ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC MÁY TÍNH

KHỬ MỜ ẢNH BẰNG MÔ HÌNH WAVELET GAN

ĐỒ ÁN MÔN: MÁY HỌC

Sinh viên: Nguyễn Đức Toàn 18521506

Trương Văn Tuấn 19522486

Trương Minh Sơn 19522143

GIÁO VIÊN HƯỚNG DẪN:

PGS.TS Lê Đình Duy

Th.S Nguyễn Phạm Trường An

Thành phố Hồ Chí Minh, Tháng 12 năm 2021

I. TỔNG QUAN

1. Mô tả bài toán

Trong thực tế, xung quanh ta luôn tồn tại các hạt vật chất nhỏ li ti trong không khí mà mắt thường không thể nhìn thấy. Nhưng khi số lượng hạt này đủ lớn thì nó sẽ ngăn cản một lượng ánh sáng từ vật thể, cảnh đến mắt ta hay đến các thiết bị camera, điều này gây nên hình ảnh mà chúng ta nhìn thấy bằng mắt, chụp, hay quay trên các thiết bị có sử dụng carmera bị mờ hay không thể thấy được. Lượng ánh sáng bị ngăn cản này nhiều hay ít còn phụ thuộc vào số lượng hạt và tính chất vật lý của hạt đó. Khử mờ hình ảnh là một vấn đề khó trong nhiều ứng dụng, các phương tiện thông minh, hệ thống vệ tinh giám sát và điều này cũng nên gây khó khăn trong các tác vụ cấp cao trong thị giác máy tính như nhận diện đối tượng, phân đoạn đối tượng, truy xuất và phân lớp.

Để giải quyết vấn đề này đã có rất nhiều phương pháp ra đời dựa trên các tính chất vật lý, tính chất quang học như là tăng cường độ tương phản hình ảnh, tinh chỉnh histogram, ánh xạ tuyến tính, điều chỉnh Gamma, phương trình tán xạ khí quyển. Tuy nhiên các phương pháp dựa trên vật lý và quang học thì chất lượng của ảnh được khử mờ còn kém hiệu quả, chất lượng cải thiện không cao hoặc với những dữ liệu thách thức chúng không thể cải thiện được. Hiện nay, với sự phát triển của học sâu các mô hình khử mờ được thiết kế dựa trên kiến trúc CNN đã có sự cải thiện đáng kể về mặt hiệu suất tái tạo ảnh mờ trên những bộ dữ liệu khó và thách thức.

Trong bài báo cáo lần này nhóm tụi em đã chọn phương pháp WAVELET GAN và tự cài đặt lại source train, và có thay đổi trọng số của loss để tăng hiệu suất của độ đo PSNR.

Input của bài toán là một ảnh mờ.

Output là ảnh đã được khử mờ.

Haze image

Haze free image

2. Mô tả dữ liệu

Trong bài báo cáo lần này nhóm em sử dụng bộ dữ liệu NTIRE 2021 do NTIRE 2021 NonHomogeneous Dehazing Challenge là một workshop của CVPR 2021.

Bộ dữ liệu có được xây dựng trên phiên bản mở rộng của tập NH-TIRE. Bộ dữ liệu NTIRE 2021 gồm 35 cặp hình ảnh mờ và hình ảnh không mờ và chúng cùng một cảnh. NTIRE 2021 chứa các cảnh ngoài trời với sương mù thật và mật độ sương mù không đồng nhất được tạo ra kỹ thuật tạo sương mù chuyên nghiệp. Để tạo ra những ảnh này ban tổ chức đã sử dụng hai máy tạo khói mù chuyên nghiệp tạo ra các hạt hơi có kích thước đường kính (1-10 microns) tương tự các hạt sương mù trong khí quyển.

Để ghi lại cảnh ban tổ chức sử dụng máy ảnh Sony A7 được điều khiển từ xa các thông số của mỗi lần lấy ảnh đều được chỉnh thủ công, và được giữ nguyên giữa hai phiên quay liên tiếp.

