

GAI Project4 Report

資訊 113 F74094025 林昱崑

以下是本次作業的 github 連結：

https://github.com/JasperLin0118/GAI_HW4

模型架構與訓練過程介紹

首先，我選擇實作本次作業說明給的 Example2: Guiding DIP Early Stopping with DDPM-inspired Supervision。針對原本的 DIP 架構，從原本只針對一張有 noise 的圖片去對它做特徵學習，我將這個架構結合了 DDPM Forward Diffusion 的不同 Stages，也就是讓機器去學習不同程度 noise 的 target image，目的是讓它能夠分辨哪些是這張圖片原有的特徵（以圖一為例，圖片原有的特徵就是飛機、背景是雪白的山等等），哪些是噪音（以圖一為例，圖片中的黑色小點點就是噪音）。從不同 stage 中，去學習圖片中哪些特徵是不變的，哪些特徵是會隨著 stage 不同而改變的，那會改變的那些特徵就是 noise，也就是我們想要去掉的部分。



圖一、左邊的圖為無 noise 原圖，右邊為有 noise 的原圖

在每個 stage 中，我讓 model 都 train 30 個 epoch，在每個 epoch 中，我使用 Mean Square Error 來算 loss，然後使用 PSNR 來跟該 stage 的 target image 做比較。如果在這個 stage 中 PSNR 有 3 次沒有上升的話，我就做 early stopping。

實驗驗證

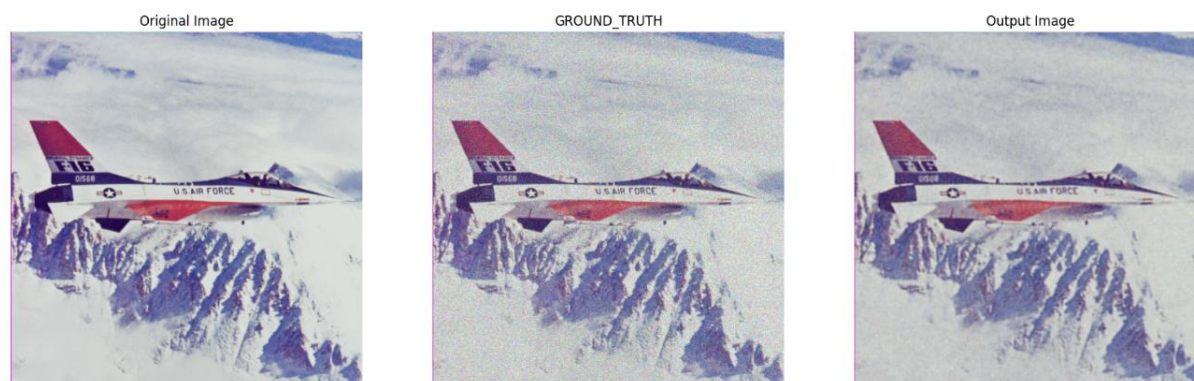
在本次作業中，我總共測試了兩種不同架構的 DIP 模型，分別是 CNN 和 UNet。在 CNN 的部分，我還有測試不同數量的 layer (32、64) 來觀察訓練效果；UNet 的部分，我則是固定它的 layer 數量，並以不同的 denoising stage 數量來觀察模型訓練效果。我將 input image 的大小都固定在 512x512，原本有想要調整圖片大小來觀察結果，但是我發現從 256 x 256 的大小開始再縮小的話，圖片會很明顯地看的到像素的塊狀，也就是非常模糊，調大的話 (1024x1024)，記憶體容量會不夠，因此我就決定不把它當作變量來調整了。

以下是我實驗的結果截圖和與原圖的 PSNR 值，包含最後幾個 stage 生成的圖片、原圖、Ground_Truth 和生成的圖片結果、Training Loss 的折線圖、每個 stage 最好的 PSNR 折線圖：

