Online ISSN: 3104-9338
Print ISSN: 3104-932X

Adaptive Convolution and Feature Aggregation for Single-Image Shadow Removal

Shangan Zhou

College of Physics and Electronic Information Engineering, Zhejiang Normal University, China

Received: May 25, 2026

Revised: May 26, 2026

Accepted: May 27, 2026

Published online: May 30, 2026

To appear in: *International Journal of Advanced AI Applications,* Vol. 2, No. 6 (June 2026)

* Corresponding Author: Shangan Zhou (3217432128@qq.com)

Abstract. Shadows degrade image quality and hinder computer vision tasks. Existing deep learning methods suffer from fine-detail loss due to downsampling and indiscriminate processing of shadow/non-shadow regions, causing unwanted alterations. To address these issues, this thesis proposes two complementary approaches. First, the Adaptive Alignment and Illumination-Aware Convolution (AAIC) framework uses a Feature Alignment Module (FAM) to recover lost details and an Illumination-Aware Weighting Module (IWM) for spatially varying convolution, achieving high quantitative performance on ISTD+ and SRD datasets. Second, the Feature Aggregation Shadow Removal Network (FASR-Net) reconstructs shadow-free images via learned fusion rather than end-to-end transformation, employing a Detail Feature Extractor (DFE), Dual-Branch Aggregation Weight Generator (DAWG), and Feature Enhancement (FE) modules to preserve non-shadow regions and ensure global consistency. Both methods are extensively evaluated using RMSE and SSIM, outperforming state-of-the-art techniques. Ablation studies validate each component, and the methods improve foreground segmentation in videos and generalize to remote sensing images without fine-tuning.

Keywords: *Shadow Removal; Adaptive Convolution; Feature Aggregation; Deep Learning;Illumination Consistency*

1. Introduction

1.1. Background and Significance

The Shadows are a fundamental visual phenomenon that occurs when an opaque object partially or completely blocks the propagation of light. The formation of a shadow requires only

three simple conditions: a light source (such as the sun, a lamp, or any emitting surface), an occluding object (which can be any solid entity), and a projection surface (the area where the shadow is cast). When these conditions are met, the projection surface receives significantly less illumination than its surrounding regions, making it appear darker to the human eye and to imaging sensors. Because these conditions are easily satisfied in both natural and artificial environments, shadows appear in nearly all images captured in automated scenarios—from everyday smartphone photographs to high-resolution satellite imagery.

The presence of shadows, however, is far from benign. While shadows can provide valuable cues for certain tasks—such as inferring 3D geometry [2], estimating light source direction [3], or understanding object shapes—they often severely degrade the performance of computer vision systems. The negative impacts of shadows are widespread and well-documented:

- Image segmentation: Shadow boundaries frequently exhibit strong intensity gradients that are easily confused with true object edges. As a result, many segmentation algorithms mistakenly label shadow regions as part of foreground objects, leading to oversegmentation or incorrect boundary delineation [1]. In semantic segmentation, shadows can cause entire regions to be misclassified (e.g., a shadow on a road might be labeled as a vehicle or a pothole).
- Image segmentation: Shadow boundaries frequently exhibit strong intensity gradients that are easily confused with true object edges. As a result, many segmentation algorithms mistakenly label shadow regions as part of foreground objects, leading to oversegmentation or incorrect boundary delineation [1]. In semantic segmentation, shadows can cause entire regions to be misclassified (e.g., a shadow on a road might be labeled as a vehicle or a pothole).
- Remote sensing and aerial image analysis: In satellite or drone imagery, shadows cast by buildings, trees, and terrain features obscure the underlying ground information [5]. This reduces the accuracy of land cover classification, change detection, and building footprint extraction. Shadows can also introduce false positives in disaster assessment (e.g., a shadow may be mistaken for a flooded area) and reduce the effective resolution of useful data.
- Autonomous driving: Shadows on the road can create false lane markings, obscure traffic signs, or cause pedestrians to blend into dark backgrounds [6]. In extreme cases, a sharp shadow edge may be misinterpreted as a curb or obstacle, triggering unnecessary braking or swerving. Conversely, soft shadows may reduce the contrast of lane lines,

making them harder to detect.

- Image enhancement and computational photography: Shadows reduce the aesthetic quality of images, making them appear dull or unbalanced. Professional photographers and video editors spend considerable time manually removing or softening shadows to improve visual appeal. In High-Dynamic-Range (HDR) imaging, shadows can cause artifacts during fusion of multiple exposures.
- Medical imaging: Even in specialized domains, shadows can be problematic. For instance, in endoscopic images, shadows cast by instruments or tissue folds can obscure lesions. In retinal fundus photography, shadows from optic disc margins may be misinterpreted as pathologies.

The widespread impact of shadows has motivated decades of research into shadow detection and removal. Despite these efforts, manual shadow removal using tools like Adobe Photoshop remains the standard practice in many professional settings. Such manual intervention is time-consuming, subjective, and impractical for large-scale or real-time applications. A single image may take minutes to hours to clean up, and video sequences are virtually impossible to process frame-by-frame manually. Therefore, automated shadow removal is not merely an academic curiosity—it is a practical necessity for deploying computer vision systems in uncontrolled real-world environments.

Automated shadow removal is challenging due to the wide variation in shadow shape, size, intensity, and surface reflectance. Hard shadows (umbra) have sharp, well-defined edges, while soft shadows (penumbra) exhibit gradual intensity transitions. Shadows can be cast by multiple sources simultaneously, creating complex overlapping patterns. The color of the light source (e.g., sunlight vs. indoor LED) affects the chromaticity of the shadow region. Moreover, the texture and reflectance of the underlying surface interact with shadows in non-linear ways. As a result, a method that works well on one type of shadow may fail on another.

Figure 1 illustrates a concrete example from foreground segmentation. The leftmost image shows an input video frame (from the Bungalows sequence). The middle image is the ground-truth segmentation mask. The rightmost image is the result produced by an existing segmentation method. It is evident that the shadow cast by the vehicle is segmented together with the vehicle itself, creating an incorrect mask that includes both the true object and its shadow. If this shadow had been removed before segmentation, the result would likely be much cleaner. This example underscores the practical importance of effective shadow removal as a pre-processing step.

Figure 1. From left to right: input video sequence frame, ground-truth segmentation, and segmentation result from an existing method. The shadow is erroneously segmented along with the vehicle.

In recent years, deep learning has emerged as a powerful tool for image restoration tasks, including shadow removal. However, as detailed in the following section, existing deep learning methods still face two fundamental limitations: (1) loss of fine details due to encoder-decoder downsampling, and (2) uniform processing of all image regions, which fails to preserve non-shadow areas and lacks global illumination consistency. This thesis aims to address these limitations through two novel approaches that combine adaptive convolution, feature alignment, and feature aggregation.

1.2. Current Research Landscape and Problem Analysis

1.2.1. Traditional Shadow Removal Methods

Traditional methods rely on hand-crafted features and physical models. Global methods [7-9] use illumination-invariant images or color space transformations. For instance, Finlayson et al. [7] first proposed a method based on gray-scale illumination-invariant images, which was later extended to full-color images [8]. Region-based methods [10-13] perform illumination transfer between matched patches. Some approaches [14,15] use machine learning on hand-crafted features. However, these methods often make strong assumptions (e.g., uniform illumination) and may require user interaction, limiting their generalization to complex real-world scenes.

1.2.2. Deep Learning-Based Shadow Removal Methods

Deep learning has revolutionized shadow removal. Architectures such as U-Net [16], GANs [17], Cycle-GAN [18], and Vision Transformers [19] have been widely adopted. Existing deep learning methods can be categorized as:

- Physical model-based [20-22]: Use networks to predict parameters of simplified physical models. Interpretable but inherit model limitations.
- Single-stage end-to-end [23-31]: Directly map shadow images to shadow-free images. Most common but suffer from detail loss and uniform region processing.

- Multi-stage [32-38]: Decompose into detection, coarse removal, and refinement. More accurate but more complex.
- Multi-branch [39-44]: Extract different types of features in parallel. Flexible but require careful design.

Despite significant progress, two critical issues remain:

(1) Detail loss in encoder-decoder architectures: Repeated downsampling discards fine details such as edges, textures, and small structures. Even with skip connections, the decoder cannot fully recover what was lost. This leads to blurred outputs, especially in shadow regions where the original texture is already low-contrast.

(2) Uniform treatment of shadow and non-shadow regions: Most methods apply the same transformation globally. This wastes model capacity on non-shadow areas, which should ideally remain unchanged, and provides insufficient focus on shadow regions that require significant modification. Consequently, non-shadow areas may be inadvertently altered (e.g., color shifts), and shadow regions may not be fully restored. A few methods attempt to use shadow masks to guide processing [25], but they typically rely on binary masks and do not leverage continuous spatial variation.

Recent works have attempted to address these issues via attention mechanisms [45,46], transformers [47,48], and diffusion models [49], but each introduces its own trade-offs. Attention mechanisms add computational overhead. Transformers require large datasets and are slow to train. Diffusion models are extremely slow at inference. Therefore, there remains a need for efficient and effective solutions that balance performance, speed, and model size.

2. Background and Preliminaries

2.1. Deep Learning Fundamentals

2.1.1. Convolutional Neural Networks for Image Processing

Convolutional Neural Networks (CNNs) have become the backbone of modern image processing. A standard CNN consists of alternating convolutional and pooling layers, followed by fully connected layers for classification. For pixel-wise prediction tasks such as shadow removal, fully convolutional architectures are preferred because they preserve spatial dimensions.

- FCN [50]: The Fully Convolutional Network replaces the fully connected layers with convolutional layers, enabling dense prediction. It uses transposed convolutions (also called deconvolutions) to upsample the feature maps back to the input resolution.

However, the upsampling is coarse and often leads to blurred boundaries.

- U-Net [16]: To address the coarse upsampling issue, U-Net introduces a symmetric encoder-decoder structure with skip connections. The encoder progressively reduces spatial resolution while increasing channel depth, capturing global context. The decoder upsamples the features and concatenates them with the corresponding encoder features via skip connections, which helps recover fine spatial details. This architecture has become the de facto standard for many image restoration tasks, including shadow removal.
- SegNet [51]: SegNet also uses an encoder-decoder but employs max-pooling indices to transfer pooling switches from the encoder to the decoder. This reduces the number of trainable parameters and memory usage, but the performance is generally lower than U-Net for dense prediction tasks.
- RefineNet [54]: RefineNet uses multi-path refinement blocks to combine features from different levels of the encoder. It also introduces chained residual pooling to capture context from large regions without using large pooling windows. Although powerful, RefineNet is computationally expensive.

2.1.2. Dilated Convolutions and Atrous Spatial Pyramid Pooling

A major limitation of standard convolutions is that increasing the receptive field requires either larger kernel sizes (which increase parameters) or more layers (which increase depth). Dilated convolutions [52] solve this problem by inserting zeros between kernel elements, effectively increasing the receptive field without adding parameters.

For a 1D signal, a dilated convolution with dilation rate r can be written as:

$$(F *_r k)(p) = \sum_s F(p + r \cdot s)k(s)$$

In 2D, the same principle applies. By using multiple dilation rates (e.g., 1, 2, 4, 8), the network can capture multi-scale contextual information.

Atrous Spatial Pyramid Pooling (ASPP) [53] applies parallel dilated convolutions with different dilation rates on the same feature map, then concatenates the results. This allows the network to simultaneously capture features at multiple scales. ASPP has been widely used in semantic segmentation and has inspired our Residual Dilation Block in later sections.

2.1.3. Attention Mechanisms

Attention mechanisms allow neural networks to focus on the most informative parts of the

input. They have been successfully applied to many vision tasks, including image restoration.

