đồ màu là một vấn đề phổ biến trong xử lý hình ảnh và PixelGAN có thể là một giải pháp nhẹ đầy hứa hẹn. Sử dụng PatchGAN 16 × 16 là đủ để quảng bá sắc nét đầu ra và đạt được điểm FCN tốt, nhưng cũng dẫn đến ốp lát hiện vật. PatchGAN 70 × 70 làm giảm bớt những tạo tác và đạt được điểm số tốt hơn một chút. Mở rộng ra ngoài điều này, đến ImageGAN đầy đủ 286 × 286, dường như không cải thiện chất lượng hình ảnh của kết quả và trong thực tế nhận được điểm FCN thấp hơn đáng kể (Bảng 3). Điều này có thể là do ImageGAN có nhiều tham số hơn và độ sâu lớn hơn PatchGAN 70 × 70, và có thể khó đào tạo hơn. Một lợi thế của PatchGAN là một phân biệt bản vá kích thước cố định có thể được
* áp dụng cho hình ảnh lớn tùy ý. Chúng tôi cũng có thể áp dụng trình tạo tích chập, trên các hình ảnh lớn hơn so với các hình ảnh trên mà nó đã được đào tạo . Sau khi đào tạo trình tạo trên hình ảnh 256 × 256, kiểm tra nó trên hình ảnh 512 × 512. Các kết quả trong Hình 8 chứng minh

hiệu quả của phương pháp này



* trong các bản đồ, có hình học cứng nhắc, hơn trong các bức ảnh chụp từ trên không, hỗn loạn hơn. đào tạo màu sắc trên ImageNet và đã thử nghiệm trên phần tách thử nghiệm với loss L1 + cGAN, đánh lừa người tham gia trên 22,5% thử nghiệm (thử nghiệm kết quả của [62] và một biến thể của phương pháp của họ đã sử dụng mất L2 (xem [62] để biết chi tiết). Các GAN có điều kiện ghi điểm tương tự như biến thể L2 của [62] sự khác biệt không đáng kể bằng thử nghiệm bootstrap, nhưng không đạt [62] Phương pháp đầy đủ của Google, đánh lừa người tham gia trên 27,8% thử nghiệm trong thí nghiệm của chúng tôi. phương pháp của họ là đặc biệt thiết kế để làm tốt về màu sắc.

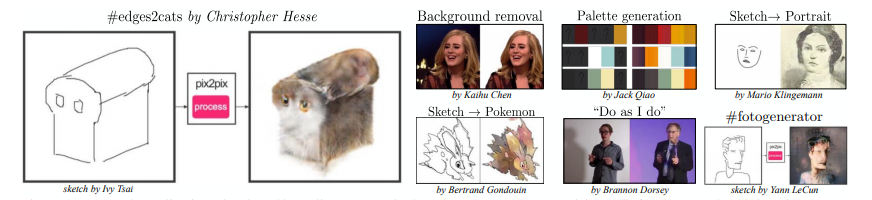


* 1. Perceptual validation

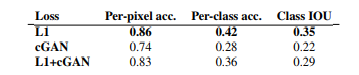
Chúng tôi xác nhận tính thực tế cảm nhận về kết quả của chúng tôi trên nhiệm vụ chụp ảnh bản đồ và thang độ xám → màu sắc. Kết quả thử nghiệm AMT của chúng tôi cho map↔photo được đưa ra trong

Bảng 4. Các ảnh chụp từ trên không được tạo bởi phương pháp của chúng tôi bị đánh lừa những người tham gia trên 18,9% thử nghiệm, cao hơn đáng kể so với L1 đường cơ sở, tạo ra kết quả mờ và gần như không bao giờ lừa người tham gia. Ngược lại, trong ảnh → hướng bản đồ, phương pháp của chúng tôi chỉ đánh lừa người tham gia trên 6,1% thử nghiệm và điều này không khác biệt đáng kể so với hiệu suất của đường cơ sở L1 (dựa trên thử nghiệm bootstrap).

* 1. Phân khúc ngữ nghĩa
* GAN có điều kiện dường như có hiệu quả đối với các vấn đề khi kết quả là phần lớn ảnh chụp ảnh, như là xử lý hình ảnh in phổ biến và các tác vụ đồ họa.