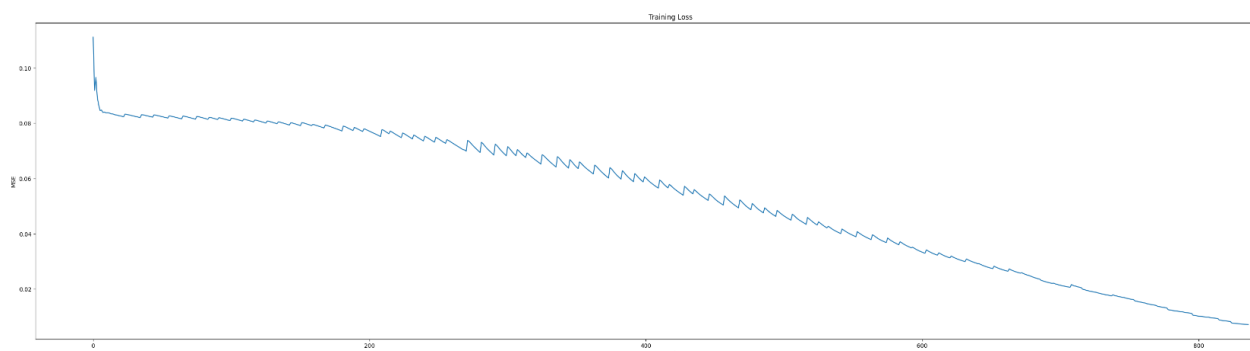
1. Guiding DIP Early Stopping with DDPM.ipynb (CNN model with 32 layers)



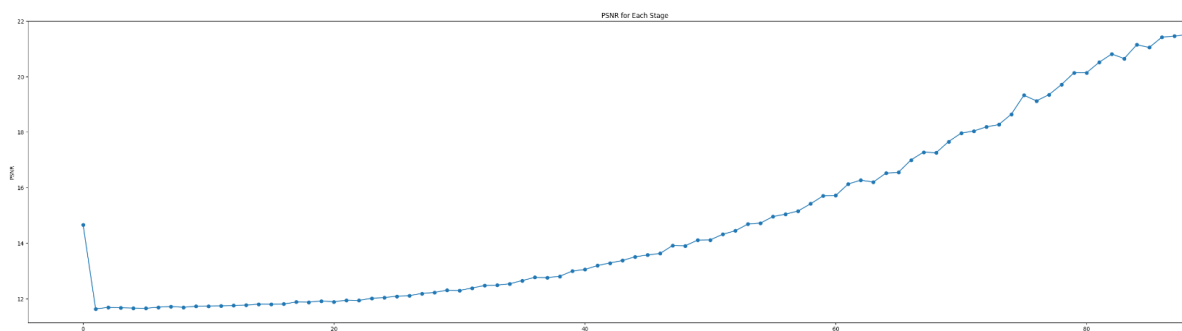
圖二、最後 10 個 stage 生成的圖片，由左至右是最後 1 個到 10 個



