Homework 4:
Image Restoration with PromptIR: Rain and Snow Removal

Marie Picquet
Student ID: 313551818
National Yang Ming Chiao Tung University
Selected Topics in Visual Recognition
using Deep Learning (535521)

May 28, 2025

1 Introduction

The goal of this assignment is to develop an image restoration model capable of recovering clean images from degraded inputs affected by rain and snow. This problem is framed as a supervised learning task, where pairs of degraded and clean images are used to train a convolutional neural network (CNN) to remove weather-induced distortions.

Two different approaches are explored: PromptIR, a transformer-based architecture utilizing degradation-specific prompts [1], and PromptIRLite, a simplified convolutional neural network that avoids prompt-based modules. Model performance is evaluated using the Peak Signal-to-Noise Ratio (PSNR), a standard metric for image reconstruction quality.

Model training was conducted using GPU-accelerated environments in Google Colab, with custom preprocessing pipelines to manage the two degradation types. The final restored outputs were compared against ground truth images to measure restoration accuracy.

The complete code for this assignment is available at the following GitHub repository: `https://github.com/mariep00/535521-HW4.git`.

2 Method

Dataset

The dataset used in this assignment focuses on image restoration under two types of degradation: rain and snow. For each degradation type, the dataset provides paired images, degraded and corresponding clean versions, allowing supervised training of restoration models.

Image type	RGB images
Degradation types	Rain, Snow
Training/Validation set	1600 degraded + clean images per type split into 90/10
Test set	50 degraded images per type (no ground truth)

Table 1: Dataset summary

Examples from the dataset are shown in Figure 1. On the left, degraded images affected by rain or snow are shown; on the right, the corresponding clean images are provided as targets. Each degradation type is treated jointly by a single restoration model.

(a) Degraded (Rain)

(b) Clean Target

Figure 1: Example image pair from the train dataset. (a) shows a degraded input affected by rain. (b) shows its corresponding clean target.

Data Preprocessing

Images were loaded from the provided directories and resized to a standard shape of 256×256. The pixel values were normalized using the ImageNet mean and standard deviation (i.e., mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225]), which is standard practice for pretrained transformer-based backbones. The images were then converted into PyTorch tensors using standard ‘ToTensor()‘ and ‘Normalize()‘ transforms. Input images were paired according to naming conventions (e.g., `rain-xxx.png` → `rain_clean-xxx.png`).

2.1 Model: PromptIR

The full restoration model used in this project is PromptIR, a transformer-based architecture that is designed to handle multiple types of image degradation within a single unified framework.

Unlike traditional deep learning models, which are often specialized for specific tasks like denoising or deblurring, PromptIR introduces a flexible mechanism based on prompts. Prompts are learnable vectors that encode important information about the degradation present in an image (e.g., rain, snow, haze, etc.).

PromptIR consists of three key components:

- **Prompt Generation Module (PGM)**: Learns a set of degradation-aware prompt vectors from the input features.
- **Prompt Interaction Module (PIM)**: Injects these prompts into various transformer layers to condition the restoration process based on the type of degradation.

- **Transformer Backbone**: A deep feature extraction and restoration pipeline that uses skip connections and multi-scale feature fusion to reconstruct high-quality images.

This design allows PromptIR to adaptively restore images affected by unknown or mixed degradations within a single-stage network, without the need for separate classifiers or task-specific models.