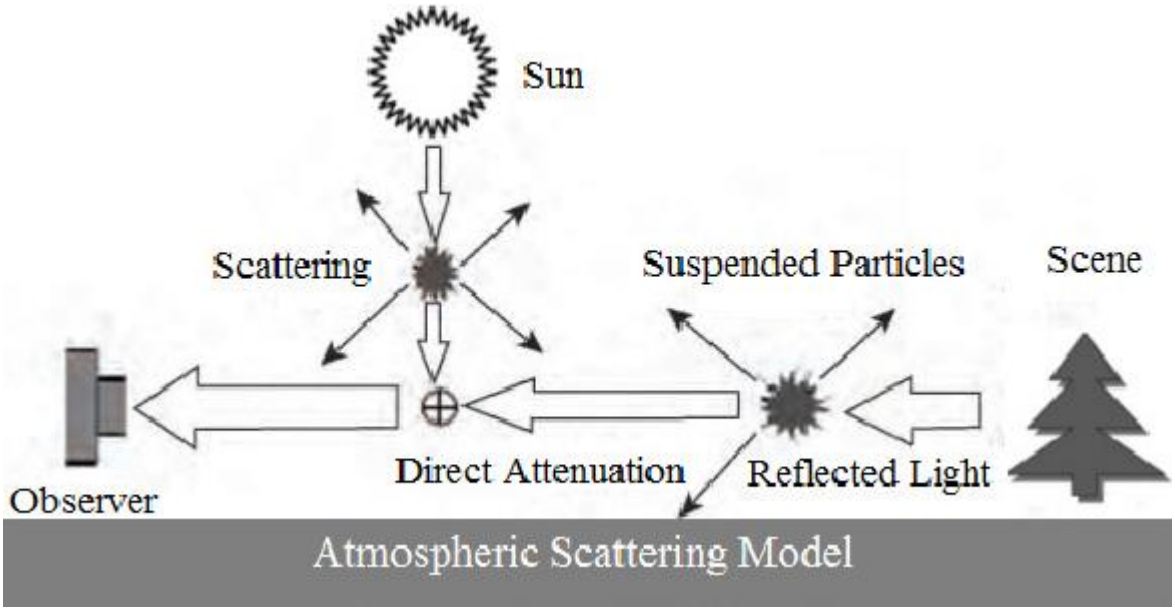
Quá trình lấy một cặp ảnh mất tầm 20-30 phút. Các tham số camera đươc set là (aperture-exposure-ISO), để cân bằng trắng (18percent gray) của color checker.

II. CÁC NGHIÊN CỨU CÓ TRƯỚC

1. DCP – Dark prior channel

Atmosphere Scattering Model (Mô hình tán xạ khí quyển)

Mô hình tán xạ khí quyển mô tả sự hình thành của ảnh mờ.



Hình 9: Mô hình tán xạ khí quyển.

$$I(x) = J(x)e^{-\beta d(x)} + A\left(1 - e^{-\beta d(x)}\right)$$

Phương trình này dựa trên kênh màu RGB

x : Tọa độ điểm ảnh(pixel).

$I(x)$: biểu diễn 1 pixel trên image.

$J(x)$: vector bức xạ tại điểm tương tác của cảnh và tia chiếu trong thế giới thực không bị tác động bởi cách tác nhân bên ngoài.

A là biểu diễn cho ánh sáng khí quyển toàn cục của bức ảnh.

$e^{-\beta d(x)}$ transmission: thể hiện phần ánh sáng tương đối có thể tồn tại trên toàn bộ đường đi từ 1 điểm của scene đến mắt hoặc camera mà không bị phân tán.

d khoảng cách từ điểm ảnh của scene đến người quan sát hoặc camera.

β: hệ số tán xạ khí quyển

Năm 2009 Kaiming He và các đồng nghiệp của ông đã thực hiện một cuộc khảo sát và thực nghiệm về đặc tính của ảnh ngoài trời không có sương mù.

Họ nhận ra rằng có ít nhất một kênh màu trong một vùng ảnh có những pixel tối có giá trị cường độ tối gần bằng 0.

Dựa trên quan sát này, họ định nghĩa một kênh tối nhưng sau:

$$J^{dark}(x) = \min_{c\in(r,g,b)} \left(\min_{y\in\Omega(x)} \left(J^c(y) \right) \right).$$

J^c kênh màu của J và $\Omega(x)$ và vùng ảnh cục bộ có tâm tại x.

Quan sát thấy ngoại trừ vùng trời, thì cường độ J^{dark} thấp và xấp sỉ bằng 0, nếu J là ảnh ngoài trời không có sương mù J^{dark} được gọi là kênh tối J. Quan sát thông kê trên gọi là DCP.

