



Transmission-Aware Hybrid Image Deblurring Using Non-Local Means and Transformer Network with Edge-Preserving Improved Iterative Back Projection Cyber-Enhanced Refinement

M.S. Vinu^{1*} , S. Pavalarajan² 

¹ Computer Science and Business Systems, JCT College of Engineering and Technology, Coimbatore 641105, India

² Computer Science and Business Systems, PSNA College of Engineering and Technology, Dindigul 624622, India

Corresponding Author Email: vinu16913@gmail.com

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/ts.430413>

ABSTRACT

Received: 12 February 2026

Revised: 8 June 2026

Accepted: 17 June 2026

Available online: 31 August 2026

Keywords:

image deblurring, Non-Local Means, Deep Convolutional Neural Network, edge-aware loss, back projection, B-spline interpolation, computational photography

Reliable image restoration is critical in cyber systems, where visual data transmitted over unstable or bandwidth-limited channels often suffer from motion blur, noise, and compression artifacts. This paper proposes a Transmission-Aware Hybrid Deblurring Framework that integrates Non-Local Means (NLM) denoising, a Transformer-enhanced Deep Convolutional Neural Network (DCNN), and an Edge-Preserving Improved Iterative Back Projection (IIBP) refinement stage. NLM pre-processing suppresses noise while retaining repetitive structures, enabling the Transformer network to model complex blur patterns and recover high-frequency details more effectively. An Edge-Aware Loss further strengthens gradient preservation, producing sharper boundaries and improved structural fidelity. The cyber-enhanced IIBP module, supported by Cubic B-Spline interpolation, refines residual errors and ensures smooth sub-pixel transitions aligned with the forward blur model. Experimental evaluation on GoPro, RealBlur-R, and RealBlur-J datasets demonstrates that the proposed method consistently outperforms state-of-the-art approaches. The framework achieves 32.74 dB Peak Signal-to-Noise Ratio (PSNR) and 0.952 Structural Similarity Index Measure (SSIM) on GoPro, exceeding Restormer by +0.68 dB PSNR. It attains 30.10 dB PSNR and 0.927 SSIM on RealBlur-R, and 29.75 dB PSNR and 0.920 SSIM on RealBlur-J, along with improved Learned Perceptual Image Patch Similarity (LPIPS) and Mean Absolute Error (MAE) scores. These results confirm the robustness and generalizability of the hybrid NLM–Transformer–IIBP pipeline for transmission-affected deblurring tasks across diverse real-world imaging conditions.

1. INTRODUCTION

Deblurring is a persistent problem in computational photography and computer vision areas, with huge implications for medical imaging, autonomous driving, commercial surveillance, and consumer photography applications. Blur degradation often results from camera motion, inadequate focus, or potentially environmental interference. Blur degradation results in lost high frequency details, smeared edges [1], and loss of perceptual clarity, ultimately complicating further vision tasks, including object recognition or describing a scene. Traditional deblurring methods were based on inverse filters, Wiener deconvolution, and relaxation and variational approaches, typically with a simplistic blur kernel or no defined kernel, and were not effective in many situations with remnant noise and blurry frames. Additionally, Non-Local Means (NLM) algorithms [2], which exploit self-similarity to suppress noise, do perform well but suffer somewhat from the problem of blurring when the blur is spatially variant. In comparison, research with deep learning methods, specifically, Deep Convolutional Neural Networks (DCNNs) [3-5] have produced promising results

establishing a complicated mapping of blur to sharpness, but may present textures that are not representative or have no structure due to being classification techniques. Iterative Back Projection (IBP) methods don't have a theoretical limitation on ensuring consistency and could result in edges going out of alignment. To reinforce these perspectives, we present a hybrid framework to develop a combined NLM denoising classifier and improve processing on a DCNN. The NLM module first preprocesses the input by reducing noise while retaining edges, and acts as a regularizer for the upcoming steps. Next, the DCNN, trained with the proposed Edge-Aware Loss that has spatial gradient penalties, presumably favors high frequency reconstruction and edge fidelity, and works around the major challenge of texture loss from data-driven methods.