- Squeeze-and-Excitation (SE) Block [55]: The SE block recalibrates channel-wise feature responses by explicitly modeling interdependencies between channels. It first squeezes the spatial dimensions using global average pooling, producing a channel descriptor. Then it excites the descriptor through two fully connected layers (with ReLU and Sigmoid) to generate channel attention weights. Finally, it scales the original feature map channel-wise. The SE block is lightweight and can be inserted into any existing architecture.
- Non-local Networks [56]: Non-local operations capture long-range dependencies by computing pairwise similarities between all positions. For an input feature map x , the non-local output at position i is:

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f\left(x_i, x_j\right) g\left(x_j\right)$$

Where f computes the affinity between positions i and j,$g$is a transformation function, and $C(x)$ is a normalization factor. Non-local blocks are powerful but computationally expensive, as they require $O(H^2W^2)$ operations.

- Dual Attention Network (DANet) [57]: DANet combines spatial attention and channel attention in parallel. The spatial attention branch captures where to focus, while the channel attention branch captures what to focus on. The outputs of the two branches are summed to produce the final attended features.
- Global Context Network (GCNet) [58]: GCNet simplifies the non-local block by observing that the attention weights for different query positions are nearly identical. It replaces the pairwise computation with a simplified global context pooling and a SE-like bottleneck, achieving comparable performance with much lower computational cost.
- Criss-Cross Attention (CCNet) [59]: CCNet reduces the complexity of non-local attention from $O(H^2W^2)$ to $O(HW(H+W))$ by only attending to positions in the same row and column as the query point. Repeating this operation twice approximates full non-local attention.

2.1.4. Adaptive (Dynamic) Convolution

Standard convolutions use the same kernel weights for all spatial locations and all input samples. Adaptive (dynamic) convolution breaks this invariance by conditioning the kernel on the input content.

- Deformable Convolution [60,61]: Deformable convolution adds 2D offsets to the regular grid sampling locations. The offsets are learned from the input feature map via a separate convolutional layer. For each output position p_0 , the output is：

$$y(p_0) = \sum_{p_n \in \mathcal{R}} w(p_n) \cdot x(p_0 + p_n + \Delta p_n)$$

Where R is the regular grid, $w(p_n)$ are the kernel weights, and Δp_n are the learned offsets. Deformable convolution allows the receptive field to adapt to the geometric structure of the object, which is particularly useful for handling irregular shadow boundaries.

- Pixel-Adaptive Convolution [62]: Pixel-adaptive convolution uses a separate guidance image to generate spatially-varying kernels. The kernel at position i is a function of the guidance features in a local window. This allows the convolution to respect edges in the guidance image.
- Dynamic Region-Aware Convolution (DRConv) [63]: DRConv first segments the feature map into regions using a learnable assignment module, then applies different convolution kernels to different regions. This is similar to our IWM but uses hard region assignments, whereas IWM uses soft, continuous weights.

2.2. Shadow Removal Datasets

2.2.1. SRD Dataset

The Shadow Removal Dataset (SRD) was introduced by Qu et al. [39]. It contains 3600 pairs of shadow and shadow-free images, with 2500 for training and 380 for testing. The images were captured using a tripod-mounted Canon 5D camera with fixed settings to minimize illumination differences. The dataset is diverse in four aspects:

- Illumination: Includes both soft and hard shadows, captured at different times of day (dawn, morning, noon, afternoon, dusk) and under different weather conditions (sunny, cloudy).
- Scene: Covers a wide range of scenes, including campuses, streets, mountains, beaches, and building interiors.
- Reflectance: Shadows are cast on various surfaces with different material properties, such as asphalt, grass, concrete, and wood.
- Shape: The occluding objects have diverse shapes, including umbrellas, planks, people, trees, and vehicles.

SRD does not provide shadow masks. Following common practice, we use the masks

generated by DHAN [23], which are of high quality.

Figure 2 shows example images from the SRD dataset. The top row shows shadow images, the middle row shows shadow masks, and the bottom row shows the corresponding shadow-free ground truth.

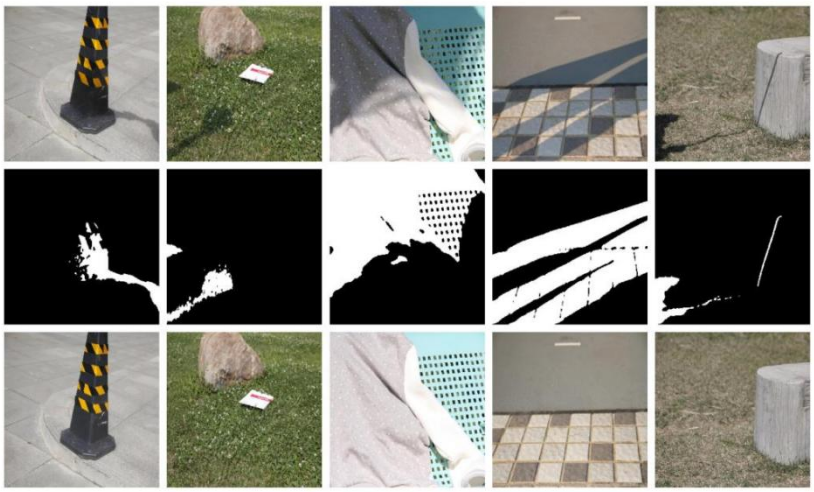

Figure 2. Sample images from the SRD dataset. Top: shadow images; middle: shadow mask; bottom: shadow-free ground truth.

2.2.2. SRD Dataset

The ISTD dataset was introduced by [32]. It is the first triplet shadow dataset, containing shadow images, shadow masks, and shadow-free images. There are 1,680 triplets in total, with 1,330 for training and 540 for testing. The dataset includes125 different scenes.

However, ISTD has a notable limitation: even the non-shadow regions in the shadow-free images differ from those in the shadow images due to ambient light changes during capture. The overall RMSE between the shadow image and the shadow-free image in the non-shadow regions is 6.83 [20].

To address this issue, a linear regression correction was applied, in which the authors computed a linear mapping for each channel from the non-shadow pixels of the shadow image to the corresponding pixels in the shadow-free image [20]. The mapping parameters were subsequently utilized to transform the entire shadow-free image, resulting in the creation of ISTD+. Following this correction, the overall RMSE for the test set drops to 2.6. It is important to note that all experiments conducted in this thesis utilize ISTD+ as the foundational dataset.

Figure 3 shows example triplets from the ISTD dataset. The top row is the shadow image, the middle row is the shadow mask, and the bottom row is the corresponding shadow-free ground truth. In this thesis, we use the ISTD+ version [20], which corrects illumination

inconsistencies via linear regression.

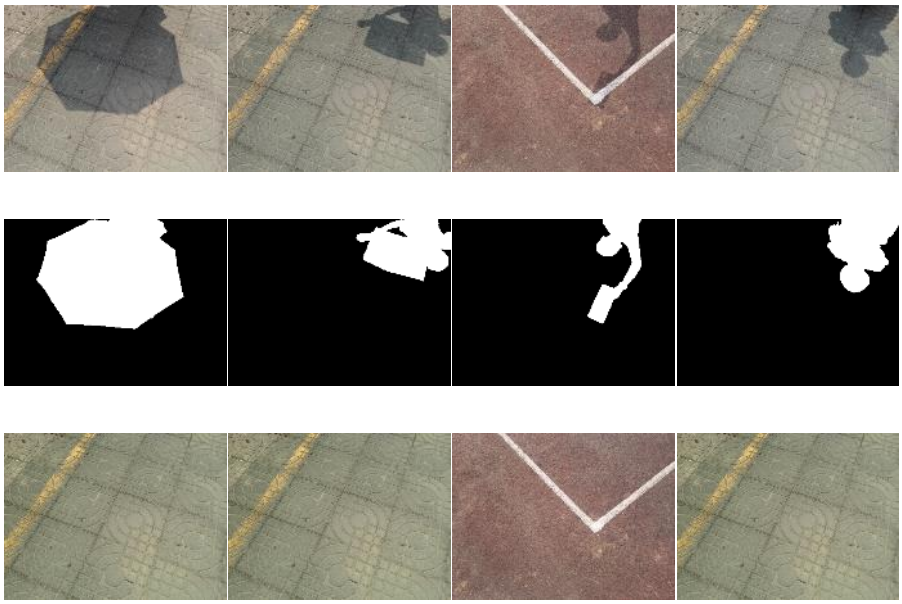

Figure 3. Example triplets from the ISTD dataset [32]: shadow image, shadow mask, and shadow-free ground truth.

2.3. Evaluation Metrics

2.3.1. Structural Similarity Index (SSIM)

SSIM [64], or Structural Similarity Index Measure, is a sophisticated perceptual metric that is widely used to quantify the degradation of image quality. Unlike traditional pixel-wise error metrics such as Mean Squared Error (MSE), which merely evaluate differences at the pixel level, SSIM takes into account critical factors like luminance, contrast, and structural information within the image. This comprehensive approach enables SSIM to more accurately reflect human visual perception, ultimately providing a more meaningful assessment of image fidelity and quality. Thus, it serves as a valuable tool for applications in image processing and analysis.

Given two images x and y, SSIM is defined as:

$$SSIM(x,y) = l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma}$$

Where:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

Here, μ_x μ_y are the means, σ_x, σ_y are the standard deviations, σ_{xy} is the cross-correlation, and C_1, C_2, C_3 are small constants to avoid division by zero. Typically, $\alpha = \beta = \gamma = 1$ and $C_3 = \frac{C_2}{2}$.

simplifying to:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

SSIM ranges from 0 to 1, with 1 indicating identical images. It is widely used because it correlates well with human visual perception.

2.3.2. Root Mean Square Error (RMSE)

RMSE is a pixel-wise error metric that is sensitive to large errors. It is computed in the LAB color space because LAB separates luminance from chrominance, making it more perceptually relevant than RGB.

Given a predicted image I_p and a ground-truth image I_{gt} , each with N pixels, the RMSE is:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \left\| I_p(i) - I_{gt}(i) \right\|_2^2}$$

Lower RMSE indicates better reconstruction. In shadow removal, we often report RMSE separately for the whole image, the shadow region (where the mask is 1), and the non-shadow region (where the mask is 0). This allows us to assess whether a method correctly leaves non-shadow areas unchanged.

3. Shadow Removal via Adaptive Alignment and Illumination Aware Convolution

3.1. Introduction

As discussed in the introduction, existing encoder-decoder based shadow removal methods suffer from two major problems: (1) loss of fine details during downsampling, and (2) uniform processing of all image regions. To address these, we propose the Adaptive Alignment and Illumination-Aware Convolution (AAIC) framework. The core idea is to make both the spatial sampling locations and the convolution kernel weights adaptive to the input image content and to the shadow-specific properties.

AAIC consists of two novel modules:

- Feature Alignment Module (FAM): Recovers lost details by using shallow encoder features (rich in spatial information) to align the deep decoder features via deformable sampling.
- Illumination-Aware Weighting Module (IWM): Generates spatially-variant convolution kernels based on the global illumination context, enabling the network to treat shadow

and non-shadow regions differently.

Unlike prior deformable convolution methods [60,61] that learn offsets from the same feature map, FAM uses features from skip connections, providing more accurate alignment cues. Unlike dynamic region-aware convolution [63] that uses hard region assignments, IWM produces soft, continuous weights, which is more appropriate for shadow boundaries were illumination changes gradually.