Các cường độ thấp trong kênh tối do 3 yếu tố:

+ Bóng tối
+ Đối tượng nhiều màu sắc
+ Các vật bề mặt tối

Ước lượng Transmission

Ta có phương trình của mô hình tán xạ ánh sáng:

+ $I(x) = J(x)e^{-\beta d(x)} + A\left(1 - e^{-\beta d(x)}\right)$
+ Đầu tiên, giả định rằng A chúng ta đã được ước lượng, $e^{-\beta d(x)}$ là không đổi trong vùng ảnh y ∈ $\Omega(x)$ và đặt là $\tilde{t}(x)$
+ $I^c(x) = \tilde{t}(x)J^c(x) + A^c\left(1 - \tilde{t}(x)\right)$ (*)
+ $J^{dark}(x) = \min_{c\in(r,g,b)} \left(\min_{y\in\Omega(x)} \left(J^c(y) \right) \right)$ (**)
+ Dựa vào (*)(**) ta có:
+ $\tilde{t}(x) = 1 - \min_c \left(\min_{y\in\Omega(x)} \frac{I^c(y)}{A^c} \right).$

Màu sắc của bầu trời thường rất giống ánh sáng khí quyển toàn cục của bức ảnh. $A^c \approx I^c(y)$

+ $\min_c \left(\min_{y \in \Omega(x)} \frac{I^c(y)}{A^c} \right) \to 1 \Rightarrow \tilde{t}(x) \to 0 \leftrightarrow d = e^{-\beta d(x)}$ tiến đến vô cực điều này là chính xác khi chúng ta không thể đo được khoảng cách không gian giữa bầu trời và người quan sát hay camera.
+ Nên ta không cần tách vùng trời trước khi dùng DCP.

Trong thực tế luôn luôn tồn tại các hạt vật chất trong không khí, vì vậy ánh sáng từ cảnh chuyển đến mắt ta không bao giờ là 100% lượng ánh sáng từ cảnh truyền ra ngoài, nhất là cảnh nhìn từ xa, aerial perspective.

Khói mù là một dấu hiệu để nhận thức được chiều sâu của cảnh.

Dẫn đến nếu chúng ta làm xóa mù kỹ lưỡng dẫn đến mất đi nhận thức về chiều sâu ảnh sẽ không tự nhiên.

Để giải quyết vấn đề này người ta sẽ giữ lại một lượng 'khói mù' bằng cách thêm một lượng omega $\omega (0 < \omega \leq 1)$ vào phương trình ước lượng transmission.

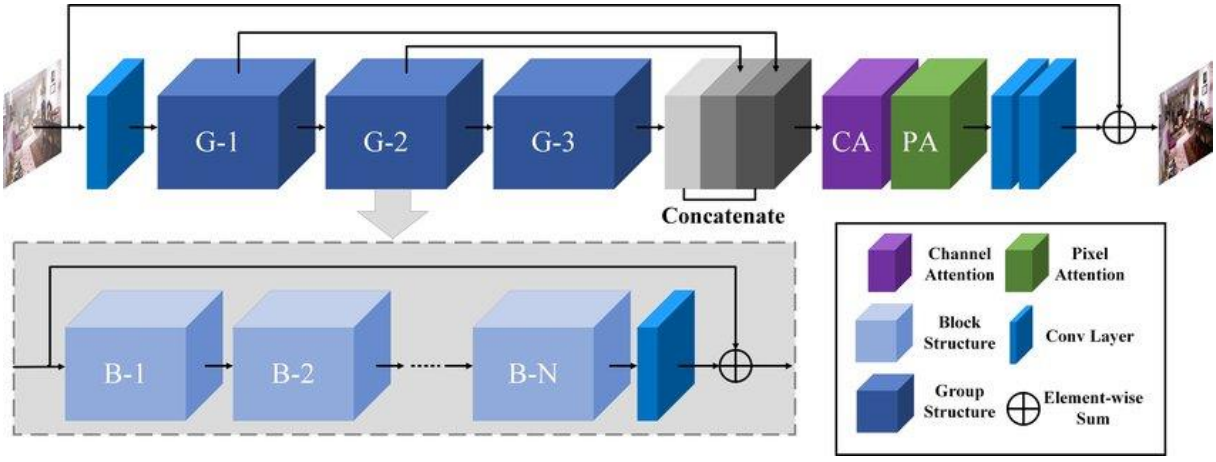
+ $\tilde{t}(x) = 1 - \omega \min_c \left(\min_{y \in \Omega(x)} \frac{I^c(y)}{A^c} \right)$

Khi $\tilde{t}(x) \to 0$ dẫn đến $J(x) = \frac{I(x)-A}{\tilde{t}(x)} + A$ không thể khôi phục được, làm cho vùng ảnh tại vị trí pixel x nhiễu.

