

Machine Learning Final Project

DeHazing

隊名：NTU_b04901104_中部綜量級選民

組員：B04901069 林志皓

B04901072 劉雨東

B04901104 吳添聚

Introduction & Motivation

在攝影的過程中，如果遇到有霧的場景，產生出的照片就會相當模糊不清，因此如何利用電腦視覺以及深度學習的方式，將霧氣去除，顯示出底下原本鮮艷的場景，是個非常重要的問題。去年及今年的 CVPR 皆針對這個任務舉辦競賽，徵求好的解法。競賽中的資料即為這次期末專題的資料，為許多成對的「有霧」、「沒霧」的照片組成 **training data**，然而這些有霧的照片並非實際拍攝，而是經過人工後製而成，基於以下公式：

$$I(x) = J(x)t(x) + A(x)(1 - t(x))$$

$I(x)$: 覆蓋著霧的照片

$J(x)$: 清晰沒有霧的照片（同一場景）

$A(x)$: 代表霧的顏色濃淡分佈(global atmospheric light indicating the intensity of the ambient light)

$t(x)$: 代表圖片該位置的深度（transmission map）(Transmission map is the distance-dependent factor that affects the fraction of light that reaches the camera sensor)

從上述可知，我們的目標是將得到的 $I(x)$ ，預測出相對應的 $J(x)$

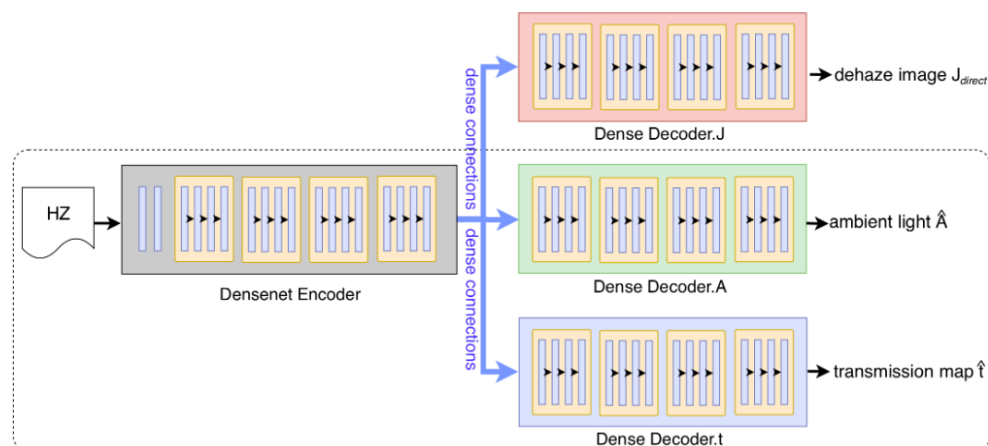
若能將這個任務做的好，可以在至少兩方面的實際應用獲得很大的方便。一是手機拍照體，能將有霧的照片經過後製得到清晰的照片；另一方面是自駕車、無人機領域，這些應用需要對於環境非常精確的偵測，如果能在充滿霧氣的環境當中，精確的描繪出環境輪廓，就能更精確的規劃出行進路線，能更加安全順暢的操作。



圖一。左為有霧的圖片，右為我們希望得到的目標

Methods

1. AtJ-DH network



這個模型在輸入 **training image** 後，分別預測出沒有霧的圖片 (**J**)、霧的分佈 (**A**)、**transmission map** (**I**)，再用這三個東西算出一張有霧的圖片並與原輸入圖片比較出 **loss**，進而訓練出模型。

$$\hat{\mathbf{J}} = \frac{\mathbf{I} - \hat{\mathbf{A}} \cdot (1 - \hat{\mathbf{t}})}{\hat{\mathbf{t}}}$$

然而效果並不理想，因此我們並無繼續優化，改由第二種方式進行嘗試。

2. Multi-scale Single Image Dehazing using Perceptual Pyramid Deep Network

我們參考 2018 這個篇論文中的方法，進行更進一步的優化，而模型的架構圖如下，在此介紹此方法流程：

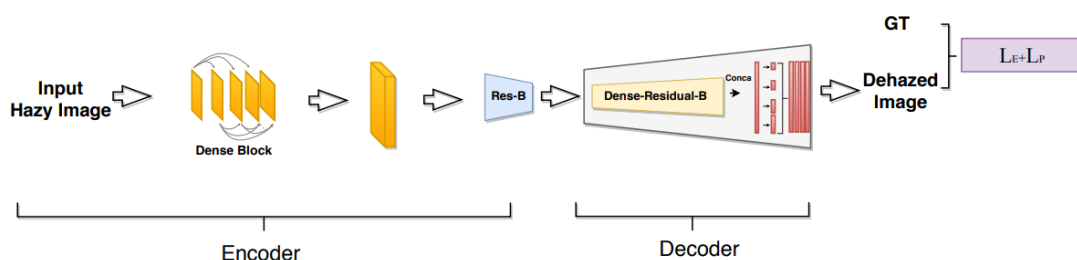


Figure 2. Overview of the proposed Multi-scale Single Image Dehazing using Perceptual Pyramid Deep Network.