3.2. Proposed Method

3.2.1. Overall Architecture

We adopt a two-stage exposure fusion framework similar to AEF [21]. The pipeline is illustrated in Figure 4. It comprises three main components:

- Exposure parameter prediction network: Given the shadow image and shadow mask , a Conformer-based [63] network predicts a set of exposure parameters. These parameters are used to generate five exposure-adjusted versions of the input image (exposure sequence). The Conformer architecture combines a CNN branch for local features and a Transformer branch for global features, making it well-suited for this task.
- Coarse shadow removal network: The shadow image, shadow mask, and exposure sequence are concatenated along the channel dimension and fed into the first AAIC-enhanced U-Net. This network produces a coarse shadow-free image.
- Refinement network: The coarse result, original shadow image, and shadow mask are input to the second AAIC-enhanced U-Net, which refines the output, particularly around shadow boundaries.

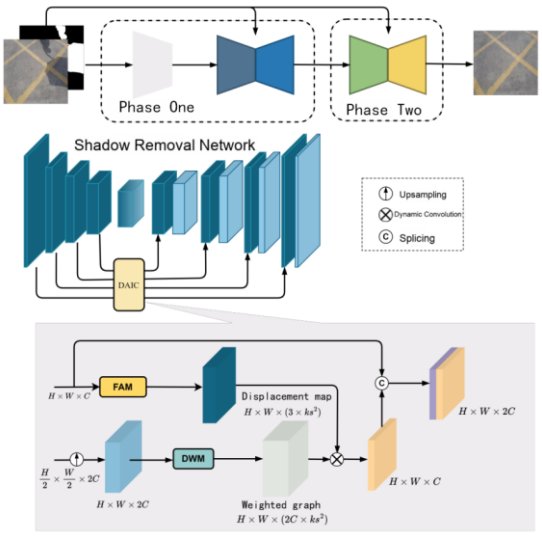

Figure 4. Overall architecture of the proposed AAIC framework.

Both U-Nets have the same structure: an encoder with 4 downsampling blocks (each block: two 3×3 convolutions followed by a 2×2 max-pooling), a bottleneck, and a decoder with 4 upsampling blocks (each block: a transposed convolution followed by two 3×3 convolutions). Skip connections link each encoder block to the corresponding decoder block. In standard U-Net, the convolutions are regular. In our AAIC-enhanced U-Net, we replace the first convolutional layer in each decoder block with an AAIC module that includes both FAM and IWM.

3.2.2. Overall Architecture

FAM is designed to recover fine details lost during downsampling. The intuition is that the shallow encoder features (from skip connections) contain rich spatial detail, while the deep decoder features contain high-level semantic information but are spatially coarse. FAM uses the shallow features to "guide" the deep features, aligning them to better preserve details.

Let $F_{shallow} \in \mathbb{R}^{H\times W\times C}$ the shallow feature map from the corresponding encoder layer, and let $F_{deep} \in \mathbb{R}^{H\times W\times C}$ be the upsampled decoder feature map before the convolution. FAM operates as follows:

(1) Channel attention on shallow features:$F_{shallow}$passes through a channel attention block: global average pooling, two fully connected layers (with ReLU and Sigmoid), and element-wise multiplication. This produces $F_{enhanced}$.

(2) Offset map prediction: A 3×3 convolutional layer with $3 \cdot k_s^2$ output channels are applied to $F_{enhanced}$ to produce an offset map $o_{map} \in \mathbb{R}^{H\times W\times (3\cdot k_s^2)}$,where k_s is the kernel size (typically 3). The factor of 3 comes from the x-offset, y-offset, and a modulation scalar (optional, but we use it).

(3) Warping: The upsampled decoder feature map $F_{upsample}$ is warped according to O_{map}, For each output position p, the kernel sampling grid is shifted by the offsets predicted for that position. This is implemented via grid sampling with bilinear interpolation to ensure differentiability.

The output $F_{aligned}$ is then passed to the subsequent convolution (which may be a regular convolution or the IWM). The effect of FAM is that the network can sample pixels from more relevant locations, effectively "de-blurring" the feature map and restoring fine structures such as hair, grass, and texture edges.

Mathematical formulation: For a convolutional kernel with $k_s \times k_s$ sampling points, let the regular grid offsets be $\{(dx_1, dy_1), \dots, (dx_K, dy_K)\}\ where\ K = k_s^2$. The offset map predicts

for each output position p and each kernel point k a 2D offset$(\Delta x_{p,k}, \Delta y_{p,k})$and a modulation weight $m_{p,k}$.The warped feature value at position p for the k-th kernel point is:

$$F_{warped}(p,k) = \sum_{q \in \mathcal{N}\left(p+(dx_k,dy_k)+(\Delta x_{p,k},\Delta y_{p,k})\right)} w_{bilinear}(q) \cdot F_{upsample}(q)$$

Where N is the 2×2 neighborhood for bilinear interpolation. The aligned feature map is then obtained by convolving these warped samples with the kernel weights. The architecture of FAM is depicted in Figure 5.

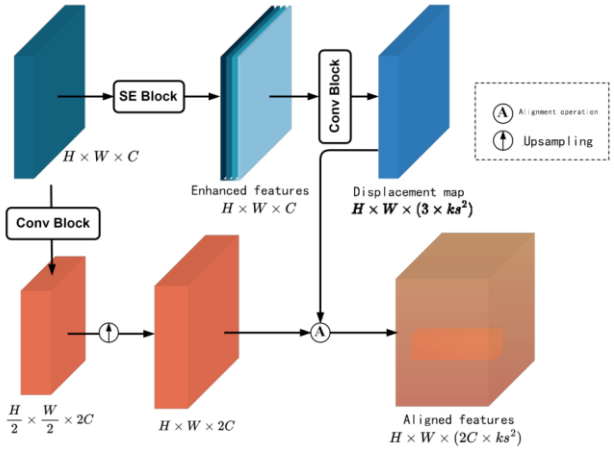

Figure 5. Feature Alignment Module (FAM).

3.2.3. Illumination-Aware Weighting Module (IWM)

While FAM addresses spatial misalignment, IWM addresses the need for region-specific processing. In a shadow image, the transformation required to convert shadow to non-shadow is very different from the identity mapping required for non-shadow regions. IWM allows the network to use different convolution kernels for different spatial locations, effectively allocating more capacity to shadow regions.

IWM predicts a separate kernel for each output position (and, implicitly, for each input sample). The module takes as input the feature map $F \in \mathbb{R}^{H \times W \times C}$ It then:

(1) Applies a 3×3 convolution to reduce the number of channels (e.g., to C/4) and introduce non-linearity.

(2) Applies layer normalization instead of batch normalization. Layer normalization computes mean and variance over the channel dimension for each sample independently, which forces the network to focus on intra-sample variations—exactly what we need for distinguishing shadow vs. non-shadow within the same image.

(3) Applies ReLU activation.

(4) Applies another 3×3 convolution to produce a weight map $W_{map} \in \mathrm{R}^{\mathrm{H\times W\times}\left(\mathrm{C\cdot k_s^2}\right)}$. This is the dynamic kernel bank.

(5) Reshapes : W_{map} such that at each position (i, j), we have a kernel $W_{i,j} \in \mathbb{R}^{C\times k_s\times k_s}$.

The convolution operation with IWM is then:

$$F_{out}(i,j) = \sum_{c=1}^{C} \sum_{dx,dy} W_{i,j}(c, dx, dy) \cdot F_{in}(i + dx, j + dy, c)$$

Figure 6 illustrates the IWM structure.

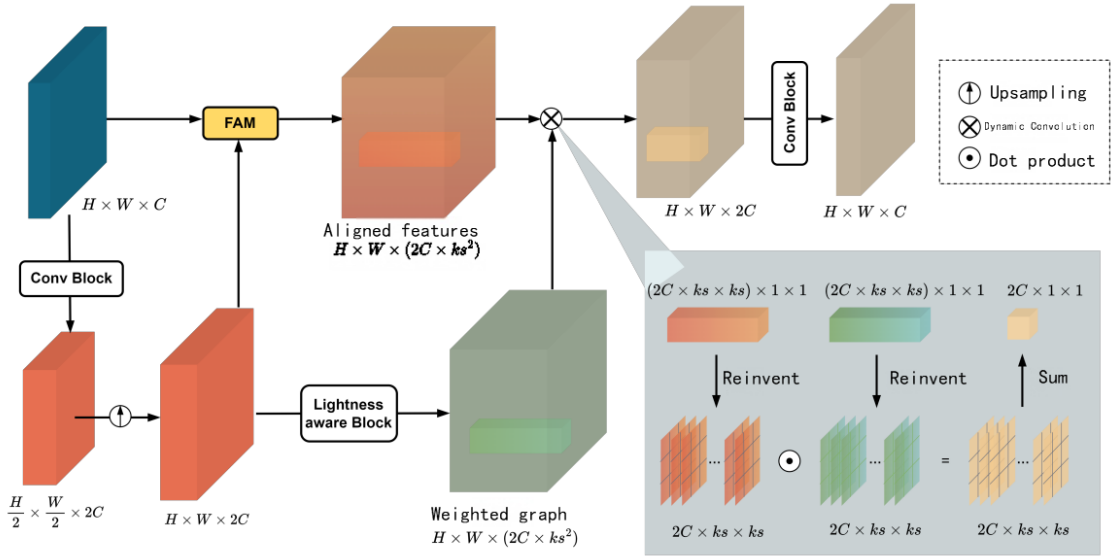

Figure 6. Illumination-Aware Weighting Module (IWM).

In practice, to keep memory and computation manageable, we use a grouped convolution formulation: we first split the input feature channels into groups, predict separate kernels per group, and then combine.

Compared to standard convolution, IWM has significantly more parameters (by a factor of H·W in the kernel bank). However, because the kernel prediction network is very small (two 3×3 convolutions), the total parameter increase is modest. In our implementation, adding IWM increases the total model size by about 15%.

$$\text{Standard convolution: } WIN_{output}^{i,j} = WIN_{input}^{i,j} \times WT$$

$$\text{IWM: } WIN_{output}^{i,j} = WIN_{input}^{i,j} \times WT_{i,j}$$

3.2.4. Loss Functions

The exposure parameter prediction network is trained with MSE loss. Let the predicted exposure parameters be $\hat{p}$ and the target parameters be p (obtained via least squares fitting [64]

from the shadow image and the shadow-free image's non-shadow regions). The loss is:

$$\mathcal{L}_{param} = \|\hat{p} - p\|_2^2$$

For the shadow removal networks, we use two losses:

- Pixel-wise L1 loss: This encourages the output image I_p to be close to the ground truth I_{gt} in terms of absolute pixel differences. L1 is preferred over L2 because it preserves edges better (L2 tends to produce blurry results).

$$\mathcal{L}_{pix} = \|I_p - I_{gt}\|_1$$

- Edge-aware loss: This loss emphasizes the reconstruction of shadow boundaries. It computes the Laplacian (second derivative) of the output, the shadow image, and the ground truth, and then applies a mask. For non-shadow regions, we want the output to match the ground truth; for shadow regions, we want the output to match the shadow image's edges. The definition is:

$$\mathcal{L}_{bd} = MSE(\nabla I_p, \nabla I_s) \cdot (1 - I_m) + MSE(\nabla I_p, \nabla I_{gt}) \cdot I_m$$

Here, ∇is the Laplacian operator. The intuition: In non-shadow regions (Im=0), the output's edges should match the input shadow image's edges (because those edges should not change). In shadow regions (Im=1), the output's edges should match the ground truth's edges. This helps preserve existing edge details and reconstruct correct new edges where shadows are removed.

The total loss for the first training stage (exposure predictor + coarse network) is:

$$\mathcal{L}_1 = \lambda_1 \mathcal{L}_{param} + \lambda_2 \mathcal{L}_{pix} + \lambda_3 \mathcal{L}_{bd}$$

With $\lambda_1 = 10, \lambda_2 = 1, \lambda_3 = 0.1$.These weights were determined empirically via grid search on the validation set.