Tinh chỉnh hàm khôi phục $J(x)$ bằng cách thêm một ngưỡng t_0 giúp ta tránh trường hợp ở trên đã nêu.

+ $J(x) = \frac{I(x)-A}{\max(t(x), t_0)} + A \; (defaut \; t_0 = 0.1)$

2. FFA Net

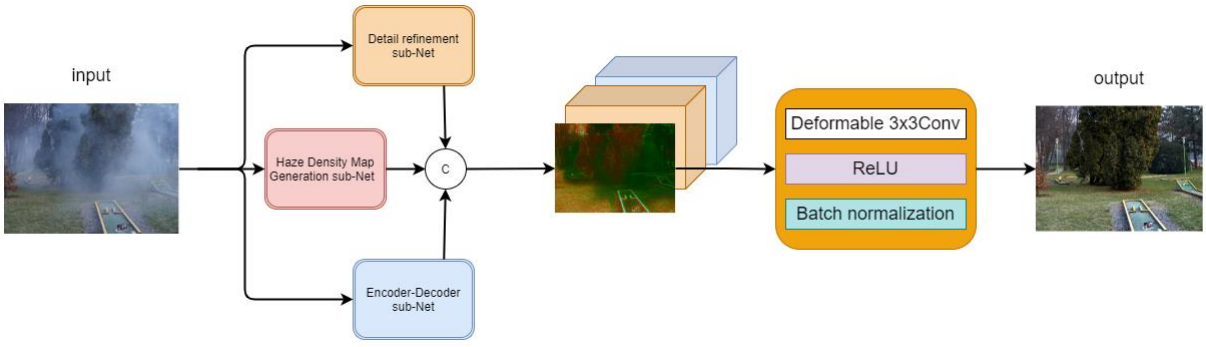


Mô-đun Feature Attention (FA) kết hợp cơ chế Channel Attention và Pixel Attention. FA xử lý các đặc trưng và điểm ảnh không đồng đều vì tác giả cho rằng sự phân bố sương mờ trên các vùng điểm ảnh khác nhau là khác nhau. Điều này tạo ra tính linh hoạt trong việc xử lý các vùng ảnh có mật độ sương dày mỏng khác nhau.

Khối kiến trúc cơ bản bao gồm Local Residual Learning (LRL) và Feature Attention làm cho quá trình huấn luyện trở nên ổn định hơn đồng thời cũng tăng hiệu quả khử sương. Điều này có được bởi vì LRL làm cho cấu trúc mạng chú ý đến các thông tin quan trọng và bỏ qua các vùng ít thông tin như vùng sương mỏng.

Kiến trúc Attention-based different levels Feature Fusion (FAA) cho phép trọng số được học thích ứng từ mô-đun FA, mang lại trọng số có giá trị cao hơn cho các thông tin quan trọng. Kiến trúc này cũng giữ lại được thông tin của các lớp ban đầu và truyền nó vào các lớp sâu hơn nhớ áp dụng Global Residual Learning

3. Trident Dehazing Network



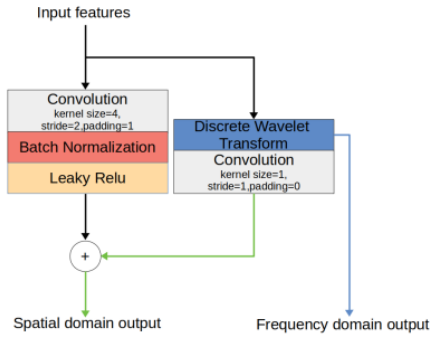
TDN được đề xuất bởi Jing Liu và các cộng sự của ông năm 2020, phương pháp này đã đứng nhất trong NTIRE2020 Challenge on NonHomogeneous Dehazing do CVPR2020 tổ chức. TDN được tạo thành bởi sự kết hợp của ba mạng con gồm có Haze Density Map Generation sub-Net, Encoder Decoder sub-Net, Detail Refinement sub-Net, mỗi mạng đều có một chức năng riêng giúp cho việc khôi phục ảnh mờ do sương tốt lên.

