NTIRE 2025 Challenge on Single Image Reflection Removal in the Wild: Supplementary Material

Kangning Yang* Jie Cai*† Ling Ouyang* Florin-Alexandru Vasluianu* Radu Timofte*
Jiaming Ding* Huiming Sun* Lan Fu* Jinlong Li* Chiu Man Ho* Zibo Meng*
Mingjia Li Hainuo Wang Qiming Hu Jiarui Wang Hao Zhao Jin Hu
Xiaojie Guo Mengru Yang Kui Jiang Jin Guo Yiang Chen Junjun Jiang
Jing He Yiqing Wang Kexin Zhang Licheng Jiao Lingling Li Fang Liu
Wenping Ma Zhiyang Chen Hao Fang Wei Zhang Runmin Cong
Dheeraj Damodhar Hegde Jatin Kalal Nikhil Akalwadi Ramesh Ashok Tabib
Uma Mudenagudi Yu-Fan Lin Chia-Ming Lee Chih-Chung Hsu Mengxin Zhang
Xiaochao Qu Luoqi Liu Ting Liu Jinshan Chen ShuYu He Sabari Nathan
K Uma A Sasithradevi B Sathya Bama S. Mohamed Mansoor Roomi
Bilel Benjdira Anas M. Ali Wadii Boulila Wei Dong Yunzhe Li Ali Hussein
Han Zhou Jun Chen Zeyu Xiao Zhuoyuan Li

1. Other Teams

1.1. Okkk

Description: Our reflection removal approach improves the Reversible Decoupling Network (RDNet [22]) to enhance the effectiveness of single-image reflection removal.

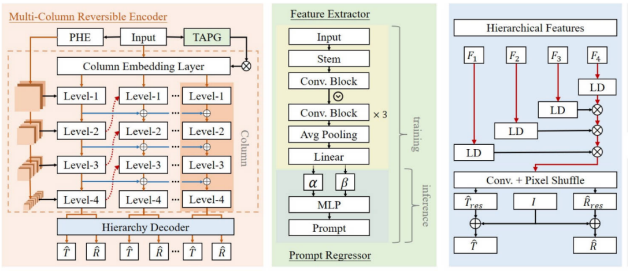


Figure 1. The overall structure of RDNet

RDNet introduces a Reversible Decoupled Network (RDNet) architecture, which employs a reversible encoder to preserve critical information while flexibly decoupling features related to the transmission layer and reflection layer during forward propagation. Additionally, RDNet incorporates a transmission rate-aware prompt generator to dynamically calibrate features, further enhancing performance. RDNet's design overcomes two key challenges:

1. Loss of high-level semantic information: As features propagate through layers, high-level semantic cues tend to be compressed or discarded, affecting the quality of reflection removal.
2. Limitations of fixed interaction patterns: The interactions between layers in dual-stream networks typically follow a fixed pattern, restricting overall model performance.

Therefore, we choose RDNet for our Baseline, and optimize the design of experiments to better achieve single-image reflection removal.

The overall structure of RDNet is shown in Fig. 1, which consists of three main core modules: the multi-column reversible encoder (MCRE), transmission-rate-aware prompt generator (TAPG) and the hierarchy decoder (HDec).

The Pretrained Hierarchy Extractor (PHE) extracts semantically rich hierarchical representations from the input image and passes them to each layer in the first col-

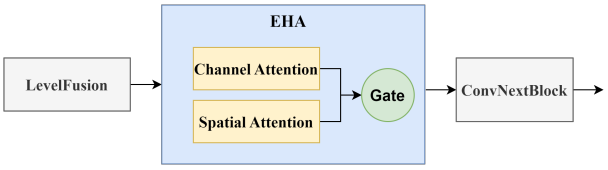


Figure 2. Enhanced Hybrid Attention

*Kangning Yang, Jie Cai, Ling Ouyang, Florin-Alexandru Vasluianu, Radu Timofte, Jiaming Ding, Huiming Sun, Lan Fu, Jinlong Li, Chiu Man Ho and Zibo Meng were the challenge organizers, while the other authors participated in the challenge. Each team described their own method in the report.

†Project lead

umn of the MCRE. Meanwhile, TAPG learns channel-level transmission-reflection ratio priors from the data and maps these learned parameters into cues to guide feature decoupling in the MCRE network. Finally, each column of the MCRE utilizes HDec to decode the hierarchical information while providing effective lateral guidance. Ultimately, the decoding results from the last column of the MCRE generate the final reflection removal output.

In the RDNet architecture, each layer relies solely on convolution operations to extract local features. However, due to the inherent limitations of convolutions, the model struggles to effectively capture long-range dependencies, weakening its ability to understand the global context of reflection regions. To address this issue, we propose an Enhanced Hybrid Attention (EHA) mechanism, as illustrated in Fig. 2, which optimizes feature processing through a dual-path design: channel attention dynamically adjusts the importance of feature channels to suppress irrelevant reflection information, while spatial attention enhances the precise localization of reflection boundaries. These two attention mechanisms are adaptively fused through a learnable gate mechanism, allowing the model to retain the local detail modeling capability of convolutions while significantly improving its ability to capture global information, ultimately achieving more precise reflection removal. The enhanced hybrid attention mechanism we designed has significant advantages in the reflection removal task in the following four aspects.