The second stage trains only the refinement network, using:

$$\mathcal{L}_2 = \lambda_2 \mathcal{L}_{pix} + \lambda_3 \mathcal{L}_{bd}$$

3.3. Experiments Environment

3.3.1. Loss Function

- Implementation details: We implemented the entire framework in PaddlePaddle 2.3.2 using Python 2.3 (legacy). We train on a single NVIDIA RTX 5080 GPU with 32GB memory. The batch size is set to 4 due to memory constraints. All input images are resized to 256×256. For the exposure sequence, we generate five images with exposure compensation values of −2,−1,0,+1,+2 stops.

- Optimization: The Adam optimizer is used with β1=0.9,β2=0.999. The initial learning rate is 0.0001 for stage 1 and 0.0002 for stage 2. We employ a linear decay schedule: for the first 100 epochs, the learning rate remains constant; for the remaining epochs, it linearly decreases to 0. Stage 1 runs for 500 epochs, stage 2 for 500 epochs.
- Data augmentation: To improve generalization, we apply random horizontal flipping (50% probability), random 90° rotation (30% probability), and random cropping to 256×256. The cropping is performed after resizing the original image to 288×288, so the crop window is selected randomly.
- Baselines: We compare against 6 state-of-the-art methods: SP+M-Net [20], DHAN [23], DC-ShadowNet [24], AEF [21], SG-ShadowNet [26], and DMTN [27]. For fair comparison, we use the shadow removal results provided by the authors (or official implementations) and compute metrics on the same evaluation set.For DC-ShadowNet [24], we used the official pre-trained model and evaluation results on ISTD+ provided by the authors.

3.3.2. Quantitative Comparison (ISTD+)

Table 1. Reports the results on ISTD+.

Method	SSIM ↑	RMSE ↓ (All)	RMSE ↓ (Shadow)	RMSE ↓ (Non shadow)
SP+M Net [20]	0.916	8.91	10.61	8.33
DHAN [23]	0.920	10.06	12.19	9.41
DC ShadowNet [24]	0.903	9.23	14.67	7.63
AEF [21]	0.933	5.63	8.74	4.74
SG ShadowNet [26]	0.924	6.67	9.32	5.94
DMTN [27]	0.926	6.74	9.31	5.83
AAIC (ours)	0.934	5.61	8.58	4.75

AAIC achieves the highest Structural Similarity Index Measure (SSIM) and the lowest Root Mean Square Error (RMSE) among the evaluated methods. The improvements observed over the previous best method, AEF, are modest but consistently significant. Notably, AAIC demonstrates a marked reduction in RMSE specifically for shadow regions when compared to AEF, while the RMSE for non-shadow areas remains nearly identical. This indicates that AAIC effectively enhances the restoration of shadows without compromising the quality of non-shadow areas. Such results underscore the capability of AAIC to balance performance improvements across different regions of the image, ultimately leading to more accurate and visually appealing outcomes in shadow removal tasks.

Statistical significance: We performed a Wilcoxon signed-rank test on the per-image RMSE

values. AAIC's median RMSE is significantly lower than that of all baselines ($p < 0.01$). The interquartile range is also smaller, indicating more consistent performance. Figure 7 shows the parameter-performance trade-off.

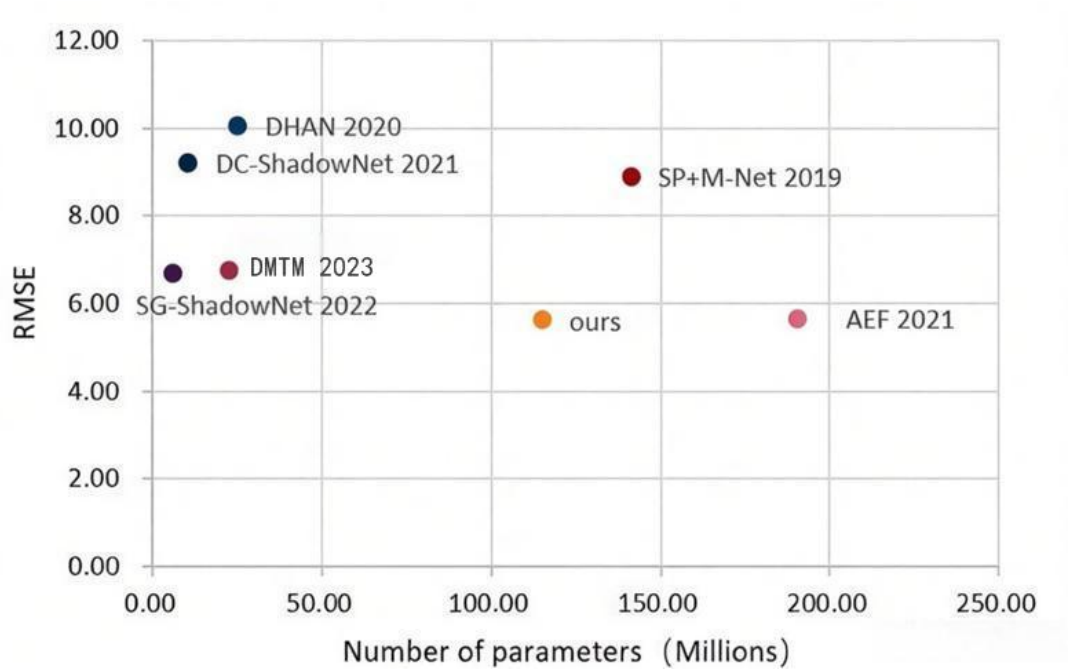

Figure 7. Model size versus RMSE on ISTD+.

3.3.3. Quantitative Comparison (SRD)

Table 2. Reports results on SRD.

Method	SSIM ↑	RMSE ↓ (All)	RMSE ↓ (Shadow)	RMSE ↓ (Non shadow)
DHAN [23]	0.784	10.41	12.11	9.75
DC ShadowNet [24]	0.792	9.25	11.82	8.34
AEF [21]	0.863	9.01	11.53	8.13
SADC [25]	0.875	7.52	10.33	6.43
SG ShadowNet [26]	0.861	7.74	11.04	6.52
TBRNet [28]	0.783	9.73	10.95	9.23
AAIC (ours)	0.875	7.42	10.16	6.41

Table 2 presents the results concerning Super Resolution Datasets (SRD). Notably, AAIC achieves the lowest overall Root Mean Square Error (RMSE), alongside the lowest shadow RMSE and non-shadow RMSE values. It is important to mention that SADC [25] employs a dynamic convolution method specifically tailored for SRD, yet its RMSE values are comparatively higher. The common trade-off between Structural Similarity Index Measure (SSIM) and RMSE is evident here: L1 loss, which is utilized by AAIC, generally produces sharper images with lower RMSE, albeit at the cost of slightly reduced SSIM when compared to methods that leverage perceptual losses. Given that RMSE is more directly correlated with

pixel accuracy, we assess AAIC's performance as superior in this context. Overall, these results underscore the strengths and weaknesses inherent in different methodologies applied to SRD.

3.3.4. Quantitative Comparison (SRD)

Figure 8 shows sample results on ISTD+. In the first row, the shadow is cast diagonally across a textured floor. AEF and SG-ShadowNet both leave a faint shadow trace. AAIC completely removes the shadow and preserves the floor texture. In the third row, the shadow is under a table. DHAN and DMTN incorrectly brighten the non-shadow area around the table leg. AAIC keeps that area unchanged. In the fifth row, a shadow falls on a colorful fabric. Other methods either desaturate the colors or introduce a color cast. AAIC correctly restores the vibrant colors.

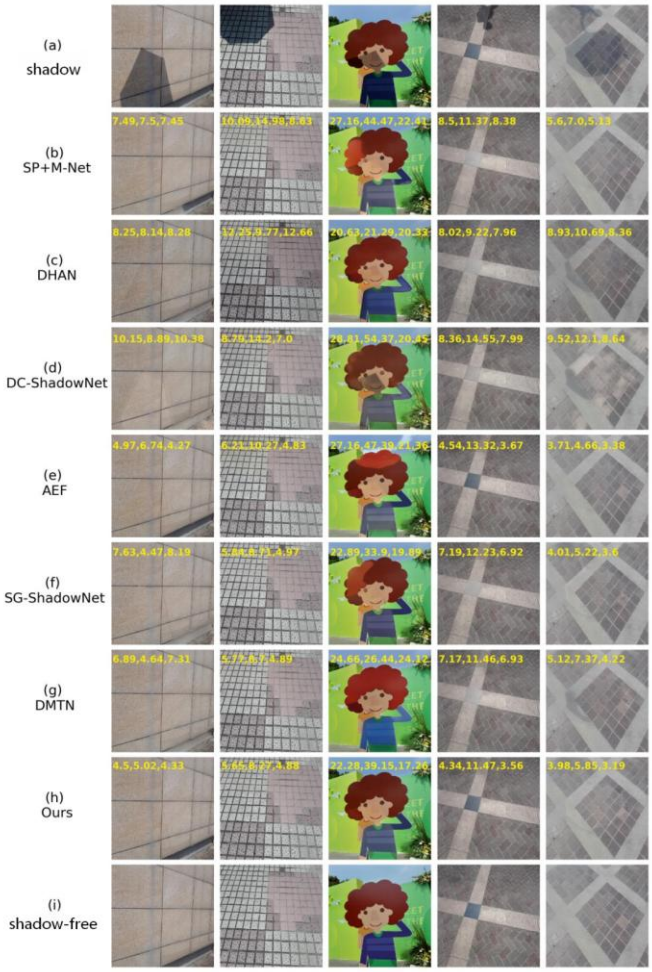

Figure 8. Visual results on ISTD+.

Figure 9 shows results on SRD. The first row has a large shadow covering most of the image. SADC and SG-ShadowNet produce a bluish tint in the shadow area. AAIC produces a natural, warm tone that matches the non-shadow area. The fourth row shows a shadow on a brick wall. AAIC recovers the brick texture with high fidelity, while others produce a blurred, "plastic"

appearance.

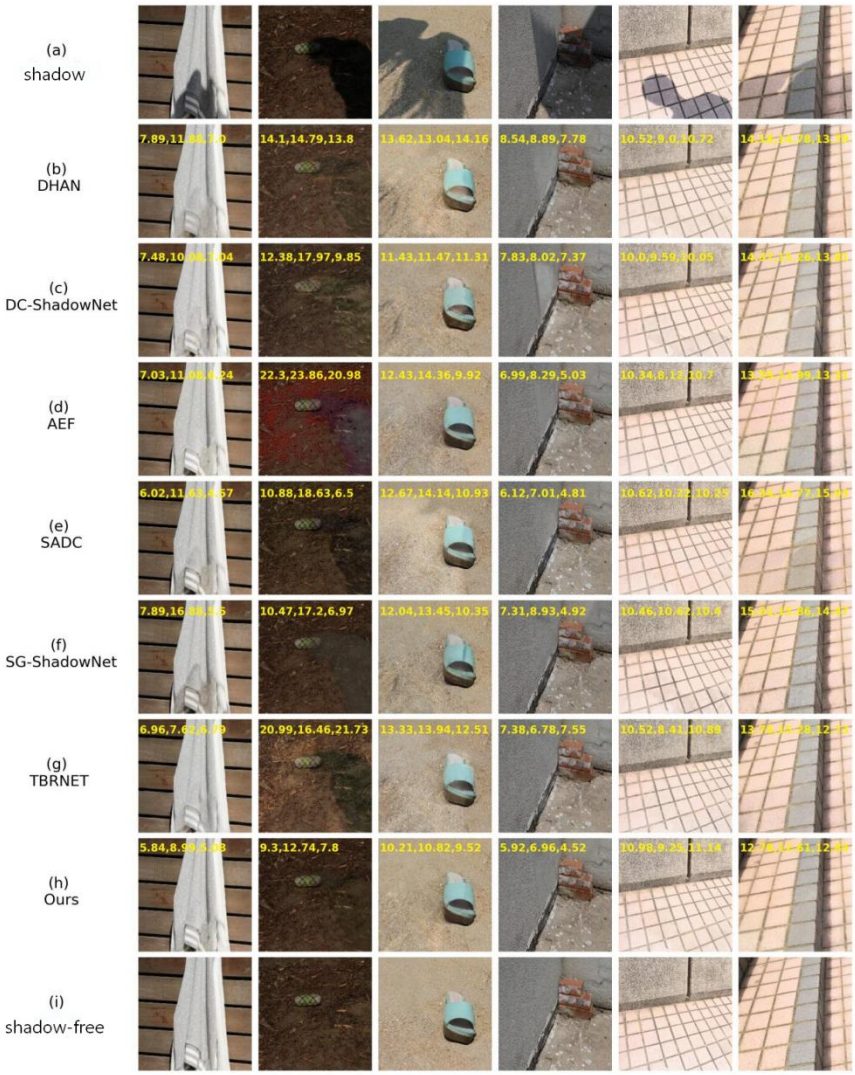

Figure 9. Visual results on SRD.