這篇 **work** 採用的是較為直觀的 **Autoencoder** 架構，在 **encoder** 的部份，利用 **DenseNet** 的架構，以及 **pretrain** 好的模型參數，使模型能夠參考不同 **layer** 之間的 **feature**，達到性能的提昇，同時降低 **gradient vanishing** 的問題。而在 **decoder** 的部份，一樣是採用 **DenseNet** 的連接方式，且預測出不同大小(**scale**)

的圖片，再進行 ensemble，得出最後的結果。

然而由於圖片大小不一且過大，所以輸入模型的圖片皆切為固定大小的較小圖片，預測完再進行拼裝回原本的大小。

我們基於這篇的結果，進行許多階段的後處理(post-processing)，以下將一一列舉：

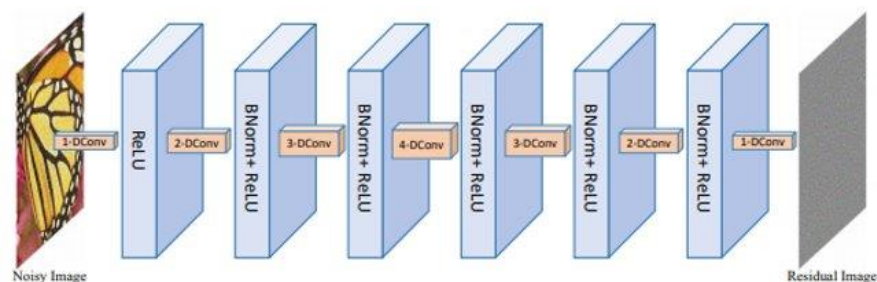
1. Patch ensemble

這篇 paper 的 inference 方式相當特別，為將 testing image 切成許多互相重疊的 patch，將 patch 通過 model 後在合成一張圖，藉以達到 ensemble 的效果。經過實驗後，我們對原 paper 的合成方法進行了改良。首先，我們將 input patch 做 flip 的動作，得到多一倍的 patch，進一步增強 ensemble 的效果。其次，我們並非跟原 paper 一樣，將水平方向重疊的 patch 和垂直方向重疊的 patch 各自合成一張圖做平均，而是直接將所有 patch 合成一張圖。接著，我們又實驗了不同的 resize 演算法，最後發現 cubic 的效果遠比其它方法好。最後，我們發現在將圖片存檔之前，是否有作 clip 和 datatype 轉換也會影響結果，推測是 skimage 有自己特殊的 datatype 轉換方式，所以如果沒有先轉成 uint8，就會使用到 skimage 內建的方法，造成結果不同。

2. 對特定圖片進行特殊處理

- histogram matching
- 調整色調

3. IRCNN postprocessing network

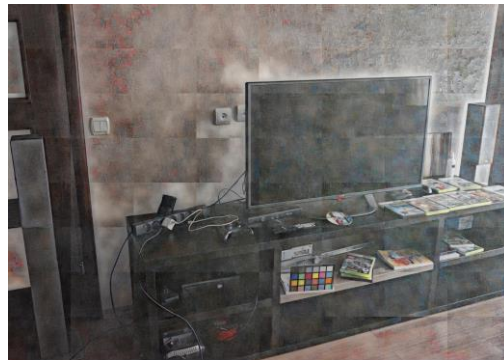
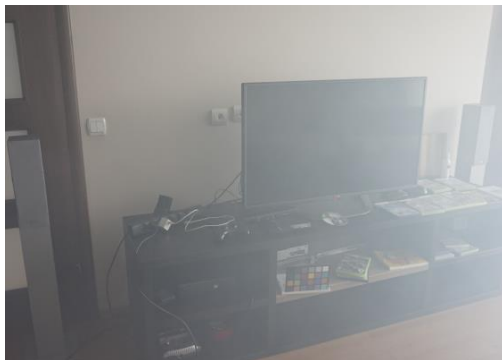


這個 model 是我們在 AtJ-DH network 的 paper 中看到的，原本是功能用來做 image denoising。由於在做完 dehazing 和 patch ensemble 後，有 noise 是非常合理的，所以可以預期將 output image 通過這個 model 能夠提升分數。