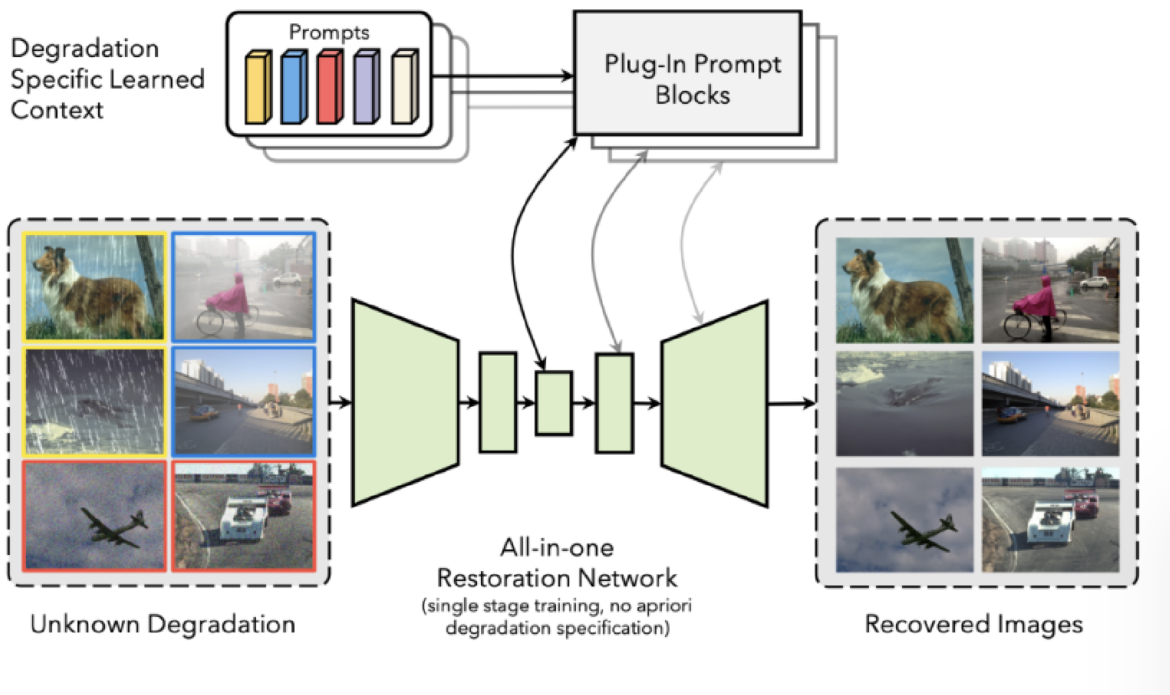

Figure 2: PromptIR handles various degradation types like rain, snow, and haze using learned prompts.

Training Hyperparameters

- Optimizer: Adam
- Learning Rate: $5e^{-4}$
- Batch Size: 4
- Epochs: 20

Loss Function

We used the Charbonnier loss, a smooth approximation to the L1 loss:

$$\mathcal{L}_{charb} = \sqrt{(y - \hat{y})^2 + \epsilon^2}$$

where ϵ is a small constant (e.g., 10^{-3}) for numerical stability. This loss is robust to outliers and more effective for image restoration compared to MSE, which was initially used [3].

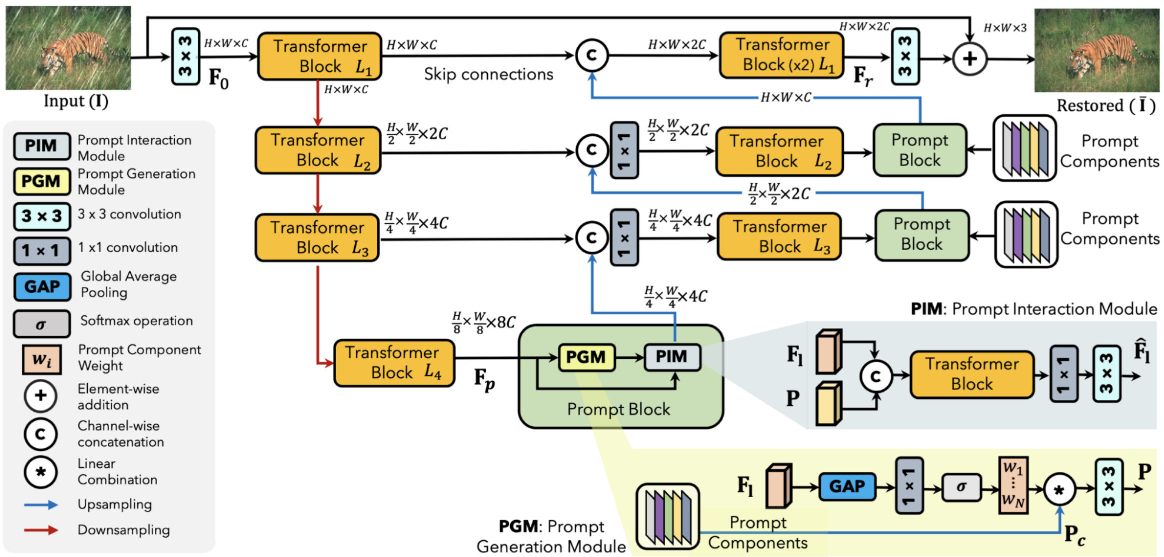

Figure 3: Detailed PromptIR architecture with Prompt Generation Module (PGM), Prompt Interaction Module (PIM), and transformer blocks.

2.2 Extra Experiment: PromptIRLite

Motivation

To evaluate whether the full complexity of PromptIR is necessary for effective rain and snow image restoration, a lightweight baseline model named **PromptIRLite** was implemented. The motivation is the following:

- **Understanding**: By removing prompt-specific modules and transformer blocks, we aim to understand the core capability of convolutional layers in this task.
- **Efficiency**: PromptIR involves prompt generation, interaction, and multiple transformer layers, which increase model size, training time, and memory consumption. A lighter architecture might perform comparably on simpler degradations like rain or snow, where spatial correlations dominate and global reasoning is less critical.