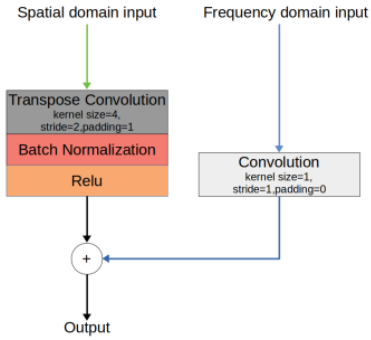
Haze Density Map Generation sub-Net lấy ý tưởng từ Unet có chức năng phân vùng mật độ sương mù. Do việc huấn luyện có số lượng dữ liệu khá ít, nên việc dựa trên những "kiến thức" trước đó đã có sẵn để tăng hiệu quả rút trích đặc trưng thì Encoder Decoder đã Pretrained IamgeNet1k trên DPN92 để học chuyển tiếp. Cuối cùng là Detail Refinement sub-Net giúp cho ảnh cải thiện từ ảnh có độ phân giải thấp

thành ảnh có độ phân giải cao. Cụ thể trong Trident Dehazing Network, tác giả đề xuất Detail Refinement sub-Net để cải thiện kết quả khử mờ giúp ảnh sắc nét và tăng tính chân thực.

III. A DISCRETE WAVELET TRANSFORM GAN – DW GAN

1. Wavelet convolution module





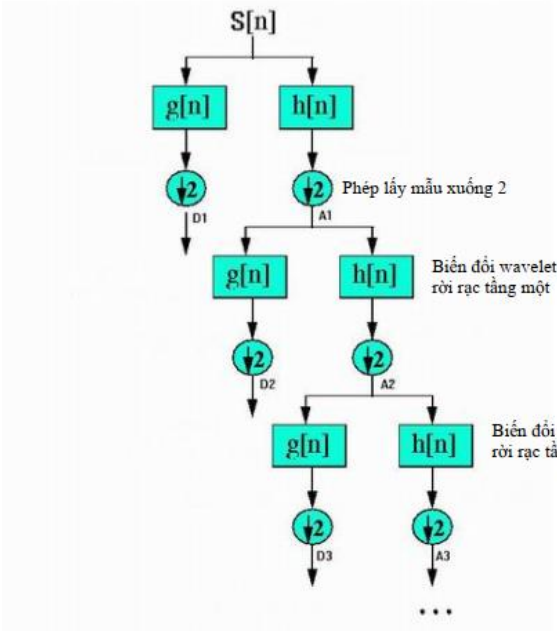
Continuous Wavelet Transform (CWT)

$$T(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\, \psi^* \frac{(t-b)}{a} dt$$

Discrete Wavelet Transform (DWT)

$$T_{m,n} = \int_{-\infty}^{\infty} x(t)\, \psi_{m,n}(t)\, dt$$

Wavelet là một dao động giống như sóng với biên độ bắt đầu bằng 0, tăng hoặc giảm, sau đó trở về 0 một hoặc nhiều lần. Wavelet được gọi là "dao động ngắn". Một phân loại các wavelet đã được thiết lập, dựa trên số lượng và hướng của các xung của nó. Wavelet có thuộc tính cụ thể làm cho chúng hữu ích cho việc xử lý tín hiệu.



2. Attention

Attention là một kỹ thuật hiện đại trong các mạng nơ ron nhân tạo. Kỹ thuật này đã chứng minh được tính hiệu quả trong các nhiệm vụ dịch máy hay xử lý ngôn ngữ tự nhiên. Nó cũng là một trong số những thành phần tạo nên đột phá trong các mô hình như BERT hay GPT-2.

Theo thực tế, mình nghĩ bộ não của chúng ta cũng có cơ chế attention của riêng nó. Ví dụ như, mắt của chúng ta có tầm nhìn 120 độ theo cả chiều thẳng đứng và ngang.

Tuy nhiên, tại một thời điểm, hầu như chúng ta chỉ xử lý một phần nhỏ thông tin của bức ảnh. Bạn có để ý khi chúng ta lái xe, lúc rẽ trái, hay phải, chúng ta chỉ chú ý vào một phần không gian trên kính chiếu hậu mà thôi, rồi từ đó mới xử lý để đưa ra quyết định di chuyển tiếp theo. Cơ chế này của bộ não giúp chúng ta không cần nhiều năng lượng để đưa ra quyết định tuy nhiên vẫn cho kết quả tin cậy.