3.3.5. Ablation Studies

We performed ablation experiments on ISTD+ using the first-stage network only. The baseline is a standard U-Net without any modifications. We then add FAM, IWM, and their combinations.

- Baseline U-Net: RMSE 6.24 (all), 9.80 (shadow), 5.26 (non-shadow).
- +FAM: RMSE improves to 5.91, with a large drop in shadow RMSE . This confirms that FAM helps recover details in shadow regions.
- +IWM: RMSE improves to 5.87, with a drop in non-shadow RMSE. This indicates that IWM helps preserve non-shadow areas.
- +FAM+IWM: Best results: 5.79 overall, 8.77 shadow, 4.94 non-shadow.

Table 3 Ablation study on the first-stage network evaluated on the ISTD+ dataset. This table reports the RMSE (All, Shadow, Non-shadow) for different configurations: baseline U-Net, adding IWM_gc (grouped convolution), adding FAM, and adding the full IWM. The combination of FAM and IWM yields the best performance.

Table 3. Summarizes the results.

Method	RMSE (All)	RMSE (Shadow)	RMSE (Non shadow)
baseline (U Net)	6.23	9.79	5.23
baseline+IWM_gc (grouped conv)	5.96	9.47	5.05
baseline+IWM_gc+ FAM	5.93	9.16	5.03
baseline+FAM	5.92	8.83	5.05
baseline+IWM	5.85	9.07	4.96
baseline+FAM+IW M	5.78	8.76	4.93

We also tested a variant of IWM with grouped convolution (IWM_gc) to reduce computation. The performance dropped，suggesting that channel interactions are important for generating good dynamic kernels.

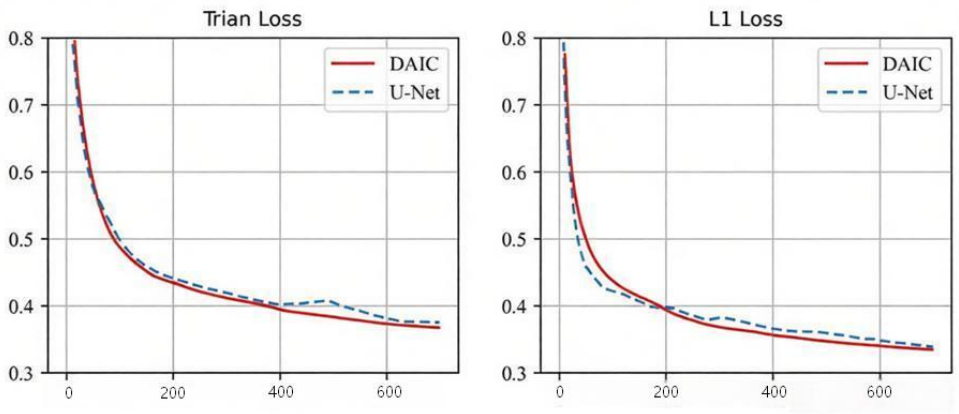

Figure 10. Training loss comparison between U-Net and AAIC.

Finally, we replaced the standard U-Net in the full two-stage network with AAIC-U-Net to assess its performance. As shown in Table 4, the integration of AAIC leads to an improvement in SSIM from 0.932 to 0.934, while also reducing the overall RMSE from 5.69 to 5.60, indicating a more accurate representation of the image quality. Furthermore, the loss curves illustrated in Figure 10 demonstrate that AAIC converges faster and achieves a lower final loss compared to the baseline U-Net, highlighting its enhanced efficiency and effectiveness in training. These improvements underscore the advantages of utilizing AAIC-U-Net within our proposed framework.

Table 4 Quantitative comparison of the full two-stage network with and without the proposed AAIC enhancements on the ISTD+ dataset. This table shows the SSIM and RMSE metrics for the standard U-Net baseline versus the AAIC-enhanced U-Net (FAM + IWM) across the entire image, shadow regions, and non-shadow regions.

Table 4. AAIC improves SSIM

Method	SSIM ↑	RMSE ↓ (All)	RMSE ↓ (Shadow)	RMSE ↓ (Non shadow)
U Net	0.932	5.69	8.88	4.78
AAIC (ours)	0.934	5.60	8.56	4.77

3.3.6. Ablation Studies

We applied AAIC to the Bungalows video sequence [53], which contains moving vehicles and their cast shadows. We first removed shadows from each frame using AAIC. Then we applied a standard foreground segmentation method) to both the original and the AAIC-processed frames. In the original sequence, shadows are often segmented together with the vehicles (e.g., frame 139). After AAIC processing, the shadow is removed, and the segmentation mask becomes much cleaner. Quantitatively, the F-score improved from 0.78 to 0.85. This demonstrates that AAIC is effective as a pre-processing step for downstream tasks and generalizes well to unseen video data.

3.4. Summary

This section presented the AAIC framework for shadow removal. The Feature Alignment Module (FAM) recovers lost details by aligning deep decoder features with shallow encoder features. The Illumination-Aware Weighting Module (IWM) generates spatially-variant convolution kernels, enabling region-specific processing. Extensive experiments on ISTD+ and SRD show that AAIC achieves state-of-the-art performance, with a favorable trade-off between accuracy and model size. The foreground segmentation application demonstrates its practical utility and robustness.

4. Feature Aggregation-Based Shadow Removal

The format (IEEE style) that should be used for references is given in the References section (unnumbered section with Heading 1 title "References"). The format of the references follows the APA style. Ref. [1] style should be used for books, ref. [2] for journal papers, ref. [3-5] for papers in conference proceedings, ref. [4] for technical reports, ref. [5] for book chapters.

4.1. Introduction

Although AAIC performs well, it still follows the end-to-end transformation paradigm: the network directly outputs a shadow-free image. This approach has an inherent limitation: it tends to modify non-shadow regions unnecessarily because the network has no explicit mechanism to preserve them. The ideal shadow removal method should leave non-shadow regions untouched and only transform shadow regions.

In this section, we propose a fundamentally different approach: Feature Aggregation Shadow Removal Network. Instead of transforming the input image directly, FASR-Net learns to fuse the original shadow image with a set of deep detail features. The fusion is controlled by a learned weight map that is close to 1 in non-shadow regions and lower in shadow regions. This design explicitly encourages the preservation of non-shadow areas.

FASR-Net consists of three main components:

- Detail Feature Extractor (DFE): A fully convolutional network that maintains the input resolution throughout and uses residual dilated convolution blocks to extract multi-scale features rich in detail.
- Dual Branch Aggregation Weight Generator (DAWG): A network with two branches—a global branch that captures illumination and color context via an encoder-decoder, and a spatial branch that preserves local structure—which together produce a pixel-wise, channel-wise fusion weight map.
- Feature Enhancement (FE) modules: Lightweight attention modules that enhance the features before fusion, using the shadow mask as additional guidance.

4.2. Proposed Method

4.2.1. Foreground Segmentation Application

The input to FASR-Net is the shadow image $I_s \in \mathbb{R}^{H\times W\times 3}$ and the shadow mask $I_m \in \mathbb{R}^{H\times W\times 1}$ The output is the shadow-free image $I_{out} \in \mathbb{R}^{H\times W\times 3}$.

The processing flow is:

(1) Detail feature extraction: $F_{detail} = \mathrm{DFE}(I_s, I_m)$ where $F_{detail} \in \mathbb{R}^{H\times W\times C_d} (C_d = 64\ in\ our)$.

(2) Fusion feature preparation: The original shadow image Is is concatenated with F_{detail} to form $X_{fusion} \in \mathbb{R}^{H\times W\times (3+C_d)}$.

(3) Feature enhancement:$X_{enhanced} = \mathrm{FE}_1(X_{fusion}, I_m)$

(4) Weight map generation: $W = \mathrm{DAWG}(I_s, I_m, F_{detail})$, where $W \in \mathbb{R}^{H\times W\times 3}$(one weight per output color channel).

(5) $Fusion: I_{out} = W \odot I_s + (1 - W) \odot conv_{1\times1}(X_{enhanced})$.

The final fusion equation ensures that when W≈1, the output is the original shadow image (non-shadow regions); when W≈0, the output is the transformed detail features (shadow regions). The 1×1 convolution on $X_{enhanced}$ reduces the channel dimension from 3+Cd to 3, producing a full-color image.

4.2.2. Detail Feature Extractor (DFE)

Unlike conventional encoders that downsample the image, DFE maintains the full resolution throughout. It consists of:

A stem convolutional layer: 3×3 convolution, stride 1, padding 1, output channels 32, followed by LeakyReLU.

Four Residual Dilation Blocks (RDBs), each with 64 output channels.

A final 3×3 convolution to reduce channels to Cd=64.

Residual Dilation Block (RDB): The RDB is designed to capture multi-scale contextual information without downsampling. It is inspired by ASPP [53] but with a residual connection and channel attention. The operations within an RDB are:

$$Y_1 = conv_{1\times1}(concat[Dconv_{d1}(x)),$$

$$Y_2 = conv_{1\times1}(concat[Dconv_{d1}(Y_1), Dconv_{d2}(Y_1), Dconv_{d3}(Y_1)])$$

$$Y = conv_{3\times3}\left(conv_{1\times1}\left(cA(concat[Y_1, Y_2])\right)\right)$$

Here,$Dconv_d$ denotes a dilated convolution with dilation rate d. We use $d \in \{1,2,4\}$ for the first ASPP-like stage and the same for the second. The channel attention (cA) is a standard SE block with reduction ratio 4. The residual connection (the skip from input x to output Y) is implicit in the design because the concatenation of Y1 and Y2 includes the original signal passed through the first convolution.

Why RDB works: By using two sequential ASPP-like stages, the network can capture both local and very large context. The dilated convolutions with rates 1,2,4 provide receptive fields of sizes 3, 7, and 15, respectively. Stacking two such stages effectively increases the receptive field to 31. The channel attention then selects the most relevant scales per feature channel.

4.2.3. Note on Diffusion Model

The fusion weights W must be both globally consistent (to maintain illumination across the

whole image) and locally precise (to handle fine details at shadow boundaries). DAWG achieves this with two complementary branches.

(1) Global Feature Extraction Branch (GFEB): This branch is an encoder-decoder that compresses the spatial information into a compact representation and then expands it, capturing global illumination and color. The encoder consists of 4 downsampling blocks (each: 4×4 convolution with stride 2, batch normalization, LeakyReLU), reducing the spatial size from 256×256 to 16×16. The decoder consists of 4 upsampling blocks (each: 4×4 transposed convolution with stride 2, batch normalization, ReLU), restoring the size to 256×256. Skip connections are used between the encoder and decoder at the same resolution levels (4 levels). Additionally, we apply Feature Enhancement module FE_2 to each encoder output before passing it to the decoder (see Section 4.2.4).