圖三、原圖、Ground Truth、生成的圖片結果



圖四、Training Loss

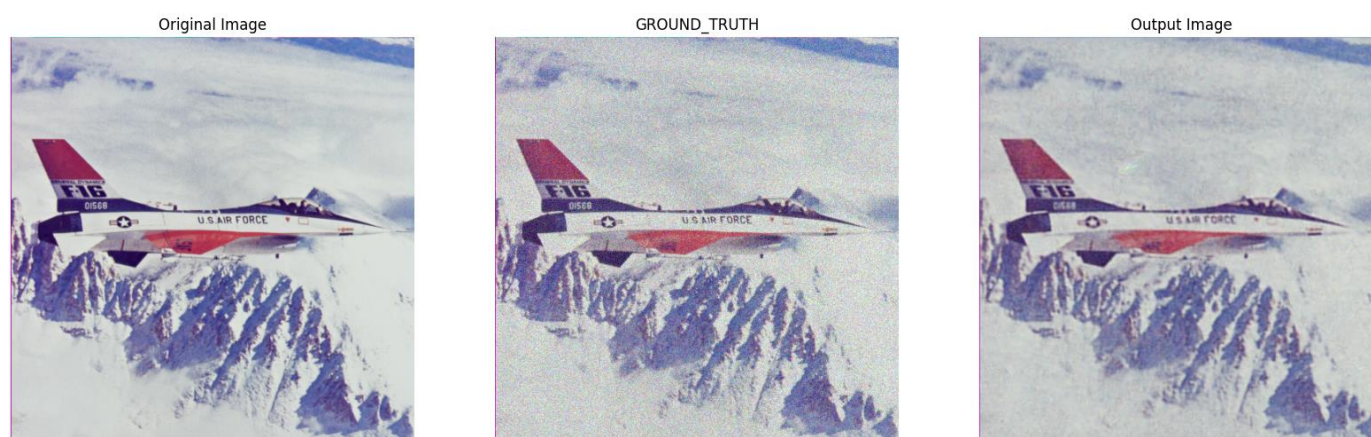


圖五、每個 stage 最好的 PSNR
與原圖相比的 PSNR：20.11034920305614

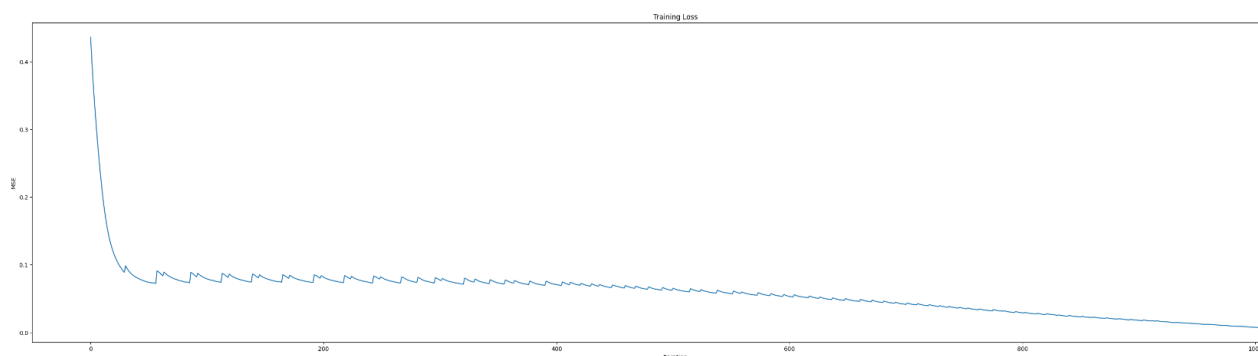
2. Guiding DIP Early Stopping with DDPM2.ipynb (CNN model with 64 layers)



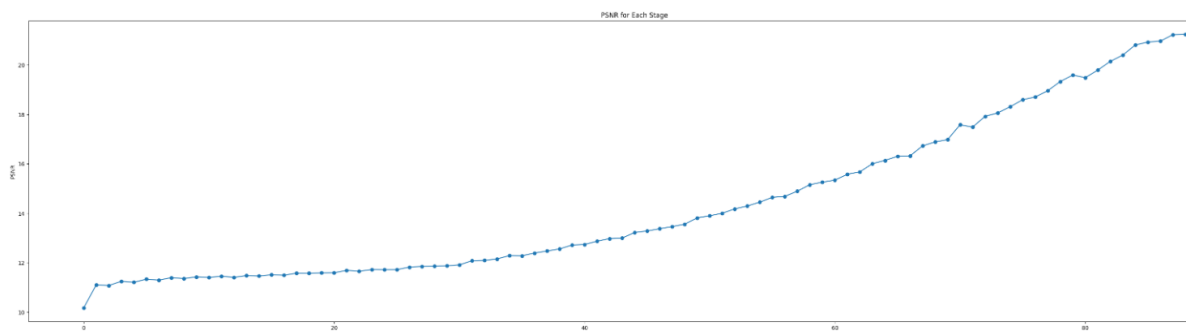
圖六、最後 10 個 stage 生成的圖片，由左至右是最後 1 個到 10 個