Có khá nhiều cơ chế Attention, nhưng tổng quan có 2 loại chính:

- Hard Attention: sử dụng reinforcement learning để huấn luyện.
- Soft Attention: sử dụng backpropagation cơ bản để huấn luyện.

Với hard attention, mô hình sẽ chọn ngẫu nhiên một vùng ảnh để chú ý, do không có nhãn vùng nào cần được chú ý, nên cũng không tính được gradient lúc này. Để giải quyết vấn đề này, sẽ áp dụng reinforcement learning. do sử dụng reinforcement learning nên cũng gặp những khó khăn của phương pháp này như khó hội tụ, mà kết quả thật ra cũng không tốt hơn so với soft attention. Tuy nhiên, cũng có ưu điểm như là vì chỉ chọn một vùng ảnh để tính toán nên sẽ giảm được tài nguyên máy tính cần để xử lý.

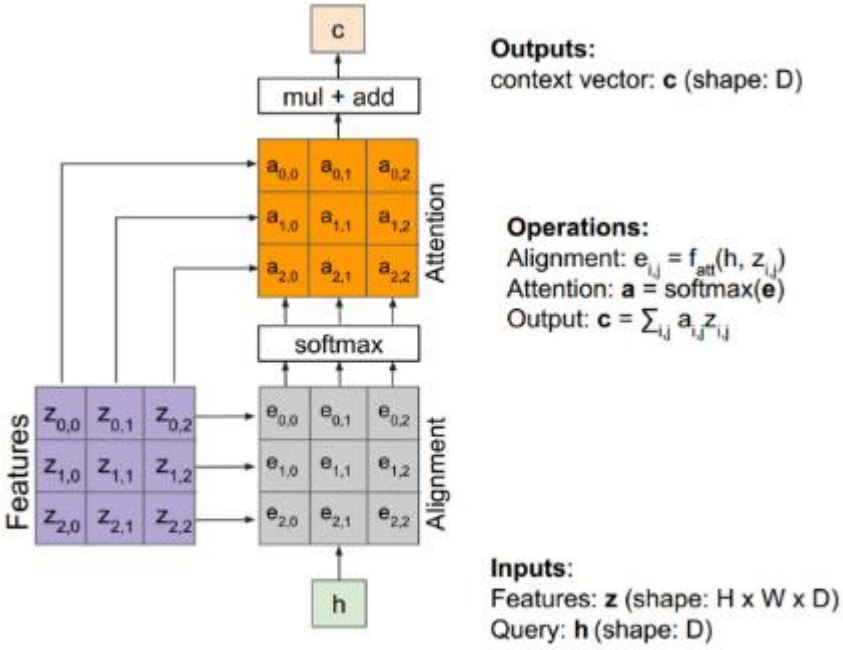
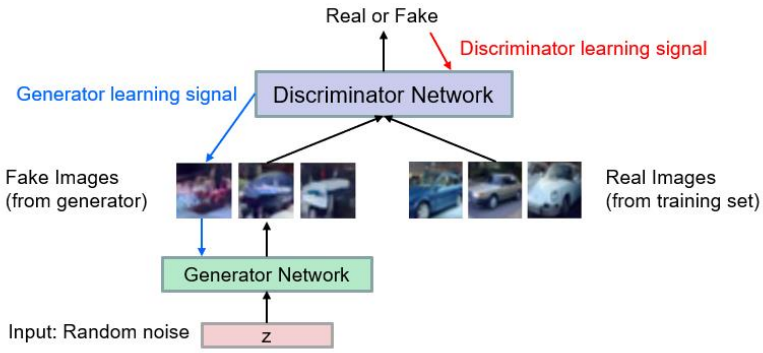


Figure 2: Attention

Với soft attention, mô hình sẽ học trọng số để chú ý trên tất cả các phần thông tin của bức ảnh, câu, hoặc bất cứ thứ gì mà nghĩ rằng việc tổng hợp thông tin của tất cả các phần là cần thiết để đưa ra dự đoán. Tổng hợp thông tin này được tính bằng trung bình cộng có trọng số của tất cả các phần thông tin. Những trọng số này được mô hình tự học dễ dàng bằng backpropagation.