Model Architecture

PromptIRLite is a pure convolutional encoder-decoder with one intermediate processing block:

- **Encoder**: Two stacked 3×3 convolutions extract 64 feature maps from the input RGB image.
- **Middle Block**: A third 3×3 convolution further refines encoded features.
- **Decoder**: A final 3×3 convolution reduces features back to 3 channels, followed by a `Sigmoid` activation to normalize output to $[0, 1]$.

Comparison to PromptIR

PromptIRLite is a much simpler model compared to PromptIR. While PromptIR uses special learned vectors called prompts to help the model understand what type of degradation (rain or snow) it's dealing with, PromptIRLite does not use any such prompts.

PromptIR includes extra components like the Prompt Generation Module and Prompt Interaction Module, as well as transformer blocks that help the model look at the image globally and understand long-range relationships. These make PromptIR more powerful, especially for complex cases.

On the other hand, PromptIRLite is made up of only standard convolutional layers. It processes the image locally, without looking at the whole image context, and does not know what type of degradation it's restoring. Because of this, PromptIRLite is much smaller, faster, and easier to train, but it might not perform as well on more difficult restoration tasks.

3 Results

3.1 PromptIR

Figures 4 and 5 show sample outputs from the trained full model, including degraded inputs, predicted restored images. The restoration results demonstrate effective removal of rain and snow, although fine textures and lighting consistency could be further improved.

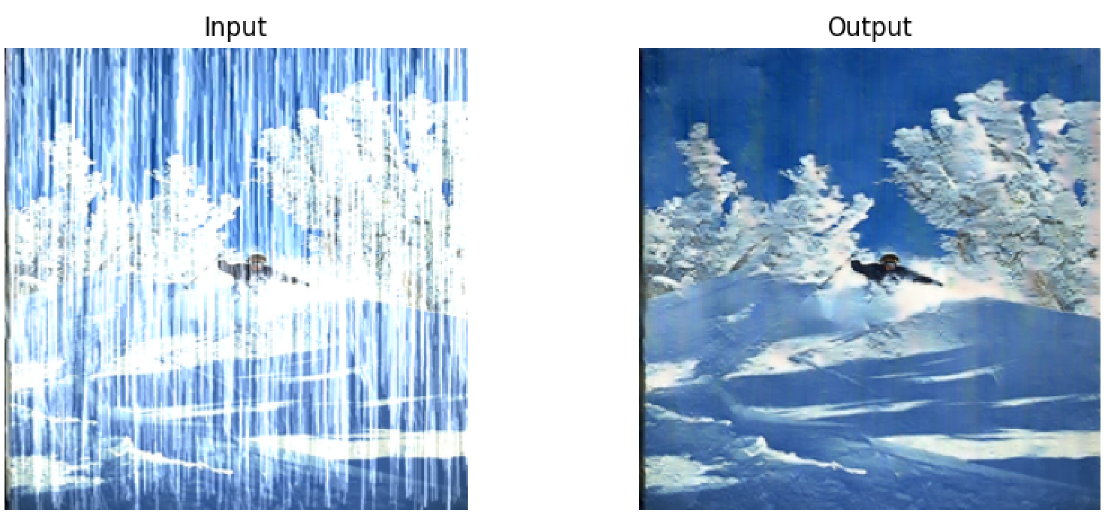

Figure 4: Visualization of degraded input and predicted restoration(output) for rain.

The model achieved a validation PSNR of 28.24 dB after 20 epochs on coda bench. The loss curve over epochs are presented in Figure 6.

3.2 PromptIRLite

Figures 7 and 8 show sample outputs from the PromptIRLite model. While the results demonstrate effective removal of visible degradation, some finer structures and lighting consistency were not fully recovered compared to the full PromptIR model.

The PromptIRLite model achieved a validation PSNR of around 20 dB after 20 epochs. Figure 9 presents the corresponding training and validation metrics.

Figure 5: Visualization of degraded input and predicted restoration (output) for snow.

3.3 Conclusion

PromptIRLite dramatically reduces model complexity and memory usage. It performs reasonably well on basic degradations, but struggles with more challenging noise patterns.

In comparison, PromptIR achieves consistently better results, with higher PSNR and better visual quality in the restored images. This is likely due to its use of prompts, transformers, and multi-scale feature fusion [1]. The model is more effective at capturing global structures and adapting to different types of degradations.