圖七、原圖、Ground Truth、生成的圖片結果



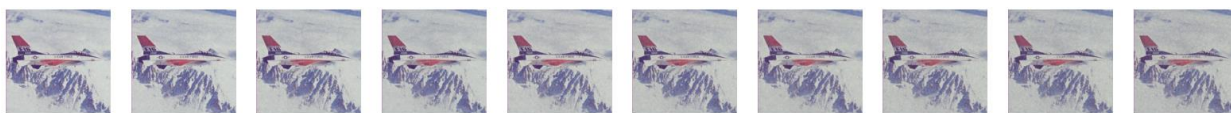
圖八、Training Loss



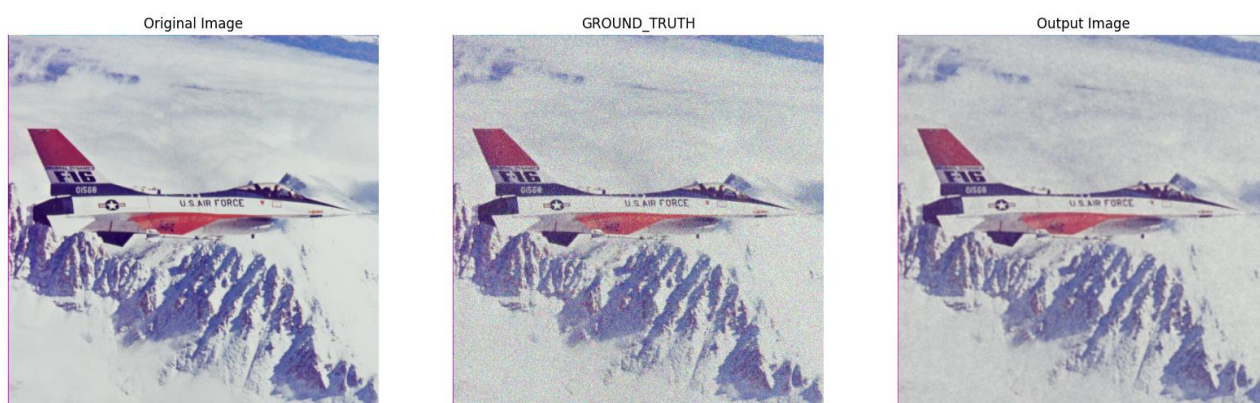
圖九、每個 stage 最好的 PSNR

與原圖相比的 PSNR：20.06573887125522

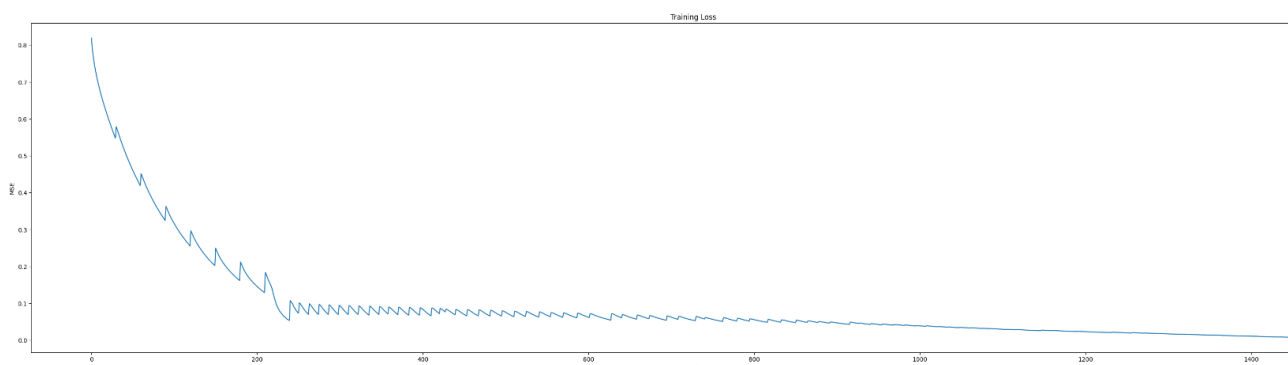
3. Guiding DIP Early Stopping with DDPM3.ipynb (UNet model with 90 stages)



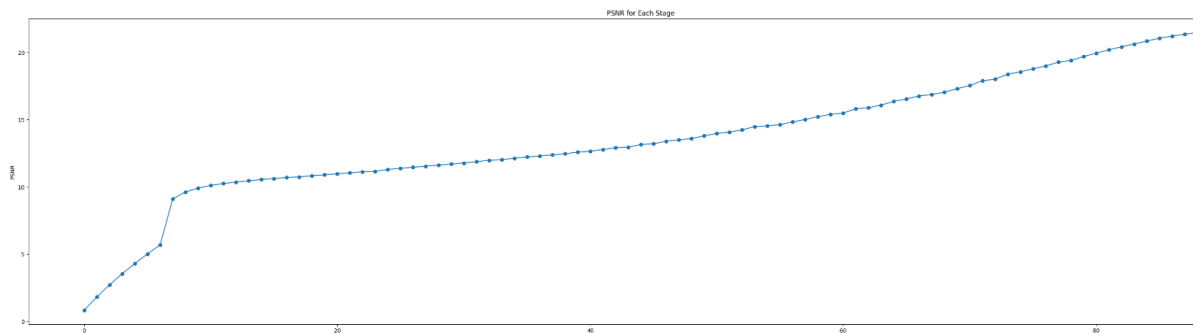
圖十、最後 10 個 stage 生成的圖片，由左至右是最後 1 個到 10 個