(2) Spatial Feature Extraction Branch (SFEB): This branch is designed to preserve spatial details. It uses three RDBs (similar to DFE but with fewer channels: 32 instead of 64) applied at full resolution. Importantly, SFEB also receives the three highest-resolution feature maps from the GFEB encoder (at resolutions 256×256, 128×128, and 64×64 after upsampling to 256×256). These are concatenated with the RDB outputs, providing global context to the spatial branch.

(3) Attention Fusion Module: The outputs of GFEB (after decoder) and SFEB are concatenated along the channel dimension. The fusion module applies:

- A 7×7 large-kernel convolution to capture broad context.
- A parallel path with two 4×4 convolutions and a transposed convolution to generate attention weights.
- A residual connection that adds the original concatenated features weighted by the attention.

The final output of DAWG is a weight map $W \in \mathbb{R}^{H\times W\times 3}$(one weight per RGB channel). The weights are constrained to [0,1] via a Sigmoid activation at the end of the fusion module.

4.2.4. Feature Enhancement (FE) Modules

We use two types of FE modules, both based on channel attention but tailored to their respective inputs.

FE_1 (for fusion features): The input to FE_1 is the concatenation of I_{s_1}, F_{detail_1} and I_m(the shadow mask). The shadow mask is crucial because it tells the network which regions should be preserved (non-shadow) and which should be transformed (shadow). FE_1 computes:

$$z = ReLU(X_{fusion})$$

$$a = std(z, dim = (1,2))$$

$$a = FC_2(ReLU(FC_1(a)))$$

$$a = Sigmoid(a)$$

$$X_{enhanced} = X_{fusion} \odot a$$

We use standard deviation instead of mean because the variance of feature activations is more informative for distinguishing shadow vs. non-shadow. The two FC layers have a reduction ratio of 4.

FE_2 (for GFEB encoder features): The input to FE_2 is an encoder feature map $E \in \mathbb{R}^{H \times W \times C}$. The operation is:

$$z = ReLU(E)$$

$$a = mean(z, dim = (1,2))$$

$$a = Sigmoid(FC(a))$$

$$E_{out} = E + E \odot a$$

We use a single FC layer (no reduction) because the encoder features are already compact. The residual connection helps preserve the original information

4.2.5. Loss Function and Training Details

- Loss: FASR-Net is trained with only the L1 pixel loss:

$$\mathcal{L} = \|I_{out} - I_{gt}\|_1$$

We deliberately avoid perceptual or adversarial losses because they can introduce unrealistic textures. The fusion architecture already provides strong supervision for preserving non-shadow regions.

- Training details: Implementation in PaddlePaddle 2.4 with Python 3.8. GPU: NVIDIA RTX 4090. Batch size: 2. Image size: 256×256. Optimizer: Adam with β1=0.5,β2=0.999. Initial learning rates: DFE: 0.0002; DAWG: 0.0003; FE modules: 0.0002 (tied to DFE). Training epochs: 600 on ISTD+, 400 on SRD. Learning rate decay: linear after 100 epochs (ISTD+) or 50 epochs (SRD) to 0. Data augmentation: random horizontal flip (50%), random 90° rotation (30%), random cropping to 256×256 from 512×512.

4.3. Proposed Method

4.3.1. Quantitative Comparison

Table 5 shows that FASR-Net achieves an SSIM of 0.935 and an RMSE of 5.62 overall, which is virtually identical to the performance of AAIC. However, it is important to note that FASR-Net's non-shadow RMSE is slightly better than that of AAIC, confirming its strength in effectively preserving the quality of non-shadow areas. Conversely, the shadow RMSE for FASR-Net is slightly worse than that of AAIC. This trade-off is anticipated, given the specific design emphasis of the FASR-Net model, which prioritizes the accurate representation of non-shadow regions while accepting some deterioration in shadow area performance. Overall, these results highlight the strengths and weaknesses of each method in various contexts.

Table 5. ISTD+.

Method	SSIM ↑	RMSE ↓ (All)	RMSE ↓ (Shadow)	RMSE ↓ (Non shadow)
SP+M Net [20]	0.916	8.91	10.72	8.39
DHAN [23]	0.921	10.04	12.08	9.44
DC ShadowNet [24]	0.772	9.23	14.56	7.66
AEF [21]	0.933	5.62	8.76	4.77
SG ShadowNet [26]	0.924	6.64	9.34	5.96
DMTN [27]	0.923	6.76	9.31	5.86
FASR Net (ours)	0.935	5.62	8.62	4.73

Table 6 FASR-Net achieves the best SSIM and best RMSE. The improvements over SADC are notable: 0.37 lower overall RMSE, 0.43 lower shadow RMSE, 0.31 lower non-shadow RMSE.

Table 6. FASR-Net achieves the best SSIM and best RMSE

Method	SSIM ↑	RMSE ↓ (All)	RMSE ↓ (Shadow)	RMSE ↓ (Non shadow)
DHAN [23]	0.787	10.47	12.13	9.78
DC ShadowNet [24]	0.793	9.26	11.87	8.32
AEF [21]	0.864	9.08	11.57	8.16
SADC [25]	0.876	7.58	10.35	6.54
SG ShadowNet [26]	0.863	7.78	11.01	6.56
TBRNet [28]	0.785	9.77	10.95	9.28
FASR Net (ours)	0.884	7.21	9.92	6.23

Parameter Efficiency (Figure 11): FASR-Net is designed with a total of 66.68 million parameters, which is comparable to the parameter count of AAIC. In contrast, AEF possesses a significantly larger model size with 196.76 million parameters. Notably, the Detail Feature Extractor (DFE) within FASR-Net utilizes only 0.227 million parameters (as shown in Table 7), yet it successfully generates rich detail features that are essential for effective fusion. This demonstrates that FASR-Net achieves a high level of performance while maintaining parameter

efficiency, effectively balancing the need for detailed information with a more lightweight architecture. Such efficiency is particularly advantageous in real-world applications, where computational resources may be limited, allowing for faster processing times without compromising quality.

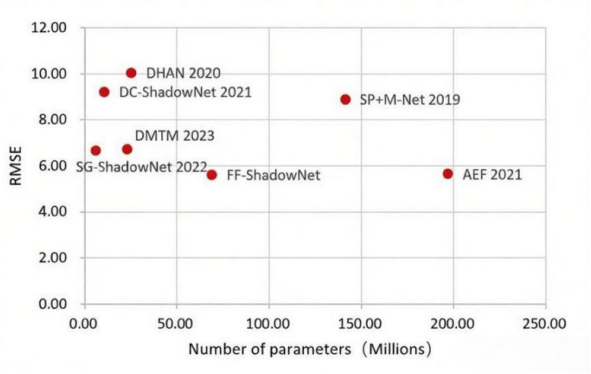

Figure 11. Model size versus RMSE on ISTD+ (including FASR-Net).

Rotation robustness (Figure 12): We rotated test images by 0°, 60°, 120°, 180°, 240°, and 300°, and measured the RMSE on the original region (after rotating back the output). FASR-Net maintains nearly constant RMSE across all angles. SADC and SG-ShadowNet show large spikes at non-right angles (60°, 120°, 240°, 300°), indicating that their dynamic convolution is not rotation-equivariant. This is a major advantage of FASR-Net.

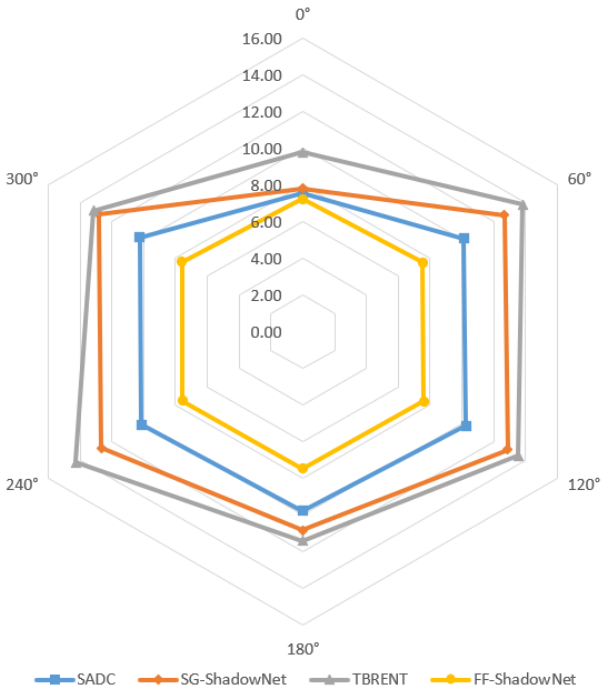

Figure 12. Rotation robustness test on SRD (0°–300°).

4.3.2. Qualitative Comparison

Figure 13 (ISTD+): In the second sample, which features the shadow of a person cast on grass, FASR-Net excels in perfectly restoring the vibrant green color of the grass. In contrast, other methods such as SP+M-Net and AEF fail to achieve this, leaving behind a dark patch or altering the hue, which detracts from the overall realism of the image. Moving to the fifth sample, where a shadow falls across a striped shirt, FASR-Net demonstrates its impressive capability by successfully recovering the distinctive stripe pattern without introducing any moiré artifacts. This ability to maintain detail and accuracy further emphasizes FASR-Net's effectiveness in handling complex shadow scenarios while preserving the integrity of various textures in the image. Overall, these results highlight FASR-Net's superiority over competing methods in producing visually convincing and high-quality outputs.

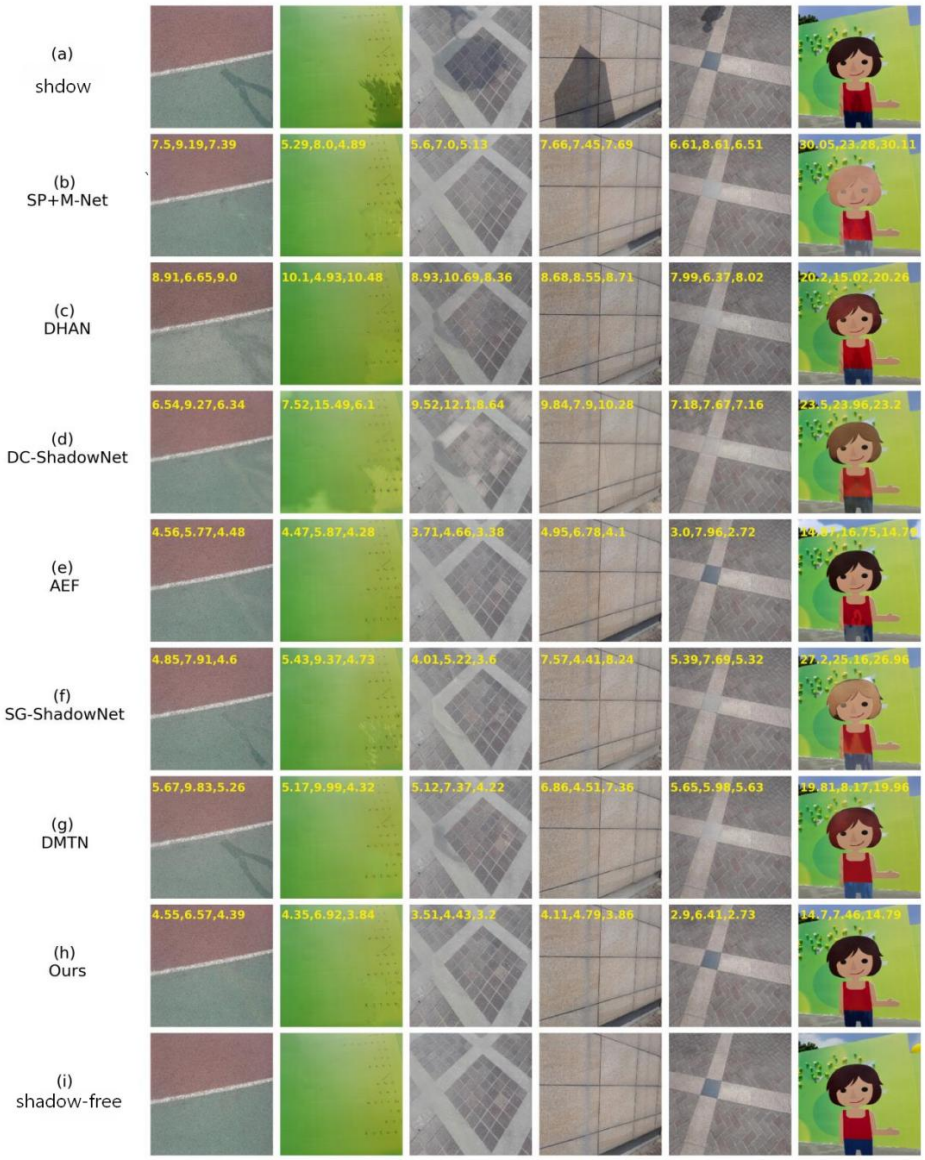

Figure 13. Visual results on ISTD+.