1. Cross-Dimensional Dynamic Feature Calibration: The Enhanced Hybrid Attention (EHA) module achieves precise multi-dimensional modeling of reflection regions through parallel channel attention and spatial attention pathways. Channel attention (global average pooling + bottleneck structure) analyzes the global statistical information of feature channels, effectively suppressing irrelevant high-frequency components related to reflections. Meanwhile, spatial attention (group convolution + pointwise convolution) focuses on local structural differences, enhancing the weighting distribution along reflection boundaries. The dynamic fusion of these two attention mechanisms is controlled by a learnable gating coefficient, enabling adaptive adjustment of the attention type ratio based on the reflection intensity of the input image. This cross-dimensional synergy overcomes the over-smoothing issue commonly seen in conventional convolutional operations when handling reflection regions, significantly improving the recognition accuracy of complex reflection patterns.

2. Hierarchical Reflection Suppression: To accommodate the multiscale nature of reflection removal, we introduce the hybrid attention module at higher network levels (level ≥ 2) . Since high-level features contain richer semantic information, incorporating attention at this stage helps distinguish background content from reflection artifacts. Specifically, we alternate convolutional blocks with attention modules (inserting one EHA after every two ConvNeXtBlocks), forming a cascaded process of local feature extraction – global context calibration. This design enhances reflection removal performance while maintaining computational efficiency.

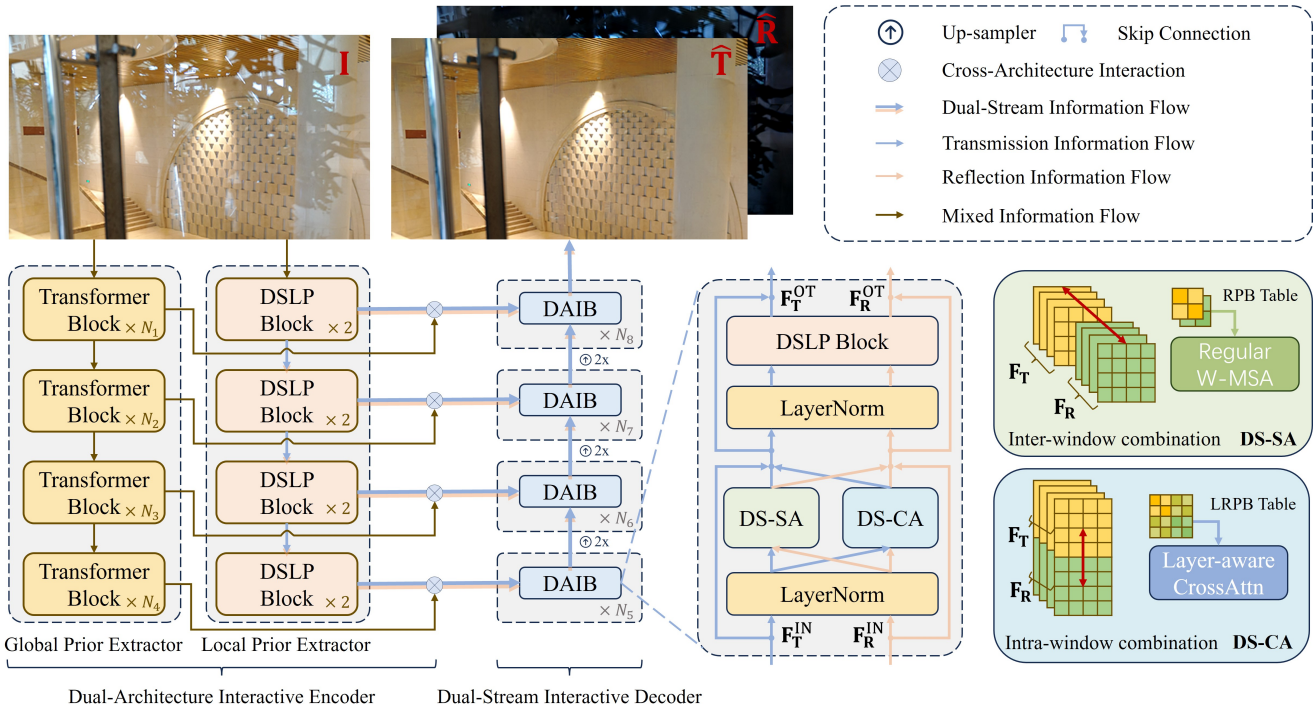
3. Efficient Attention Optimization: We optimize the attention module to enhance computational efficiency. The channel attention adopts a 4× channel compression ratio, reducing parameters to within 8% of the original convolutional block. The spatial attention employs a cascaded structure of grouped convolutions and 1×1 convolutions, preserving the receptive field of 3×3 convolutions while reducing computational cost by 75%. Additionally, the gating network predicts weights using a single-layer adaptive pooling mechanism, eliminating complex computations and ensuring efficient inference.

4. Physics-Guided Attention Learning: The unique physical properties of reflection removal, such as higher brightness and lower sharpness in reflection regions, are explicitly incorporated into the design of our attention mechanism. The channel attention module, utilizing a Sigmoid activation function, naturally suppresses high-brightness feature responses, aligning with the optical characteristics of reflections. Meanwhile, the spatial attention module employs grouped convolutions, which inherently preserve local gradient variations, making it more effective at capturing abrupt transitions at reflection boundaries.

By deeply integrating domain knowledge into the attention design, our approach enables the model to maintain stable performance even with limited training data. Furthermore, extensive cross-dataset evaluations demonstrate its superior generalization ability, effectively adapting to diverse real-world reflection patterns.