圖十一、原圖、Ground Truth、生成的圖片結果



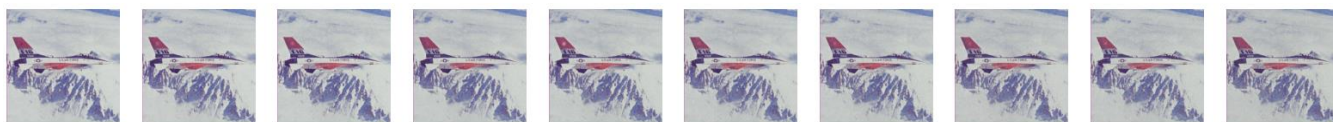
圖十二、Training Loss



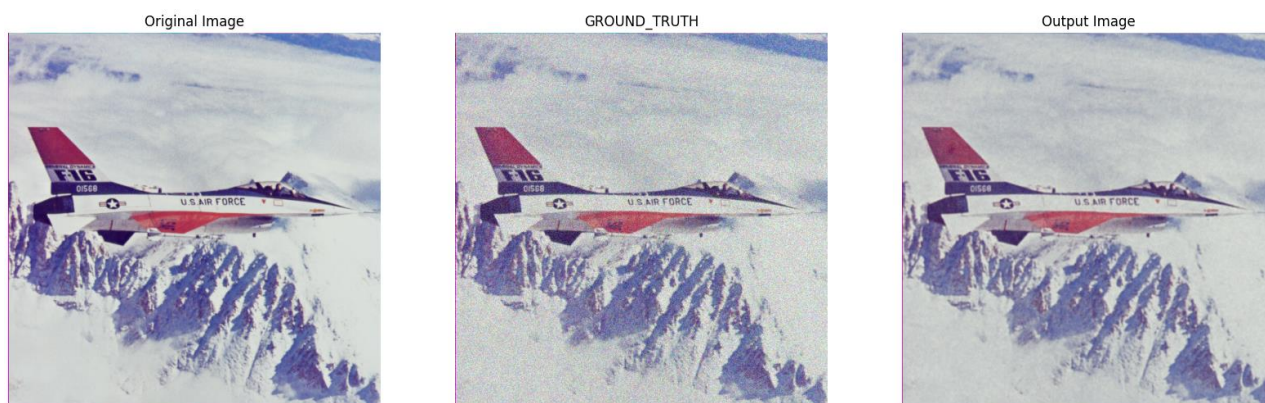
圖十三、每個 stage 最好的 PSNR

與原圖相比的 PSNR：20.1212163243743

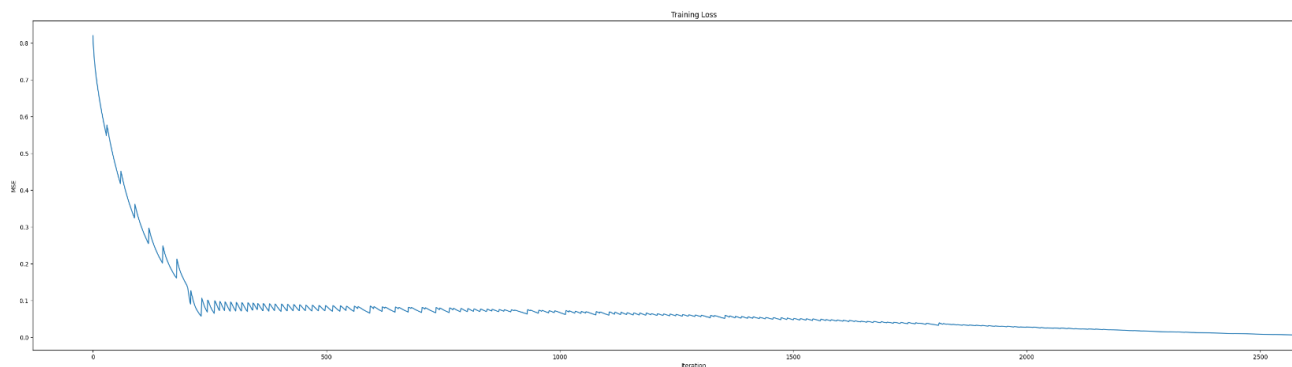
4. Guiding DIP Early Stopping with DDPM4.ipynb (UNet model with 180 stages)