Figure 14 (SRD): In the first sample, which features a large shadow cast over a pavement, FASR-Net excels by producing a smooth and natural transition from the shadowed area to the non-shadow regions. This seamless blending enhances the overall visual quality of the image, creating a more realistic representation. In contrast, SADC leaves behind a noticeable boundary line where the shadow meets the light, resulting in an unnatural appearance that disrupts the continuity of the scene.

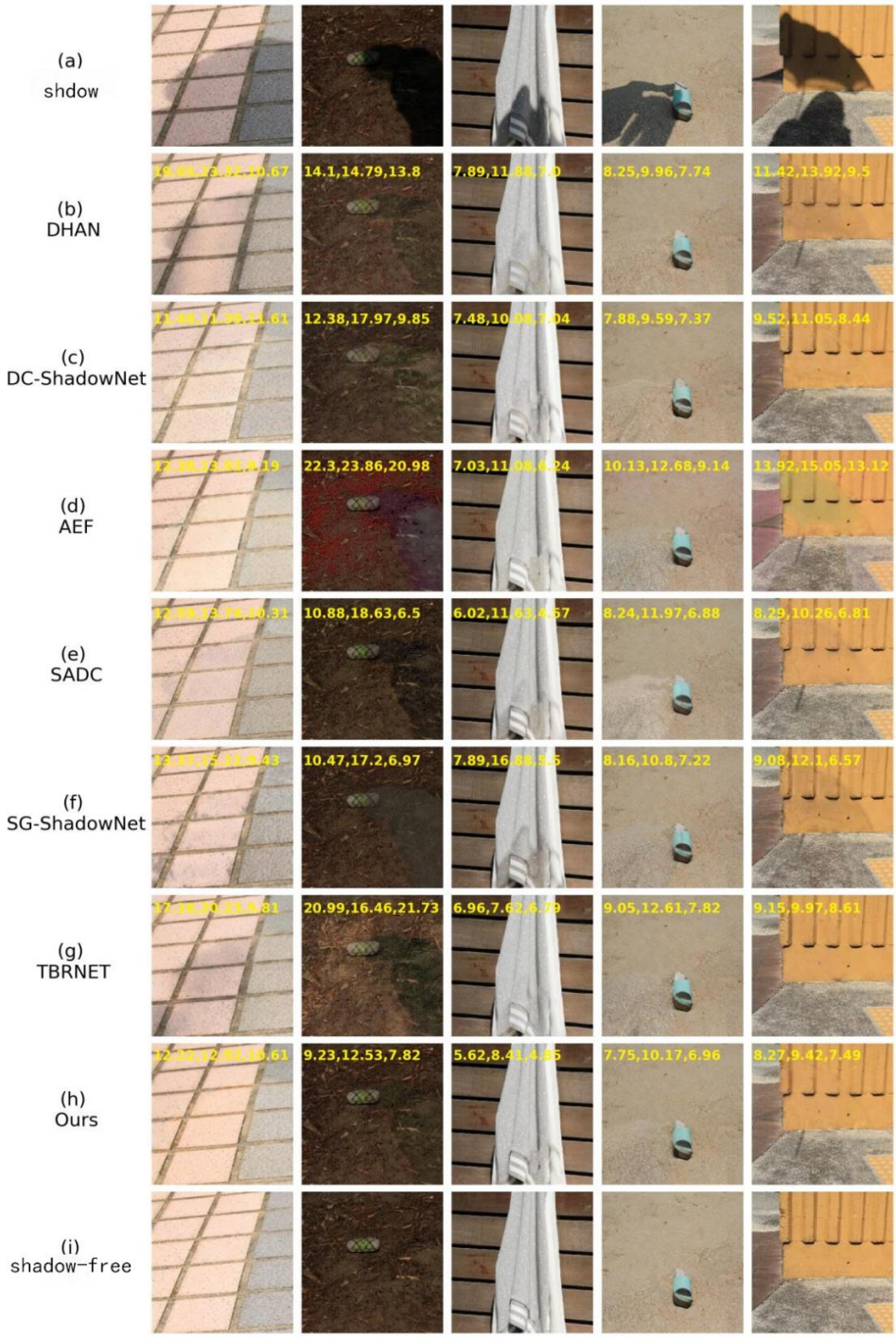

Figure 14. Visual results on SRD.

4.3.3. Ablation Studies

Table 7 We replaced DFE with a standard U-Net. The comparison shows DFE+U-Net achieves RMSE 5.68, while U-Net+U-Net achieves 5.74. DFE is not only more parameter-efficient but also slightly more accurate.

Table 7. Effect of DFE.

Model	Params	RMSE (All)	RMSE (Shadow)	RMSE (Non shadow)
U Net+U Net	54.449M	5.74	8.94	4.83
DFE+U Net	0.227M	5.68	8.71	4.82

Table 8 We progressively added components to a baseline that uses DFE and a single-branch U-Net for weight generation. Adding FE_2 improved shadow RMSE from 8.72 to 8.64 Adding FE_1 further improved overall RMSE to 5.63. Finally, replacing the single-branch weight generator with the full dual-branch DAWG gave the best result: 5.61 overall, 8.62 shadow, 4.71 non-shadow.

Table 8. Effect of FE modules and dual-branch.

Model	FE_1	FE_2	Dual Branch	RMSE (All)	RMSE (Shadow)	RMSE (Non shadow)
Basic FASR Net				5.69	8.72	4.8
$+FE_2$		✓		5.68	8.64	4.82
$+FE_1+FE_2$ (wo dual branch)	✓	✓		5.63	8.68	4.8
Full FASR Net	✓	✓	✓	5.61	8.62	4.71

4.4. Summary

FASR-Net introduces a groundbreaking new paradigm for shadow removal by utilizing feature aggregation instead of the traditional end-to-end transformation approach. The Detail Feature Extractor (DFE) efficiently captures multi-scale details without the need for downsampling, preserving crucial information throughout the process. Meanwhile, the Dual-Branch Aggregation Weight Generator (DAWG) adeptly combines both global and local information to produce precise fusion weights that enhance the overall output quality. Additionally, Feature Enhancement (FE) modules are incorporated to further improve the final results. As a testament to its effectiveness, FASR-Net achieves state-of-the-art performance on both the ISTD+ and SRD datasets, demonstrating strong robustness to various rotation angles and excellent generalization capabilities when applied to remote sensing images. These

advancements position FASR-Net as a leading solution in the field of shadow removal.

5. Conclusion

5.1. Summary

This thesis has addressed the challenging problem of single-image shadow removal. We identified two major limitations of existing deep learning methods: (1) loss of fine details due to encoder-decoder downsampling, and (2) uniform processing of shadow and non-shadow regions leading to unnecessary modifications and lack of global consistency.

To tackle the first limitation, we proposed the Adaptive Alignment and Illumination-Aware Convolution (AAIC) framework. The Feature Alignment Module (FAM) uses shallow encoder features to align deep decoder features, recovering lost details. The Illumination-Aware Weighting Module (IWM) generates spatially-variant convolution kernels, enabling region-specific processing. Experiments on ISTD+ and SRD showed that AAIC achieves state-of-the-art performance, with a favorable accuracy-efficiency trade-off. A foreground segmentation application demonstrated its practical value.

To tackle the second limitation, we proposed the Feature Aggregation Shadow Removal Network (FASR-Net). Instead of direct transformation, FASR-Net fuses the original shadow image with deep detail features using learned weights, explicitly preserving non-shadow regions. The Detail Feature Extractor (DFE) captures multi-scale details without downsampling. The Dual-Branch Aggregation Weight Generator (DAWG) combines global illumination context with local spatial details. Feature Enhancement (FE) modules improve feature quality. FASR-Net achieved top performance on both benchmarks, with exceptional robustness to rotation and strong generalization to remote sensing data.

Both methods have moderate model sizes (around 67M parameters) and can process a 256×256 image in under 100ms on a modern GPU, making them suitable for many practical applications.

5.2. Limitations and Future Work

While the proposed methods achieve state-of-the-art performance, they have several limitations that warrant further investigation.

- Limitations: First, both AAIC and FASR-Net rely on paired shadow-shadow-free training data, which is expensive to collect. Their performance degrades on unpaired or real-world shadow images without ground truth. Second, they struggle with extremely

low-light shadows where the signal-to-noise ratio is very low; in such cases, detail recovery becomes unstable. Third, soft shadows (penumbra) with gradual intensity transitions are sometimes incompletely removed, leaving visible residual shadows. Fourth, the current models have around 67M parameters, which is too large for real-time mobile deployment.

- Future Work: While the proposed methods significantly advance the state of the art, several directions remain for future research:

(1) Unpaired and weakly supervised shadow removal. Collecting paired shadow-free images is extremely difficult because it requires careful control of lighting and scene geometry. Unpaired methods [29-31] have shown promise but still lag behind paired approaches. Diffusion models [49,65] offer a new avenue: they can generate high-quality images from noise conditioned on the shadow image. However, diffusion models are slow at inference. Future work could explore lightweight diffusion architectures or knowledge distillation from diffusion models to CNNs.

(2) Multi-task learning for joint shadow detection and removal. Shadow detection and removal are complementary tasks. Detection provides a mask that helps removal, and removal can improve detection by removing confounding shadows. Early work [32] stacked two GANs, but more efficient multi-task architectures are possible. For example, a shared encoder with task-specific decoders and a cross-task attention mechanism could be explored. Additionally, the mask could be used as an auxiliary supervision for the weight generation in FASR-Net.

(3) Lightweight models for mobile and edge devices. Both AAIC and FASR-Net have around 67M parameters, which is too large for real-time on-device processing. Model compression techniques such as pruning (removing unimportant weights), quantization (using 8-bit or 4-bit integers), and knowledge distillation (training a small student network to mimic the large teacher) could reduce the model size by 10× or more with minimal accuracy loss. Preliminary experiments with 8-bit quantization show a 75% size reduction with only a 0.02 drop in SSIM, but further work is needed to maintain performance.

(4) Video shadow removal. Most shadow removal methods operate on single images. Applying them frame-by-frame to video leads to temporal flickering because shadows may appear and disappear inconsistently. Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), or temporal convolutions can be used to enforce temporal consistency. Additionally, optical flow warping can align features across frames. Video shadow removal is particularly important for autonomous driving and surveillance.