Implementation: During the training phase, we only utilized the 800 training image pairs provided by the official dataset and loaded the pre-trained weights of RDNet. These weights were pre-trained on multiple reflection removal datasets, including Real20 [21], Objects, Postcard, and Wild [17]. RDNet adopts a two-stage training strategy, where we fine-tuned the model based on the pre-trained weights. In the first stage, we froze the ConvNeXt structure and fine-tuned only the TPAG head. In the second stage, we unfroze the MCRE module for end-to-end optimization.

Specifically, we trained the model for 180 epochs on a VGPU-32GB with an initial learning rate of 1×10^{-4} . The first 100 epochs utilized a step-based learning rate schedule, followed by 80 epochs with cosine annealing. During the testing phase, we further applied Test-Time Augmentation (TTA) to enhance the reflection removal performance. Specifically, we performed horizontal flipping, vertical flipping, and 90°, 180°, and 270° rotations on the input images, conducted reflection removal for each transformation, and



(a) Dual-Stream Interactive Transformer (DSIT) **(b) Dual-Attention Interactive Block (DAIB)**

Figure 3. Dual-Architecture Interactive Encoder (DAIE) and a Dual-Stream Interactive Decoder (DSID). [8]

then fused the results to optimize the final output.

1.2. MVP Lab

Description: Our overall architecture comprises a Dual-Architecture Interactive Encoder (DAIE) and a Dual-Stream Interactive Decoder (DSID), which is shown in Fig. 3. The mixed global information is then injected into the dual-stream local flows via Cross-Architecture Interactions (CAI), ensuring comprehensive information utilization. Subsequently, the DSID separates and aggregates the embeddings hierarchically through Dual-Attention Interactive Block.

Implementation: In the training phase, we adopted the model scheme of DSIT [8], but because the dataset resolution of the competition was larger than that of the open datasets, we changed the original input resolution of 384 to 768. We trained 100 epochs on the additional open datasets as pre-training. Specifically, we used 7,643 images from the Pascal VOC dataset [7] (center-cropped as 224 x 224 slices to synthesize training pairs), 90 real-world training pairs provided by Zhang et al. [21], 200 real-world training pairs provided by IBCLN [11]. Then, 200 epochs are trained on the dataset of the competition, resulting in a weight file. In addition, we find that the quality of enhancement can be improved by replacing the pixels in the non-reflective region of the enhanced image with the original image, in which the DSIT model can obtain the corresponding non-reflective region. So we ended up with post-processing by replacing the non-reflective area. Our models are implemented using the PyTorch framework and use the Adam optimizer. The learning rate is fixed as 10^{-4} with a batch size of 1 on a single RTX 4090 GPU.

1.3. KLETech-CEVI

Description: We propose Reversible Hierarchical Reflection Removal Decoupling Network (RRR), as described in Fig. 4. In our experiment we employ RDNet [22] as our baseline model. Our proposed model retains key strengths of RDNet while introducing significant enhancements. The multicolumn reversible encoder now comprises five levels with an extra convolution block at the added level to deepen feature extraction. The column embedding layer uses a 7×7 convolution (stride 2) to generate overlapping patches F_{-1} for further processing. For the i-th column ($i \in \{1, 2, \ldots, N\}$), each level feature F_j^i (for $j \in \{0, 1, \ldots, 5\}$) is computed as:

$$F_j^i = \omega\Big(\theta(F_{j-1}^i) + \delta(F_{j+1}^{i-1})\Big) + \gamma F_j^{i-1}, \qquad (1)$$

where ω denotes the network operation as described by RDNet [22]. The Prompt Generator (PrG) is similar to that in RDNet, with the only modification being the addition of