Hình 11: Các ứng dụng ví dụ được phát triển bởi cộng đồng trực tuyến dựa trên cơ sở mã pix2pix của chúng tôi: # edge2cats [3] của Christopher Hesse, Xóa nền [6] bởi Kaihu Chen, Tạo bảng [5] bởi Jack Qiao, Phác thảo → Chân dung [7] của Mario Klingemann, Phác thảo → Pokemon [1] của Bertrand Gondouin, hoay Do As Do Do Do đặt ra chuyển nhượng [2] bởi Brannon Dorsey và #f photoenerator của Bosman et al. [4].



Bảng 6: Hiệu suất của ảnh → nhãn trên cảnh quan thành phố.

* Về các vấn đề tầm nhìn, như phân khúc ngữ nghĩa, trong đó đầu ra thay vì ít phức tạp hơn đầu vào?
* Để bắt đầu thử nghiệm điều này, chúng tôi đào tạo một cGAN (có / không có L1 mất) trên ảnh cảnh quan thành phố → nhãn. Hình 10 cho thấy kết quả định tính và độ chính xác phân loại định lượng được báo cáo trong Bảng 6. Điều thú vị là các cGAN, được đào tạo mà không có Mất L1, có thể giải quyết vấn đề này ở mức độ hợp lý độ chính xác. Theo hiểu biết của chúng tôi, đây là minh chứng đầu tiên về GAN tạo thành công các nhãn mác, đó là gần như rời rạc, thay vì hình ảnh trên mạng, với biến thể được định giá liên tục. Mặc dù cGAN đạt được một số thành công, họ là xa phương pháp tốt nhất có sẵn để giải quyết điều này vấn đề: chỉ cần sử dụng hồi quy L1 sẽ đạt điểm cao hơn sử dụng một cGAN, như trong Bảng 6. Chúng tôi lập luận rằng đối với các vấn đề về tầm nhìn, mục tiêu (nghĩa là dự đoán đầu ra gần với sự thật mặt đất) có thể ít mơ hồ hơn các tác vụ đồ họa, và tổn thất tái thiết như L1 là đủ.
  1. Nghiên cứu dựa vào cộng đồng

Kể từ khi phát hành ban đầu của bài báo và pix2pix của chúng tôi codebase, cộng đồng Twitter, bao gồm các chuyên gia về thị giác và đồ họa máy tính cũng như các nghệ sĩ thị giác, đã áp dụng thành công khuôn khổ của chúng tôi cho một loạt các tiểu thuyết nhiệm vụ dịch thuật từ hình ảnh sang hình ảnh, vượt xa phạm vi giấy gốc. Hình 11 và Hình 12 chỉ ra một một vài ví dụ từ hashtag # pix2pix, bao gồm Xóa nền, tạo bảng màu, Phác thảo → Chân dung, Phác thảo → Pokemon, Lôi làm khi tôi thực hiện chuyển giao, Học để xem: Chủ nhật ảm đạm, cũng như phổ biến kỳ lạ # edge2cats và #f photoenerator. Lưu ý rằng các ứng dụng này là các dự án sáng tạo, không được kiểm soát, điều kiện khoa học, và có thể dựa vào một số sửa đổicơ sở mã hóa. Vui lòng nhấp vào hình ảnh để phát video trong trình duyệt. mã pix2pix chúng tôi phát hành. Tuy nhiên, họ thể hiện lời hứa về cách tiếp cận của chúng tôi như là một hàng hóa chung công cụ cho các vấn đề dịch ảnh sang hình ảnh.



Hình 12: Học để xem: Gloomy Chủ nhật: Một bản demo nghệ thuật tương tác được phát triển bởi Mem Akten [8] dựa trên pix2pix của chúng tôi

* 1. Kết luận

Các kết quả trong bài viết này cho thấy các mạng đối nghịch có điều kiện là một cách tiếp cận đầy hứa hẹn cho nhiều nhiệm vụ dịch thuật hình ảnh, đặc biệt là các nhiệm vụ liên quan cao cấu trúc đầu ra đồ họa. Các mạng này học một sự mất mát thích nghi với nhiệm vụ và dữ liệu trong tay, điều này giúp chúng có thể áp dụng trong nhiều cài đặt khác nhau.