Experiment and Discussion

1. 使用 AtJ-DH 後結果如下：

- 室內圖片（圖片 2）：



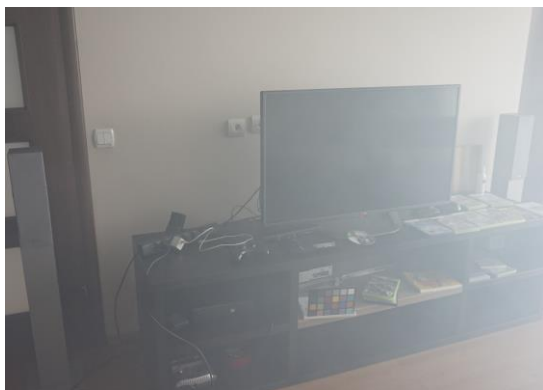
- 室外圖片（圖片 8）：



對圖片使用此 model 後，發現室內圖片結果看起來比較不真實，格線也很明顯；室外圖片結果看起來比較自然，但整體畫面還是糊糊的。這大概是因為這篇 paper 使用的是 DENSE HAZE 這個 dataset。相較於本次 project 的 dataset，DENSE HAZE dataset 的 haze 濃了許多，所以這個 model 在 haze 沒有那麼濃的 dataset 上表現就不太好。10 張測試圖片的分數(PSNR*SSIM 結果平均)為 14.148504 。

2. 使用 Multi-scale Single Image Dehazing using Perceptual Pyramid Deep Network 的結果如下：

- 第 2 張 testing image



testing image



output image

此圖片為室內圖，output image 非常清晰，格線不明顯。

針對特殊圖片進行特殊處理：

- 第 7 張 testing image



testing image



output image

此圖為室外圖片，產生出的圖片格線不明顯，且輪廓與色調較為自然。但我們在 training data 中找到與此圖片相近的場景(Training data 5)，對比之下，可以明顯看出 output image 的色調不夠鮮豔，有些蒼白。為了解決這個問題，我們以 training data 5 為 reference，對原 output image 做 histogram matching，使圖片整體色調與 reference image 更接近。



Training data 5 (reference image)



After histogram matching

- 第 10 張 testing image



testing image



output image

此圖為室外圖片，產生的圖片輪廓清晰。但與我們在 training data 中找到與此圖片相近的場景(Training data 25)相比，會發現色調偏藍，這大概是因為原本 testing image 的 haze 就偏藍的緣故。由於 training data 25 的主要顏色分布與 testing image 有很大不同，直接使用 histogram matching 會讓顏色髒掉，故我們採用了不同的解決方法。我們先將 output image 三個 channel 的 mean 求出，並將它們分別除以一個手動調的參數(我們選擇 85)，藉此得到三個 shrinking factor，再將 output image 的三個 channel 分別除以各自的 shrinking factor。這樣能夠讓三個 channel 的 mean 變成相同的值，便能解決色調偏藍的問題。



Training data 25



調整色調後

最後結果：

將 testing image 輸入 model 的 output 分數為 20.385703。對第 7,10 張圖片的處理後，分數為 20.632757。最後再使用 IRCNN postprocessing network 後，分數為 20.669156，IRCNN postprocessing network 可以讓圖片變更好，但是效果比較小。

Conclusion

在這次的期末專題當中，我們共同研讀了許多前人的方法及論文，模型一個疊的比一個還深，GPU 一個比一個還高級，經過反覆實驗之後，挑出了一個表現相當好的模型。如果要從頭到尾自己重新來過，我們認為達到更高的表現機率並不高，因此最關鍵的問題便是如何基於現有的成果，再更上一層樓。或許可以更進一步的 fine tune 參數，然而我們採取了更特別的路，利用傳統方式增進效能，雖然難免在一些部份會有針對性，但是我們發現了這些傳統方式是確實有效的，或許可以利用更自動化的方式(學一個模型去做這件事?)來達到這樣的效果，這也是未來可以更進一步研究的方向。

Reference

1. Dense Scene Information Estimation Network For Dehazing <https://reurl.cc/d32jV>
2. Multi-scale Single Image Dehazing using Perceptual Pyramid Deep Network
http://openaccess.thecvf.com/content_cvpr_2018_workshops/papers/w13/Zhang_Multi-Scale_Single_Image_CVPR_2018_paper.pdf
3. Learning Deep CNN Denoiser Prior for Image Restoration
http://openaccess.thecvf.com/content_cvpr_2017/papers/Zhang_Learning_Deep_CNN_CVPR_2017_paper.pdf