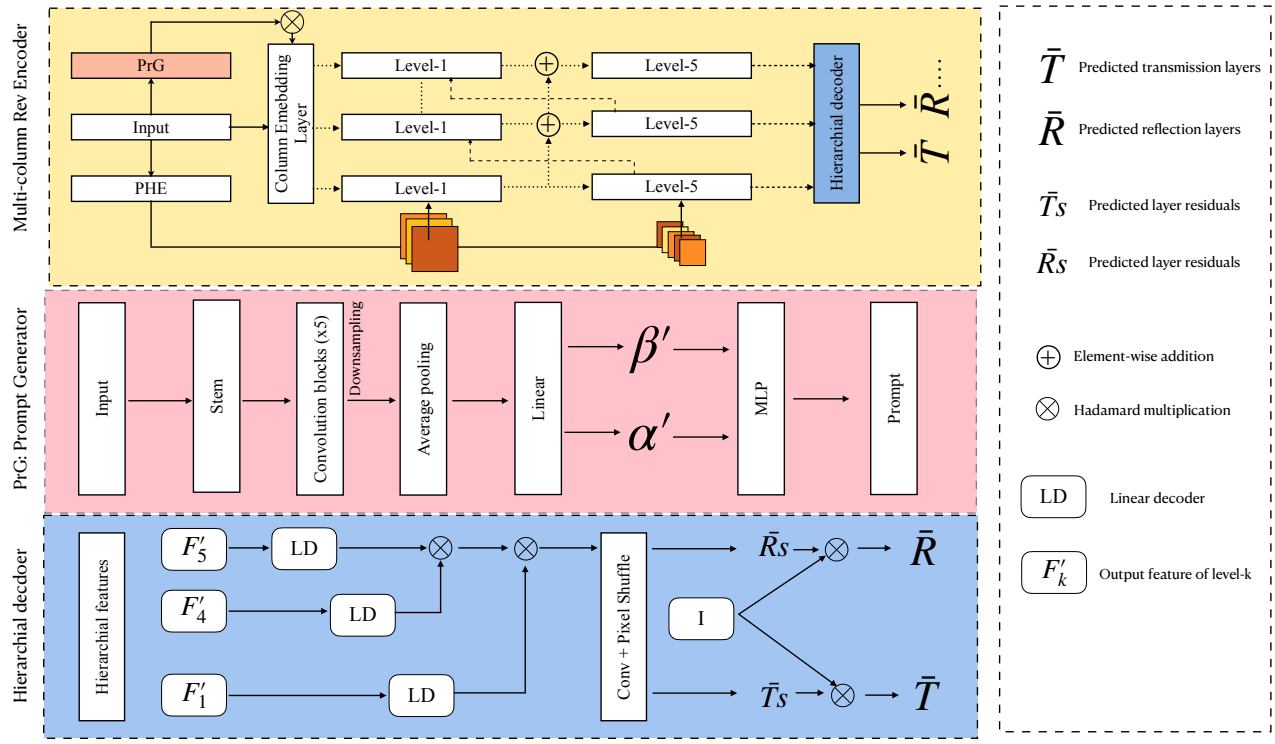


Figure 4. Our proposed framework (RRR) comprises three primary modules that collectively address reflection removal in images.

an extra convolution layer while retaining all other components. In RDNet, a simplified ConvNext model estimates six parameters $\alpha_{\{R,G,B\}}$ and $\beta_{\{R,G,B\}}$ by minimizing

$$\|\alpha_i T + \beta_i - I\|_2,$$

for each $i \in \{R, G, B\}$. A three-layer MLP then produces a prompt P, which modulates the column embeddings via the Hadamard product $P \circ F$. Finally, the hierarchical decoder fuses multiscale features using pixel-shuffle upsampling to generate layer residuals ($\bar{T}_S$ and $\bar{R}_S$), which are added to the original input to yield the final transmission ($\bar{T}$) and reflection ($\bar{R}$) outputs.

Implementation: We train our proposed method, RRR, using Python (v3.8) and the PyTorch framework. The training is executed on an NVIDIA RTX 6000 Ada generation GPU in conjunction with an Intel Xeon Gold CPU with a batch size of 1. We employ the Adam optimizer throughout the process. Initially, we use RDNet [22] as our baseline model and train it for 20 epochs. Subsequently, we fine-tune the network for an additional 80 epochs to achieve optimal reflection-free image outputs.

1.4. ACVLab

Description: As shown in Fig. 5, our solution builds upon OmniSR, proposed by Xu et al. [18]. Initially, reflection-affected input images are processed by DINOv2 [15] to extract semantic features and by Depth Anything V2 [19] to generate depth maps, from which normal maps are derived via gradient-based transformations. The original RGB image is then concatenated with the depth map to form an RGBD input, which is fed into the model. Semantic features from DINOv2 are concatenated with bottleneck features and subsequently processed by a decoder enhanced with the FreqFusion module [1]. By incorporating this frequency-aware fusion mechanism, we effectively

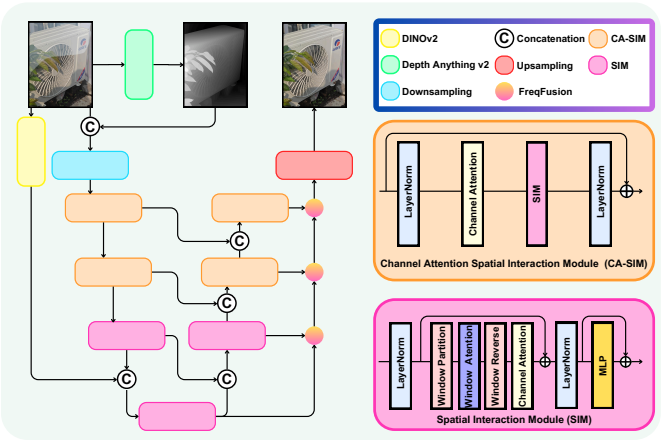


Figure 5. Network architecture of proposed solution.

recover high-frequency texture details impacted by reflections, enabling reflection-free image reconstruction, without any postprocessing.

Implementation: For the dataset preparation, we augment the training data using random cropping to 512×512 patches, along with random flipping and 90-degree rotations, before feeding it into the model for training. The training dataset exclusively comprises data provided by Codalab, with no additional external data used to assist training. To enable the reconstruction of full high-resolution images during inference, we employ a sliding window strategy in both the inference and testing phases. This approach, with a window size of 512 pixels and an overlap of 64 pixels, ensures that high-resolution inputs are processed seamlessly while maintaining compatibility with the model's patch-based design.

The training process is divided into two distinct phases, each with tailored hyperparameter configurations. During the first phase, we adopt the Charbonnier Loss due to its robustness against outliers commonly found in reflection regions. This phase employs a CosineAnnealingLR scheduler (T_max=10, eta_min=5e-5), with the learning rate initialized at 2e-4 and optimized using the AdamW optimizer across 2000 epochs. In the second phase, the model undergoes fine-tuning using L2 Loss to enhance texture fidelity further. Here, a ReduceLROnPlateau scheduler is applied, starting with a learning rate of 1e-5 and decaying it by a factor of 0.85 based on validation performance. Both training phases use a batch size of 10 and are initialized with weights pre-trained on OmniSR [18]. All experiments are conducted on three NVIDIA GeForce RTX 3090 GPUs using PyTorch 2.0.1.