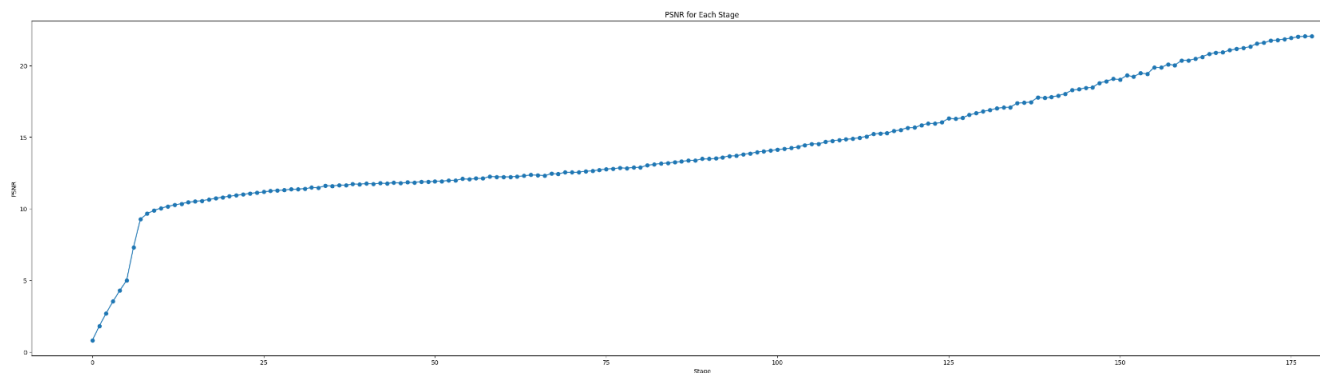
圖十四、最後 10 個 stage 生成的圖片，由左至右是最後 1 個到 10 個