Attention khá nổi tiếng trong các bài toán về NLP dịch máy, image captioning. mô tả kiến trúc attention được nếu trong 'Show, Attend and Tell: Neural Image Caption Generation with Visual Attention'. Đầu vào của attention được nhắc trong bài báo này là các feature đã được trích xuất từ các lớp convolution đầu tiên các feature này sẽ được đi qua một lớp MLP để sinh ra một vector query h có hình dạng D. Tiếp tục, vector query và feature được nêu trên sẽ đưa vô lớp MLP thứ hai và sinh ra một ma trận hoặc vector có tên là Alignment. Các phần tử Alignment này sẽ được chuẩn hóa bằng softmax layer đế lấy ra bộ trọng số attention và có tổng bằng một. Cuối cùng bộ trọng số attention này sẽ được nhân với feature đầu vào để ra được context vector c cũng là output của attention.

3. GAN

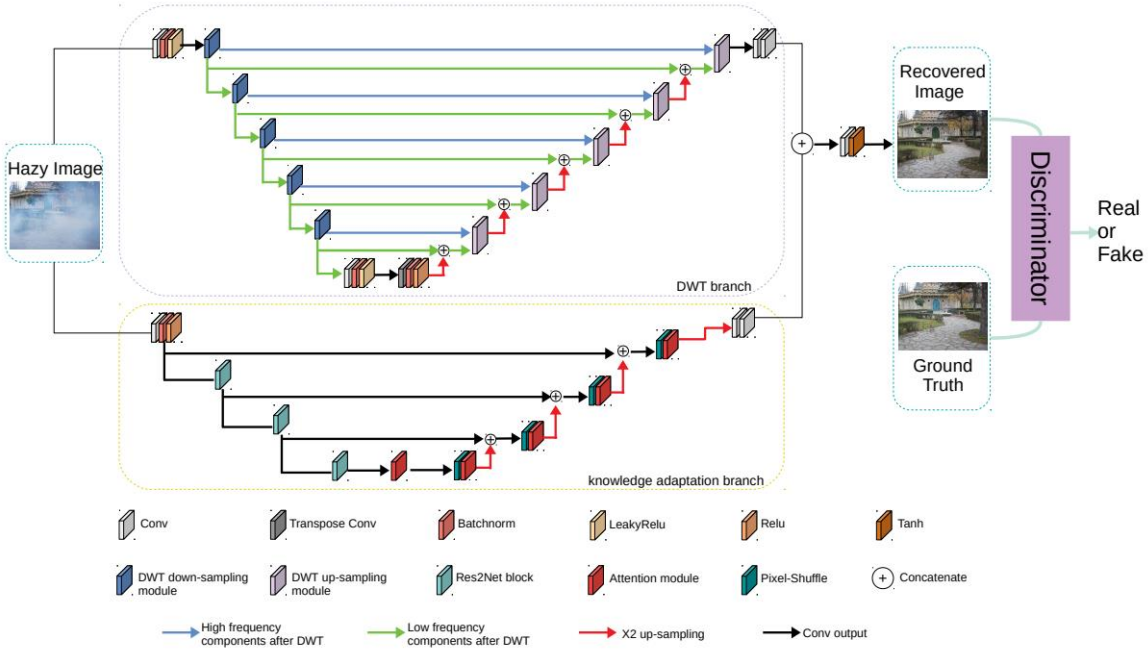


$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{\boldsymbol{x} \sim \boldsymbol{p_{data}}} \log D_{\theta_d}(x) + \mathbb{E}_{\boldsymbol{z} \sim \boldsymbol{p_{(z)}}} \log(1 - D_{\theta_d}\left(G_{\theta_g}(z)\right)) \right]$$

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.

GAN viết tắt cho Generative Adversarial Networks. Generative giống như ở trên, Network có nghĩa là mạng (mô hình), còn Adversarial là đối nghịch. Tên gọi như vậy là do GAN được cấu thành từ 2 mạng gọi là Generator và Discriminator, luôn đối nghịch đầu với nhau trong quá trình train mạng GAN.