1.5. i am a bug

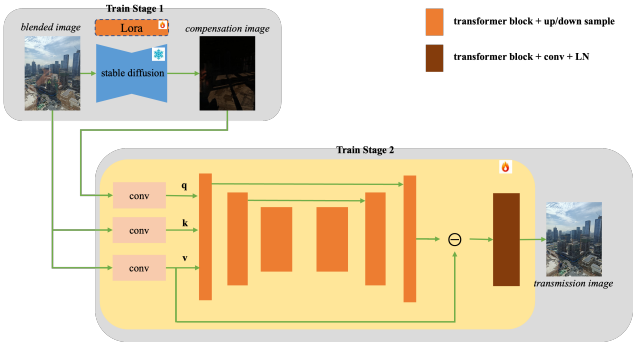


Figure 6. Network architecture of proposed solution.

Description: Most of the existing methods of SD model start from random noise to reconstruct the transmission image under the guidance of the given blended image in Single Image Reflection Removal (SIRR), Meanwhile, the random noise introduces uncertainty in the output, which is unfriendly to SIRR tasks. To address these issues, we propose a simple and effective way:

a. It only requires training a Lora [9] and does not require any modifications to the training process of the SD model, to predict the compensation image of the reflective area in the blended image.

b. A refine model is used to integrate the difference between the blended image and the transmission image, and the clean transmission image is obtained.

The network architecture is shown as Fig. 6.

Implementation: In train stage1, we use Flux.1-dev as the base SD model to train Lora(rank=16) to predict the reflection compensation image; In train stage2, we use a transformer model (pretrained on the ImageNet) to predict clean transmission image based on the blended image and compensation image in first stage. Specifically, we build a transformer model consisting of 6 layers of transformer blocks, each of which uses cross-attention optimization to compensate image in first stage. Then we calculate the difference between blended image and compensate image to obtain a clean projection image. At the same time, we use L1 loss, ssim loss and perceptual loss by VGG16. The learning rate is fixed as 2e-4 on a single NVIDIA A100(80G). In the testing phase, we first use the flux+lora model to predict the reflectance compensation image of all images in test set. Then, we input the predicted compensation images and blended images into the transformer model to obtain the final prediction results.

Only 100 images of data provided by the competition can be used to train a Lora that can predict accurate compensation image in train stage1; After completing the train stage1, This Lora can be used to generate the data needed for the train stage2. In the end, we only used the 800 training images provided by the competition to achieve a high evaluation index.

1.6. ImageLab

Description: The Lightweight Self-Calibrated Attention-Based Reflection Removal (LSCA-RR) network, as shown in Fig. 7, is a compact yet effective architecture designed to suppress reflections while preserving essential image details. The proposed network features a three-stage encoder with convolutional layers of 16, 16, and 32 filters, each enhanced by *Residual Dense Attention (RDA)* [13] blocks to effectively extract and refine spatial features while maintaining contextual consistency. At the network's front-end, the image undergoes a series of *Self-Calibrated Convolutions with Pixel Attention (SCPA)* [14, 29] blocks, which adaptively enhance pixel-level features and guide the network's focus toward non-reflective regions. The encoded features are progressively compressed via MaxPooling layers, and skip connections are preserved for later use in the decoding stages. The decoder path comprises two convo-

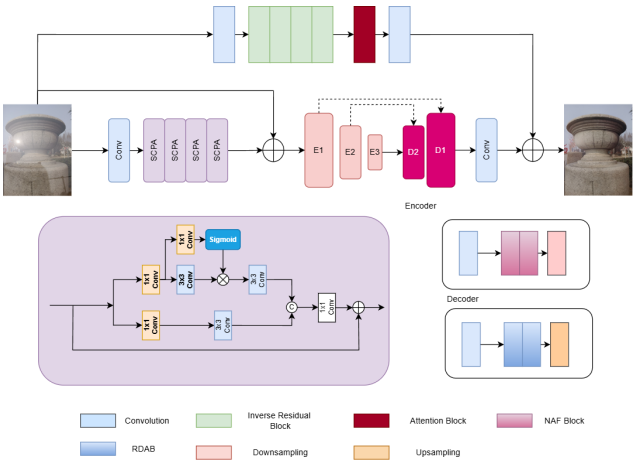


Figure 7. Architecture diagram.

lutional stages (with 16 filters each), also equipped with *Residual Dense Attention (RDA)* [12] blocks for deep hierarchical feature integration. Upsampling is performed using *Sub-Pixel Convolutional Blocks* to preserve texture and avoid checkerboard artifacts.

The network generates three intermediate outputs: O_X (shallow-refined output), E_1 (primary decoder output), and E_2 (auxiliary enhancement path output)[2]. These, along with the original input image I, are fused using an element-wise addition operation to produce the final output $\hat{I}$, formally expressed as:

$$\hat{I} = O_X + E_1 + E_2 + I \tag{2}$$

This multi-path fusion strategy allows the model to leverage both deep contextual and shallow spatial features, enabling high-quality reflection suppression in a computationally efficient manner.