圖十五、原圖、Ground Truth、生成的圖片結果

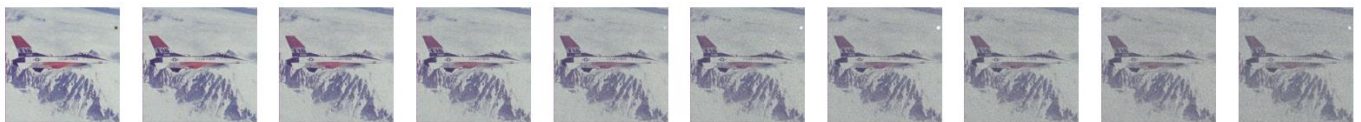


圖十六、Training Loss

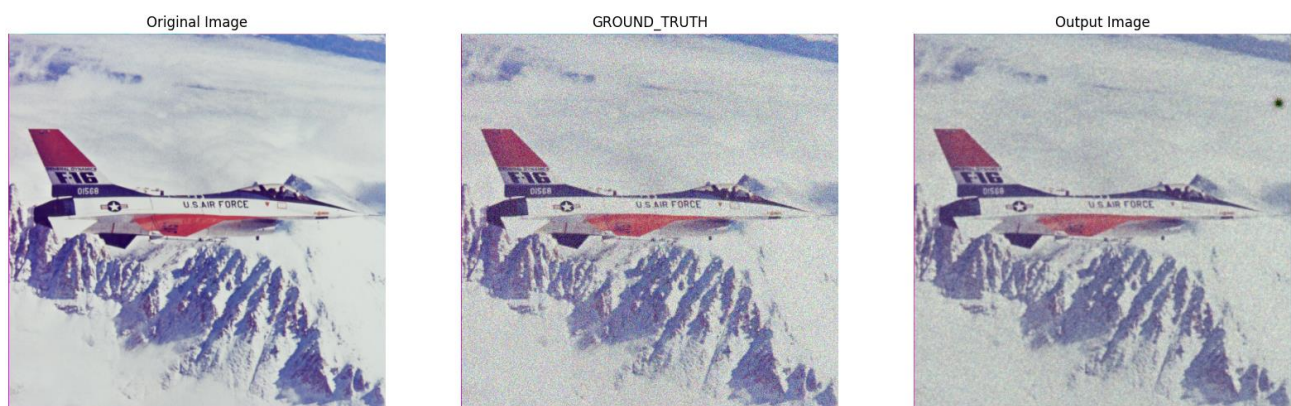


圖十七、每個 stage 最好的 PSNR
與原圖相比的 PSNR：21.046787776254064

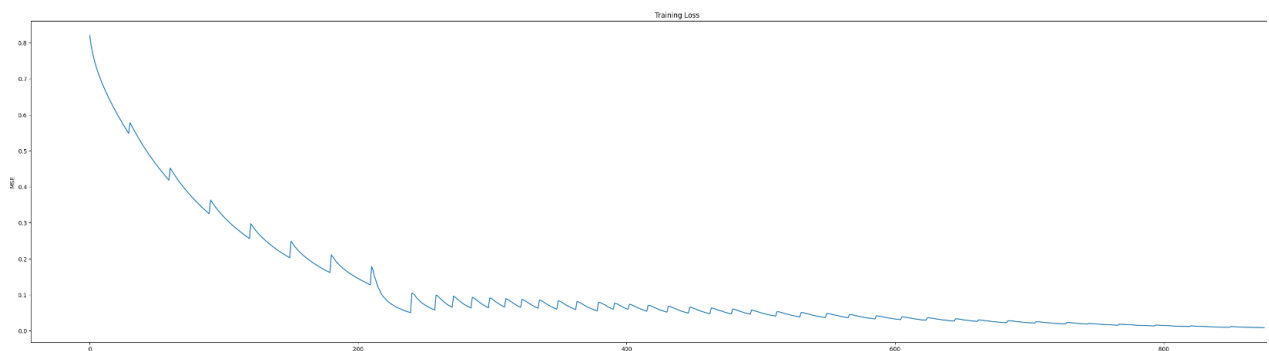
5. Guiding DIP Early Stopping with DDPM5.ipynb (UNet model with 45 stages)



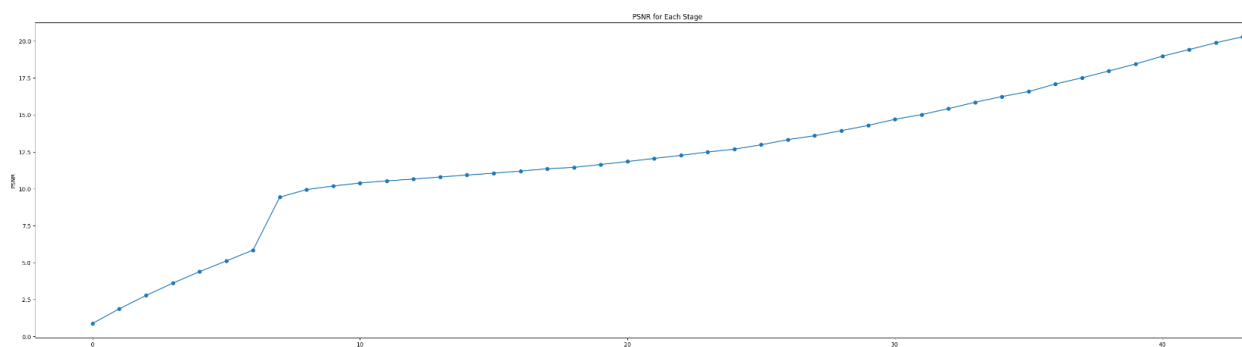
圖十八、最後 10 個 stage 生成的圖片，由左至右是最後 1 個到 10 個



圖十九、原圖、Ground Truth、生成的圖片結果



圖二十、Training Loss



圖二十一、每個 stage 最好的 PSNR

與原圖相比的 PSNR：18.54290014521502

6. DIP Baseline.ipynb

與原圖相比的 PSNR：18.30681979572961

分析與總結

從以上測試結果來看，基本上 PSNR 都比原本的 baseline 還要高，以 CNN 架構來看，我認為理論上 layer 越多，能夠取到的特徵就越多，更容易找到噪音的特徵，進而提升去噪品質，但經過測試，越多 layer 並沒有太大的變化。以 UNet 架構來看，PSNR 的值只比 CNN 架構略微高一些，最主要的差別還是 denoising stage 的數量，從圖十一、圖十五及圖十九來看，stage 數量越少，越能明顯看出 denoise 品質的好壞，還有圖的模糊程度。跟原圖對比的話，看得出噪音的那些黑點有變不見，圖變得比較平滑，有可能是因為我把圖片所小，因此生成的圖片比較模糊，如果能夠增大圖片大小（硬體資源不足），應該就能解決問題了。

從生成速度或是 training 速度來看，我認為時間是差不多的，因此沒有特別測量。總結來說，DIP 結合 DDPM 的 forward diffusion stages 對於 denoising 的效果是有提升的，主要從 forward diffusion stages 數量下手，還有調整圖片大小，應該就有明顯的效果了。