4. DW GAN



DW GAN được tạo thành bởi các kiến trúc được nếu ở trên, có hai module chính là phần DWT branch và Knowledge adaptation branch. DWT branch sử dụng biến đổi wavelet rời rạc 2d với sóng embed haar. Việc học chuyển tiếp để tận dụng các trọng số đã được học trên các bộ dữ liệu lớn đã được huấn luyện để cải thiện việc học trên bộ dữ liệu nhỏ ở đây Knowledge adaptation branch sử dụng Res2Net được pretrained trên ImageNet để học chuyển tiếp.

IV. TRAINING VÀ ĐÁNH GIÁ MODEL

1. Training

Do ban tổ chức chỉ public 25 ảnh nên nhóm thực hiện chia ra làm 20 ảnh train và 5 ảnh validation.

Với bộ data này nhóm thực hiện random crop 256x256, xoay trái phải, xoay (90, 180, 270). Sau đó chuẩn hóa standard normal.

2. Optimazation

Trong bài toán này có 4 hàm loss: Smooth L1 Loss, Perceptual Loss, MS-SSIM Loss, Adversarial Loss được mô tả cụ thể dưới đây.

Smooth L1 loss:

$$L_1 = \frac{1}{3N}\sum_{i=1}^{N}\sum_{c=1}^{3}\alpha\left(\hat{I}_c(i) - I_c^{gt}(i)\right)$$

$$\alpha(e) = \begin{cases} 0.5e^2, if\ |e| < 1 \\ |e| - 0.5 \end{cases}$$

Perceptual loss:

$$L_2 = \sum_{j=1}^{3}\frac{1}{C_j H_j W_j}\left\|\phi_j(I^{gt}) - \phi_j(\hat{I})\right\|_2^2$$

SSIM Loss: Chi tiết về SSIM sẽ được trình bày ở phần metric

$$L_3 = |SSIM(I^{gt}) - SSIM(\hat{I})|$$

Adversarial loss:

$$L_4 = \sum_{n=1}^{N} -\log\left(D\left(G(I^{hazy})\right)\right)$$

Total loss:

$$L = \alpha L_1 + \beta L_2 + \gamma L_3 + \delta L_4$$

Các tham số $\alpha, \beta, \gamma, \delta$ dùng để hiệu chỉnh tầm quan trọng của các loss.

Để huấn luyện sử dụng ADAM optimizer ($\beta_1 = 0.9, \beta_2 = 0.999$) Learning rate được khởi tạo là 1e-04 được giảm một nữa khi qua epoch thứ 3000, 5000 và 6000 trên tổng 8000 epoch. Được huấn luyện trên 2 con GPU RTX 2080 ti.

3. Độ đo

PSNR: dùng để đo chất lượng dữ liệu khôi phục được của các thuật toán có mất mát dữ liệu. Thông thường PSNR càng cao thì chất lượng khôi phục càng tốt. Đơn vị đo (lb).

$$\text{PSNR} = 20\log_{10}\left(\frac{\text{MAX}_\text{I}}{\sqrt{\text{MSE}}}\right)$$

MAX_I là giá trị pixel lớn nhất tương ứng với loại ảnh n bit ($\mathbf{2^{n-1}}$).

Với ảnh 8 bit khoảng giá trị chấp nhận ổn là tầm 30db – 50db.

Với ảnh 16 bit khoảng giá trị chấp nhận ổn là tầm 60db – 80db

SSIM: (tructural similarity index measure): là một độ đo sự tương đồng cấu trúc giữa hình ảnh bị thay đổi cấu trúc, ánh sáng, suy giảm thông tin với hình ảnh gốc cần xét.

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

+ μ_x, μ_y là các giá trị trung bình của 2 ảnh x và y.
+ σ_{xy} là hiệp phương sai của 2 ảnh x và y.
+ $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$, chose $k_1 = 0.01$ $k_2 = 0.03$,với ảnh n bit thì $L = 2^n - 1$.
+ σ_x, σ_y là phương sai của ảnh x và y.
+ $ssim \in [0,1]$, bằng không là không có sự tương đồng giữa 2 ảnh, ngược lại thì 2 ảnh giống nhau hoàn toàn.

4. Kết quả thực nghiệm

	NTIRE21	
	PSNR	**SSIM**
DCP	**11.68**	**0.7090**
AOD-Net	**13.30**	**0.4693**
GCANet	**18.79**	**0.7729**
FFA	**20.45**	**0.8043**
TDN	**20.23**	**0.7622**
DW-GAN	**21.99**	**0.8560**
Our	**22.10**	**0.8490**