The model is trained on the proposed loss function, which combines structural similarity, pixel-wise accuracy, and edge-preserving constraints to guide high-quality reflection removal. Let I be the ground truth image, and $\hat{I}$ be the predicted image. The total loss function $\mathcal{L}_{\text{total}}$ is defined as:

$$\mathcal{L}_{\text{total}} = 0.5 \cdot \Big(1 - \text{SSIM}(I, \hat{I})\Big) + \text{MAE}(I, \hat{I}) + \mathcal{L}_{\text{grad}} + \epsilon \tag{3}$$

where $\text{SSIM}(I, \hat{I})$ is the Structural Similarity Index, and the Mean Absolute Error is given by:

$$\text{MAE}(I, \hat{I}) = \frac{1}{N} \sum_{i=1}^{N} \left| I_i - \hat{I}_i \right| \tag{4}$$

The gradient loss $\mathcal{L}_{\text{grad}}$ is used to enforce edge fidelity and is defined as:

$$\mathcal{L}_{\text{grad}} = \frac{1}{N} \sum_{i=1}^{N} \Big(\left| \nabla_x \hat{I}_i - \nabla_x I_i \right| + \left| \nabla_y \hat{I}_i - \nabla_y I_i \right| \Big) \tag{5}$$

Here, ∇_x and ∇_y denote image gradients in the horizontal and vertical directions, respectively, and ϵ is a small constant added for numerical stability. This composite loss ensures that the output not only matches the ground truth in intensity and structure but also preserves edge sharpness, leading to perceptually superior reflection suppression.

Implementation: The proposed network was trained using the NVIDIA Tesla P100 with 16GB RAM and the TensorFlow Keras platform. The 400x400x3 patches are randomly extracted from the image. The model was trained using 4281 training patches and 755 validation patches. The augmentation was applied to training the model. We utilized the Adam optimizer, with a learning rate decreasing from 0.001 to 0.00001 over 250 epochs.

1.7. PSU TEAM

Description: We propose *OptimalDiff*, a novel image enhancement framework tailored for low-light image restoration, with a visual overview presented in Fig. 8. It reformulates the enhancement task as an optimal transport problem between degraded and clean image distributions using Schrödinger Bridge theory. The method integrates:

1. A hierarchical Swin Transformer-based encoder-decoder that effectively captures long-range dependencies and hierarchical context using window-based self-attention.
2. A Schrödinger Bridge diffusion module [3] that models the bidirectional transformation between noise and clean images through a forward degradation process and a conditional reverse denoising process.
3. A multi-scale refinement network that preserves global structure while enhancing fine details at full resolution.
4. A PatchGAN-based adversarial discriminator that encourages perceptually realistic outputs with natural textures.

The network is trained end-to-end using a composite loss function that includes diffusion loss, optimal transport loss (Sinkhorn divergence), multi-scale SSIM+L1 loss for perceptual quality, and adversarial loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{diff}} + \lambda_1 \mathcal{L}_{\text{OT}} + \lambda_2 \mathcal{L}_{\text{MS-SSIM+L1}} + \lambda_3 \mathcal{L}_{\text{adv}} \tag{6}$$

where:

- $\mathcal{L}_{\text{diff}}$: diffusion noise prediction loss,
- $\mathcal{L}_{\text{OT}}$: Sinkhorn divergence for optimal transport,
- $\mathcal{L}_{\text{MS-SSIM+L1}}$: perceptual and structural loss,
- $\mathcal{L}_{\text{adv}}$: PatchGAN-based adversarial loss.

We used the AdamW optimizer with a cosine annealing learning rate schedule, a batch size of 16, and trained the model for 300 epochs.

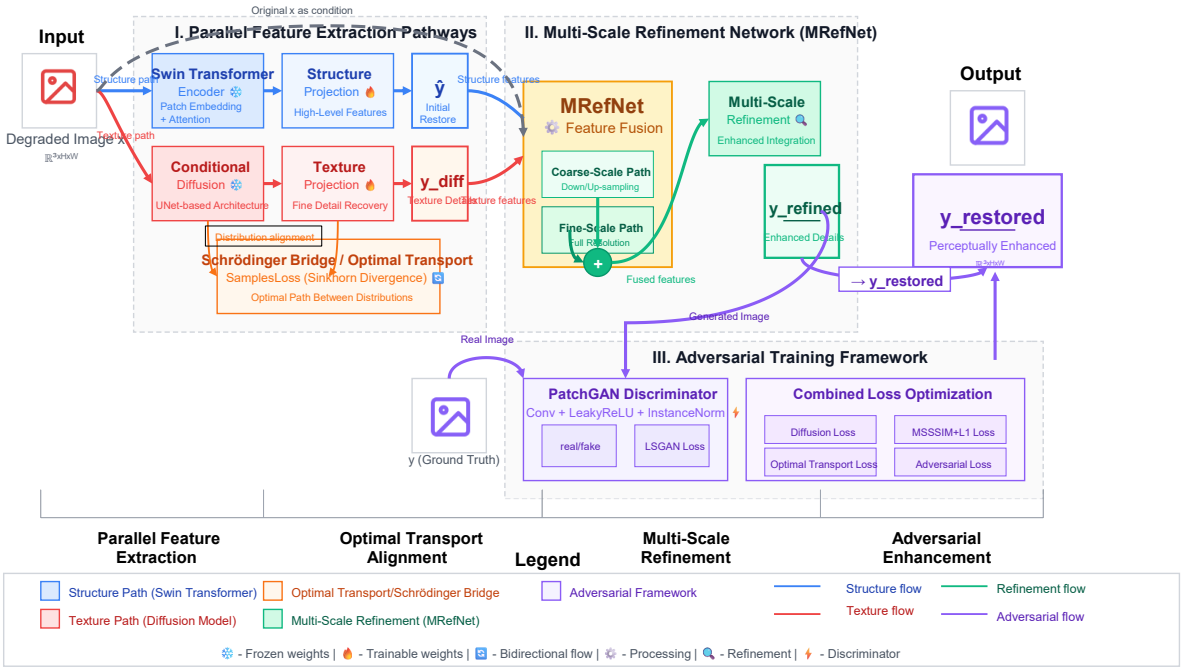


Figure 8. OptiMalDiff Architecture: The framework integrates optimal transport, diffusion models, and multi-scale refinement. It includes: (I) parallel structure and texture feature extraction; (II) MRefNet for coarse-to-fine refinement; and (III) adversarial training with combined losses. Schrödinger Bridge-based optimal transport aligns degraded and clean image distributions for enhanced restoration.