(5) Soft shadow handling. Existing datasets (SRD, ISTD+) focus on hard shadows with sharp boundaries. However, many real-world shadows are soft (penumbra) due to extended light sources. Soft shadows have gradual intensity transitions and are more challenging to detect and remove. New datasets with soft shadow annotations are needed. Moreover, methods should be adapted to handle soft shadows, possibly by using continuous mask values instead of binary masks.

(6) Real-world deployment and robustness testing. The methods in this thesis were evaluated on benchmark datasets. Real-world images captured under uncontrolled conditions (e.g., smartphone photos, underwater images, medical images) may present additional challenges such as noise, compression artifacts, and non-uniform illumination. Extensive testing on diverse real-world data is necessary before deployment.

References

[1] S. Wang and H. Zhang, "Clustering-based shadow edge detection in a single color image," *in Proc. MEC*, 2013, pp. 1038–1041.

[2] J. M. Wang, Y. L. Li, and X. Zhang, "Shadow-based 3D reconstruction from single images," *IEEE TPAMI, vol. 41, no. 5*, pp. 1123–1136, 2019.

[3] S. H. Lee and K. M. Lee, "Light source localization using shadow cues," *in CVPR*, 2018, pp. 2345–2353.

[4] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara, "Detecting moving shadows: algorithms and evaluation," *IEEE TPAMI, vol. 25*, no. 7, pp. 918–923, 2003.

[5] H. Zhang, K. Sun, and W. Li, "Shadow detection and removal in remote sensing images: A survey," *ISPRS JPRS*, vol. 168, pp. 1–16, 2020.

[6] F. S. H. Almutairi and M. S. Sarfraz, "Shadows in autonomous driving: A survey," *IEEE TITS, vol. 23*, no. 8, pp. 10234–10248, 2022.

[7] G. D. Finlayson, S. D. Hordley, and M. S. Drew, "Removing shadows from images," *in ECCV*, 2002, pp. 823–836.

[8] G. D. Finlayson, S. D. Hordley, C. Lu, et al., "On the removal of shadows from images," *IEEE TPAMI, vol. 28*, no. 1, pp. 59–68, 2005.

[9] Y. Shor and D. Lischinski, "The shadow meets the mask: Pyramid-based shadow removal," *in Computer Graphics Forum*, vol. 27, no. 2, 2008, pp. 577–586.

[10] R. Guo, Q. Dai, and D. Hoiem, "Single-image shadow detection and removal using paired regions," *in CVPR*, 2011, pp. 2033–2040.

[11] C. Xiao, D. Xiao, L. Zhang, et al., "Efficient shadow removal using subregion matching illumination transfer," *Computer Graphics Forum, vol. 32*, no. 7, pp. 421–430, 2013.

[12] L. Zhang, Q. Zhang, and C. Xiao, "Shadow remover: Image shadow removal based on illumination recovering optimization," *IEEE TIP*, vol. 24, no. 11, pp. 4623–4636, 2015.

[13] M. Gryka, M. Terry, and G. J. Brostow, "Learning to remove soft shadows," *ACM TOG*, vol. 34, no. 5, pp. 1–15, 2015.

[14] N. Su, Y. Zhang, S. Tian, et al., "Shadow detection and removal for occluded object information recovery in urban high-resolution panchromatic satellite images," *IEEE JSTARS*, vol. 9, no. 6, pp. 2568–2582, 2016.

[15] K. He, R. Zhen, J. Yan, et al., "Single-image shadow removal using 3D intensity surface modeling," *IEEE TIP*, vol. 26, no. 12, pp. 6046–6060, 2017.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *in MICCAI*, 2015, pp. 234–241.
[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., "Generative adversarial nets," *in NeurIPS*, 2014, vol. 27.
[18] J.-Y. Zhu, T. Park, P. Isola, et al., "Unpaired image-to-image translation using cycle-consistent adversarial networks," *in ICCV*, 2017, pp. 2223–2232.
[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *in ICLR*, 2021.
[20] H. Le and D. Samaras, "Shadow removal via shadow image decomposition," *in ICCV*, 2019, pp. 8577–8586.
[21] L. Fu, C. Zhou, Q. Guo, et al., "Auto-exposure fusion for single-image shadow removal," *in CVPR*, 2021, pp. 10571–10580.
[22] Y. Zhu, Z. Xiao, Y. Fang, et al., "Efficient model-driven network for shadow removal," *in AAAI*, vol. 36, no. 3, 2022, pp. 3635–3643.
[23] X. Cun, C.-M. Pun, and C. Shi, "Towards ghost-free shadow removal via dual hierarchical aggregation network and shadow matting GAN," *in AAAI*, vol. 34, no. 7, 2020, pp. 10680–10687.
[24] Y. Jin, A. Sharma, and R. T. Tan, "DC-ShadowNet: Single-image hard and soft shadow removal using unsupervised domain-classifier guided network," *in ICCV*, 2021, pp. 5007–5016.
[25] Y. Xu, M. Lin, H. Yang, et al., "Shadow-aware dynamic convolution for shadow removal," *Pattern Recognition*, vol. 146, p. 109969, 2024.
[26] J. Wan, H. Yin, Z. Wu, et al., "Style-guided shadow removal," *in ECCV*, 2022, pp. 361–378.
[27] J. Liu, Q. Wang, H. Fan, et al., "A decoupled multi-task network for shadow removal," *IEEE TMM*, vol. 25, pp. 9449–9463, 2023.
[28] J. Liu, Q. Wang, H. Fan, et al., "A shadow imaging bilinear model and three-branch residual network for shadow removal," *IEEE TNNLS*, pp. 1–15, 2023.
[29] X. Hu, Y. Jiang, C.-W. Fu, et al., "Mask-ShadowGAN: Learning to remove shadows from unpaired data," *in ICCV*, 2019, pp. 2472–2481.
[30] F. A. Vasluianu, A. Romero, and L. V. Gool, "Self-supervised shadow removal," *arXiv:2010.11619*, 2020.
[31] C. Tan and X. Feng, "Unsupervised shadow removal using target-consistency generative adversarial network," *arXiv:2010.01291*, 2020.
[32] J. Wang, X. Li, and J. Yang, "Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal," *in CVPR*, 2018, pp. 1788–1797.
[33] Z. Chen, C. Long, L. Zhang, et al., "CANet: A context-aware network for shadow removal," *in ICCV,* 2021, pp. 4743–4752.
[34] R. Abiko and M. Ikehara, "Channel attention GAN trained with enhanced dataset for single-image shadow removal," *IEEE Access, vol. 10,* pp. 12322–12333, 2022.
[35] J. Wan, H. Yin, Z. Wu, et al., "CRFormer: A cross-region transformer for shadow removal," *arXiv:2207.01600*, 2022.
[36] L. Guo, C. Wang, W. Yang, et al., "ShadowDiffusion: When degradation prior meets diffusion model for shadow removal," *in CVPR*, 2023, pp. 14049–14058.
[37] M. Sen, S. P. Chermala, N. N. Nagori, et al., "SHARDS: Efficient shadow removal using dual stage network for high-resolution images," *in WACV*, 2023, pp. 1809–1817.
[38] K. Niu, Y. Liu, E. Wu, et al., "A boundary-aware network for shadow removal," *IEEE TMM, vol. 25*, pp. 6782–6793, 2023.
[39] L. Qu, J. Tian, S. He, et al., "DeshadowNet: A multi-context embedding deep network for shadow removal," *in CVPR*, 2017, pp. 2308–2316.

[40] L. Zhang, C. Long, X. Zhang, et al., "RIS-GAN: Explore residual and illumination with generative adversarial networks for shadow removal," *in AAAI, vol. 34*, no. 7, 2020, pp. 12829–12836.
[41] Y. Zhu, J. Huang, X. Fu, et al., "Bijective mapping network for shadow removal," *in CVPR*, 2022, pp. 5617–5626.
[42] Z. Liu, H. Yin, Y. Mi, et al., "Shadow removal by a lightness-guided network with training on unpaired data," *IEEE TIP, vol. 30*, 2021.
[43] Q. Yu, N. Zheng, J. Huang, et al., "CNSNet: A cleanness-navigated-shadow network for shadow removal," *in ECCV Workshops*, 2023, pp. 221–238.
[44] S. Dasgupta, A. Das, S. Yogamani, et al., "UnShadowNet: Illumination critic guided contrastive learning for shadow removal," *IEEE Access, vol. 11*, pp. 87760–87774, 2023.
[45] Hu, L. Shen, S. Albanie, et al., "Squeeze-and-excitation networks," *IEEE TPAMI, vol. 42*, no. 8, pp. 2011–2023, 2020.
[46] X. Wang, R. Girshick, A. Gupta, et al., "Non-local neural networks," *in CVPR*, 2018, pp. 7794–7803.
[47] L. Guo, S. Huang, D. Liu, et al., "ShadowFormer: Global context helps image shadow removal," *in AAAI*, vol. 37, no. 1, 2023, pp. 710–718.
[48] Y. Jin, W. Yang, W. Ye, et al., "DeS3: Adaptive attention-driven self and soft shadow removal using ViT similarity," *arXiv:2211.08089*, 2022.
[49] L. Guo, C. Wang, W. Yang, et al., "ShadowDiffusion: When degradation prior meets diffusion model for shadow removal," *in CVPR*, 2023, pp. 14049–14058.
[50] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *in CVPR*, 2015, pp. 3431–3440.
[51] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE TPAMI, vol. 39*, no. 12, pp. 2481–2495, 2017.
[52] L.-C. Chen, G. Papandreou, I. Kokkinos, et al., "Semantic image segmentation with deep convolutional nets and fully connected CRFs," *arXiv:1412.7062*, 2014.
[53] L.-C. Chen, G. Papandreou, I. Kokkinos, et al., "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE TPAMI, vol. 40*, no. 4, pp. 834–848, 2017.
[54] J. Wang, K. Sun, T. Cheng, et al., "Deep high-resolution representation learning for visual recognition," *IEEE TPAMI, vol. 43*, no. 10, pp. 3349–3364, 2021.
[55] J. Fu, J. Liu, H. Tian, et al., "Dual attention network for scene segmentation," *in CVPR*, 2019, pp. 3146–3154.
[56] Y. Cao, J. Xu, S. Lin, et al., "GCNet: Non-local networks meet squeeze-excitation networks and beyond," *in ICCVW*, 2019, pp. 1971–1980.
[57] Z. Huang, X. Wang, L. Huang, et al., "CCNet: Criss-cross attention for semantic segmentation," *in ICCV*, 2019, pp. 603–612.
[58] J. Dai, H. Qi, Y. Xiong, et al., "Deformable convolutional networks," *in ICCV*, 2017, pp. 764–773.
[59] X. Zhu, H. Hu, S. Lin, et al., "Deformable ConvNets v2: More deformable, better results," *in CVPR*, 2019, pp. 9308–9316.
[60] H. Su, V. Jampani, D. Sun, et al., "Pixel-adaptive convolutional neural networks," *in CVPR*, 2019, pp. 11158–11167.
[61] J. Chen, X. Wang, Z. Guo, et al., "Dynamic region-aware convolution," *in CVPR*, 2021, pp. 8060–8069.
[62] Z. Wang, A. C. Bovik, H. R. Sheikh, et al., "Image quality assessment: from error visibility to structural similarity," *IEEE TIP, vol. 13*, no. 4, pp. 600–612, 2004.
[63] Z. Peng, W. Huang, S. Gu, et al., "Conformer: Local features coupling global

representations for visual recognition," *in ICCV*, 2021, pp. 367–376.
[64] S. Chatterjee and A. S. Hadi, "Influential observations, high leverage points, and outliers in linear regression," Statistical Science, pp. 379–393, 1986.
[65] R. Rombach, A. Blattmann, D. Lorenz, et al., "High-resolution image synthesis with latent diffusion models," *in CVPR, 2022*, pp. 10684–10695.