This comparison suggests that while PromptIRLite is suitable for fast inference or constrained environments, PromptIR remains the better choice for high-fidelity image restoration tasks.

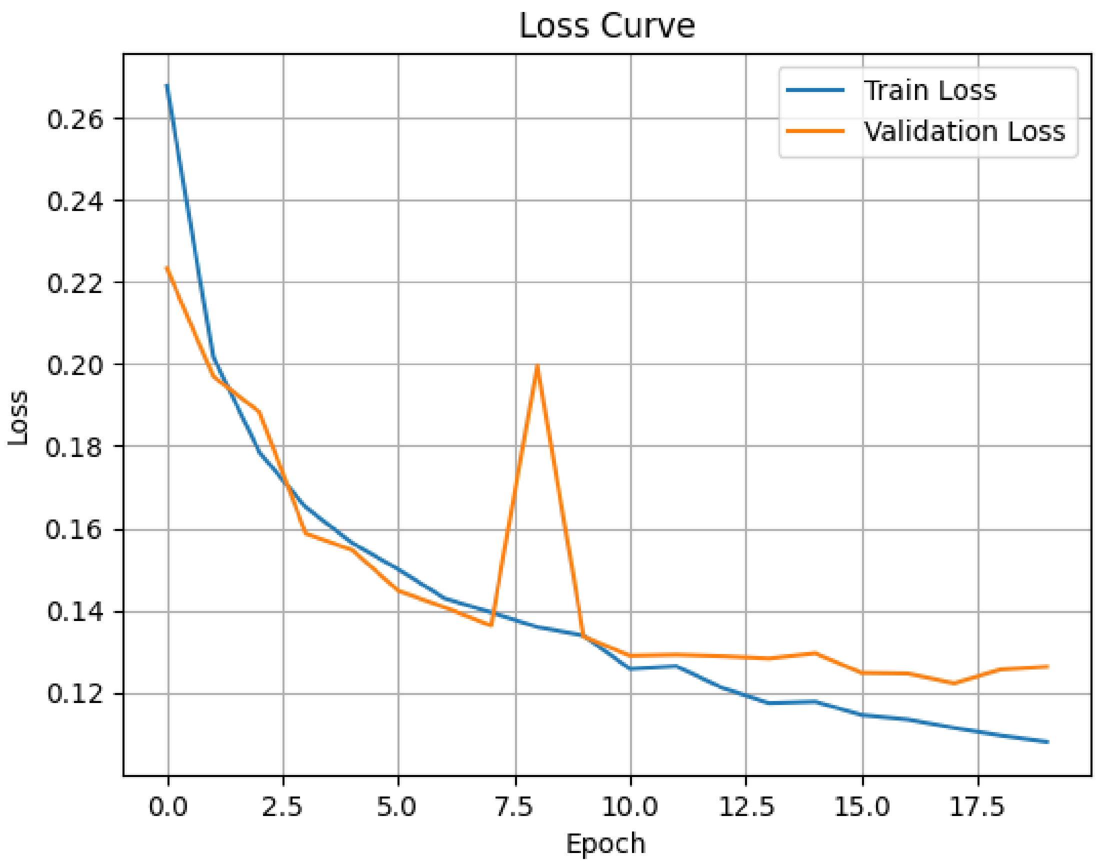

Figure 6: Training and validation loss / PSNR over 20 epochs.

References

[1] Li, Y., Pan, J., Zhang, W., Zuo, W., and Loy, C.C. (2023). *PromptIR: Prompting for All-in-One Blind Image Restoration*. arXiv preprint arXiv:2303.07353.

[2] PromptIR GitHub Repository. `https://github.com/ugatr120/PromptIR`.

[3] Barron, J.T. (2019). *A General and Adaptive Robust Loss Function*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[4] Xu, J., Wang, Y., Sun, J., Zeng, Y., and Zhang, L. (2022). *AirNet: Towards Fully Disentangled Representations of Image Corruptions*. arXiv preprint arXiv:2204.00442.