Implementation: The proposed method, OptimalDiff, was implemented in Python using the PyTorch deep learning framework (version 2.0+). All components including the Transformer backbone, diffusion module, refinement network, and discriminator were implemented from scratch. The model was trained using the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay of 2×10^{-4}. A cosine annealing learning rate scheduler was used with a starting learning rate of 2e-4. The batch size was set to 16, and the model was trained for 300 epochs. All experiments were conducted on a machine equipped with an NVIDIA RTX quadro 8000 GPU (48 GB VRAM).

1.8. RefLap

Description: The proposed reflection removal method combines a Laplacian decomposition branch (DWT-FFC [5, 23]) and an image reconstruction branch (UHDM [20]) in a hierarchical neural network. The DWT-FFC branch utilizes discrete wavelet transforms and Fast Fourier Convolutions in a U-Net architecture for effective multi-scale frequency-domain information extraction. Concurrently, the UHDM branch employs pixel unshuffling, residual dense blocks(RDB), and scale attention modules(SAM) to calibrate spatial features. Multi-scale residuals from the Laplacian branch assist the UHDM decoder with adaptive

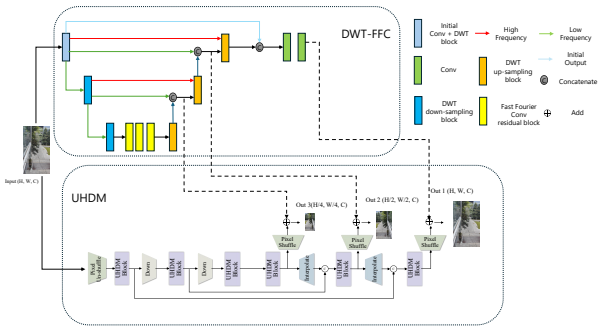


Figure 9. The overall architecture of Team RefLap. The DWT-FFC branch [23] aims to perceive the reflection artifacts and these reflection degradations are transferred into the UHDM branch [20] via multi-scale addition.

cross-scale addition fusion, facilitating detailed reflection-free image reconstruction. This two-domain strategy leverages complementary frequency and spatial features for the efficient elimination of reflections [4, 23]. The frame of the proposed shadow removal method is shown as Fig. 9.

Implementation: The model was implemented using Pytorch framework. The training leveraged the Adam optimizer with a base learning rate of 2e-4, and a cosine an-

nealing learning rate scheduler was used to gradually reduce the learning rate from 3e-4 to 1e-6 over two training cycles, promoting stable convergence. Training was conducted on a single GPU. The dataset used for training consisted of paired images for image reflection removal, with geometric augmentations applied to enhance generalization. Images were processed in batches of 12 with patch sizes of 352 × 352 pixels, totaling 150,000 iterations. Efficient optimization strategies included gradient clipping to maintain training stability and prefetching mechanisms for efficient data loading. The loss function combined four components: Charbonnier loss [10], structure loss [26] (weighted at 0.1) for pixel-level accuracy, perceptual loss using VGG feature layers [16] (weighted at 0.01), and multi-scale SSIM loss [6, 24, 25] (MSSSIM, weighted at 0.4) to enhance perceptual fidelity. Overall, the training environment was carefully designed for high efficiency, stability, and scalability.

1.9. X-L

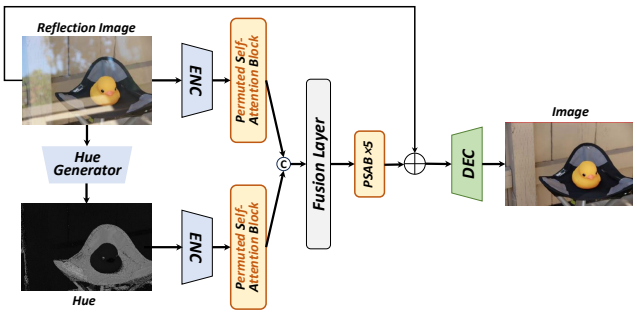


Figure 10. An overview of efficient Hue Guidance Network.

Description: We proposes an efficient Hue Guidance Network for Single Image Reflection Removal, which is shown in Fig. 10. Our method is inspired by two works: SRFormer [27] and HGNet [28]. In our framework, hue is generated, and the hue information aids reflection removal by highlighting reflection-distorted regions in the HSV color space. With the hue guidance, the hue features is extracted and used to locate and suppress reflection in the whole network effectively. In detail, first, the reflection image and the generated hue image serve as the two inputs. Both of these inputs are fed into encoding modules, respectively. And the two encoded features enter the permuted self-attention blocks (PSAB) to further achieve the compact feature representation, which is inspired by the efficiency of SRFormer [27]. Then, these features are fused by the fusion layer, integrating the compact information from different sources. The fused features then pass through the five PSAB blocks, and the multiple repetition structure is used for feature enhancement, optimizing the combined features further. Finally, the processed features are fed into the decoding module, which converts the features back into the clean image.

Implementation: We follow the same training strategy and settings as employed in SRFormer [27] and data self-ensemble is utilized during testing.

References

[1] Linwei Chen, Ying Fu, Lin Gu, Chenggang Yan, Tatsuya Harada, and Gao Huang. Frequency-aware feature fusion for dense image prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 4

[2] Marcos V Conde, Florin Vasluianu, Sabari Nathan, and Radu Timofte. Real-time under-display cameras image restoration and hdr on mobile devices. In *European Conference on Computer Vision*, pages 747–762. Springer, 2022. 6

[3] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021. 6

[4] Wei Dong, Han Zhou, Yuqiong Tian, Jingke Sun, Xiaohong Liu, Guangtao Zhai, and Jun Chen. Shadowrefiner: Towards mask-free shadow removal via fast fourier transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6208–6217, 2024. 7

[5] Wei Dong, Han Zhou, Ruiyi Wang, Xiaohong Liu, Guangtao Zhai, and Jun Chen. Dehazedct: Towards effective non-homogeneous dehazing via deformable convolutional transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6405–6414, 2024. 7

[6] Wei Dong, Han Zhou, Yulun Zhang, Xiaohong Liu, and Jun Chen. Ecmamba: Consolidating selective state space model with retinex guidance for efficient multiple exposure correction. *Advances in Neural Information Processing Systems*, 37:53438–53457, 2024. 8

[7] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 3

[8] Qiming Hu, Hainuo Wang, and Xiaojie Guo. Single image reflection separation via dual-stream interactive transformers. *Advances in Neural Information Processing Systems*, 37:55228–55248, 2024. 3

[9] Lianghua Huang, Wei Wang, Zhi-Fan Wu, Yupeng Shi, Huanzhang Dou, Chen Liang, Yutong Feng, Yu Liu, and Jingren Zhou. In-context lora for diffusion transformers. *arXiv preprint arXiv:2410.23775*, 2024. 5

[10] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Fast and accurate image super-resolution with deep laplacian pyramid networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2599–2613, 2018. 8

[11] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E Hopcroft. Single image reflection removal through cascaded refinement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3565–3574, 2020. 3

[12] Xiaoning Liu, Zongwei Wu, Ao Li, Florin-Alexandru Vasluianu, Yulun Zhang, Shuhang Gu, Le Zhang, Ce Zhu, Radu Timofte, Zhi Jin, et al. Ntire 2024 challenge on low light image enhancement: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6571–6594, 2024. 6

[13] D Sabari Nathan, K Uma, D Synthiya Vinothini, B Sathya Bama, and SM Roomi. Light weight residual dense attention net for spectral reconstruction from rgb images. *arXiv preprint arXiv:2004.06930*, 2020. 5

[14] Sabari Nathan and Priya Kansal. End-to-end depth-guided relighting using lightweight deep learning-based method. *Journal of Imaging*, 9(9):175, 2023. 5

[15] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 4

[16] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 8

[17] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Benchmarking single-image reflection removal algorithms. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3922–3930, 2017. 2

[18] Jiamin Xu, Zelong Li, Yuxin Zheng, Chenyu Huang, Renshu Gu, Weiwei Xu, and Gang Xu. Omnisr: Shadow removal under direct and indirect lighting. *arXiv preprint arXiv:2410.01719*, 2024. 4, 5

[19] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024. 4

[20] Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Jiajun Shen, Jia Li, and Xiaojuan Qi. Towards efficient and scale-robust ultra-high-definition image demoiréing. *arXiv preprint arXiv:2207.09935*, 2022. 7

[21] Xuaner Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4786–4794, 2018. 2, 3

[22] Hao Zhao, Yiming Zhu, Jiuqing Dong, Kui Jiang, Junjun Jiang, and Yang Chen. Reversible decoupling network for single image reflection removal. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–10, 2025. 1, 3, 4

[23] Han Zhou, Wei Dong, Yangyi Liu, and Jun Chen. Breaking through the haze: An advanced non-homogeneous dehazing method based on fast fourier convolution and convnext. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1895–1904, 2023. 7

[24] Han Zhou, Wei Dong, Xiaohong Liu, Shuaicheng Liu, Xiongkuo Min, Guangtao Zhai, and Jun Chen. Glare: Low light image enhancement via generative latent feature based codebook retrieval. In *European Conference on Computer Vision*, pages 36–54. Springer, 2024. 8

[25] Han Zhou, Wei Dong, Xiaohong Liu, Yulun Zhang, Guangtao Zhai, and Jun Chen. Low-light image enhancement via generative perceptual priors. *arXiv preprint arXiv:2412.20916*, 2024. 8

[26] Han Zhou, Wei Dong, and Jun Chen. Lita-gs: Illumination-agnostic novel view synthesis via reference-free 3d gaussian splatting and physical priors. *arXiv preprint arXiv:2504.00219*, 2025. 8

[27] Yupeng Zhou, Zhen Li, Chun-Le Guo, Song Bai, Ming-Ming Cheng, and Qibin Hou. Srformer: Permuted self-attention for single image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12780–12791, 2023. 8

[28] Yurui Zhu, Xueyang Fu, Zheyu Zhang, Aiping Liu, Zhiwei Xiong, and Zheng-Jun Zha. Hue guidance network for single image reflection removal. *IEEE Transactions on Neural Networks and Learning Systems*, 35(10):13701–13712, 2024. 8

[29] Wenbin Zou, Tian Ye, Weixin Zheng, Yunchen Zhang, Liang Chen, and Yi Wu. Self-calibrated efficient transformer for lightweight super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 930–939, 2022. 5