[5] He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). *Mask R-CNN*. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

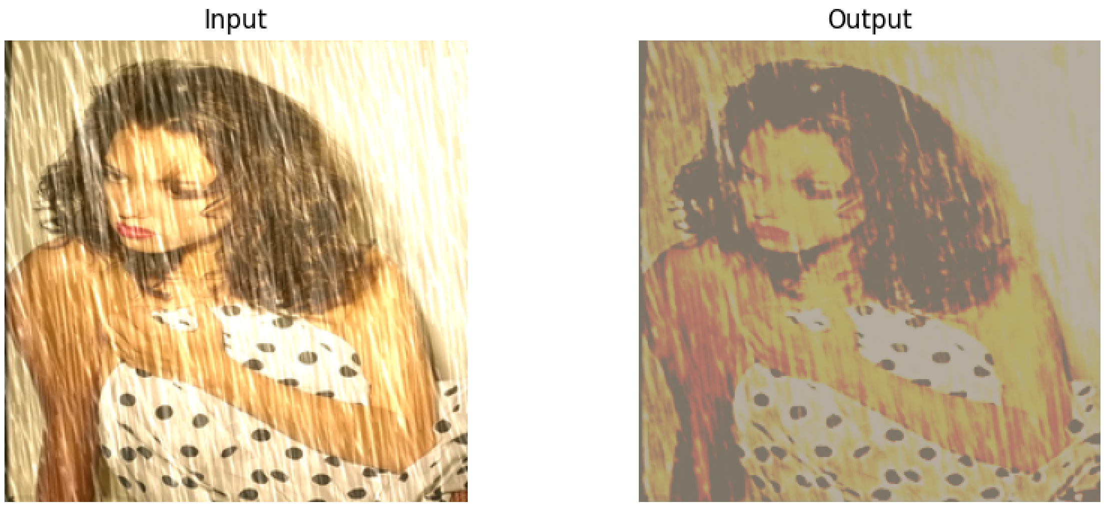

Figure 7: PromptIRLite output for rain-degraded image.

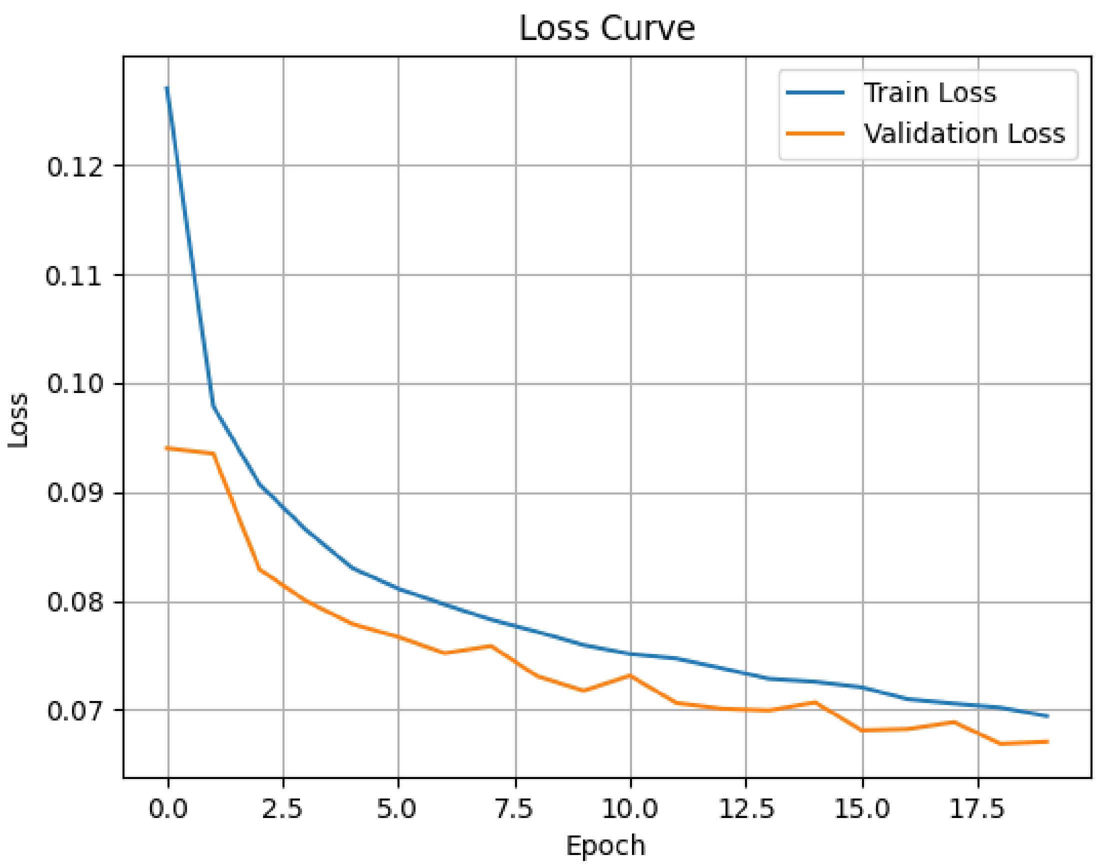

Figure 8: PromptIRLite output for snow-degraded image.

Figure 9: PromptIRLite training and validation loss / PSNR over 20 epochs.