The Improved Iterative Back Projection (IIBP) module then minimizes the reconstruction error against a forward model of blur with the Convolutional Neural Network (CNN) output, while Cubic B-Spline with gradient interpolation provides both sub-pixel accuracy and a smooth transition between frames. The Edge-Aware Loss also adds gradient-based nudges to the optimization target of the CNN, enforcing

geometrical consistency and alleviating blur-artifact trade-offs. This holistic approach not only leverages the strengths of each component but also mitigates their individual weaknesses: NLM's dependency on patch similarity, CNN's tendency toward overfitting, and IBP's sensitivity to initialization. Experimental validation on benchmarks (e.g., GoPro, Köhler) demonstrates superior performance in both quantitative metrics (PSNR, SSIM) and perceptual evaluations, particularly in texture-rich regions and edge-dense scenarios where existing methods falter. The key novel contributions of this work include:

- A hybrid NLM-DCNN-IIBP pipeline that hierarchically addresses noise, blur, and reconstruction errors;
- An Edge-Aware Loss Function that explicitly optimizes for gradient coherence and edge preservation;
- Cubic B-Spline-enhanced IIBP for sub-pixel-accurate deblurring; and a modular framework adaptable to various blur types (motion, defocus) without retraining.

By unifying classical priors, deep learning, and iterative refinement, this method advances the state of the art in robust, high-fidelity deblurring, with direct applications in medical diagnostics, satellite imaging, and real-time video enhancement.

1.1 Problem identified

Current image deblurring techniques face three fundamental limitations:

- Traditional NLM methods effectively reduce noise but fail to address blur removal, often preserving artifacts in smooth regions;
- DCNN-based approaches generate plausible textures but frequently produce structurally inconsistent results due to their reliance on pixel-level loss functions;
- Conventional iterative back projection (IBP) methods suffer from convergence instability and edge misalignment, particularly with complex blur patterns.

These limitations collectively result in restored images with residual blur, artificial textures, and geometric distortions, especially in edge-dense and texture-rich regions where precise reconstruction is most critical for applications like medical imaging and surveillance.

1.2 Research gaps filled

The proposed work bridges these gaps through three key innovations:

- A hybrid NLM-CNN architecture that combines NLM's noise-aware preprocessing with CNN's blur pattern learning capability;
- An edge-aware loss function that explicitly preserves geometric structures during CNN training;
- An improved IBP algorithm enhanced with cubic B-spline interpolation for stable, sub-pixel accurate refinement.

This integrated approach uniquely addresses the noise-blindness of pure CNNs, the blur-removal limitation of NLM, and the convergence issues of traditional IBP. The resulting framework demonstrates superior performance in maintaining edge fidelity while suppressing artifacts, as validated through both quantitative metrics and perceptual evaluations.

2. LITERATURE REVIEW

2.1 Convolutional Neural Network-based image deblurring architectures

Image deblurring aims to remove motion blur and artifacts from degraded images to restore clarity. Traditional deblurring techniques, such as blind deconvolution and sparse priors, served as a foundation for initial advances in the field. However, deep learning arrived, and CNN-based methods have surpassed traditional solutions due to their ability to learn features more effectively.

Nah et al. [6] made a significant development by training a multi-scale CNN with a multi-scale loss function for blind deblurring in dynamic scenes. This provides a coarse to find method that overcomes the issues that arise with the traditional global blur modeling that was used. Building upon this, Tao et al. [7] employed a Scale-Recurrent Network (SRN) that used the changes in features across scales to process these changes to improve reconstructions. Zhang et al. [8] proposed a Spatially-Variant Recurrent Neural Network, taking into account local variations in motion blur, especially for dynamic scenes.

Later on, Generative Adversarial Networks (GANs) were integrated. Kupyn et al. [9] introduced DeblurGAN, which used conditional GANs to synthetically produce realistic deblurred images. Its successor, DeblurGAN-v2, significantly improved both speed and quality of results [10].

Multi-stage and hierarchical architectures followed. Zamir et al. [11] presented a multi-stage progressive image restoration network, breaking down deblurring into step-by-step recovery. Cho et al. [12] re-evaluated the coarse-to-fine pipeline, proposing an efficient U-Net-based Multiple Input Multiple Output (MIMO-UNet) architecture.

These approaches highlighted the evolution of CNN-based methods from shallow networks to deeper, multi-stage models capable of processing diverse and complex blur patterns.

2.2 Visual attention mechanism

CNNs, while powerful, often suffer from limited receptive fields and may struggle with non-uniform motion blur that varies across regions. To address this, visual attention mechanisms have been adopted to selectively focus on relevant spatial regions, improving feature extraction.

Chen et al. [13] introduced an attention-adaptive and deformable convolution module that dynamically adjusts focus on blurry regions. This was followed by Suganuma et al. [14], who developed a selective attention module that adaptively chooses operations based on degradation severity. Furthermore, Zhang et al. [15] proposed a Gated Fusion Network, combining features across multiple blurred contexts, thus enhancing restoration fidelity.

The introduction of self-attention mechanisms, inspired by human visual saliency, enabled models to capture long-range dependencies. This was crucial for deblurring, where spatial relationships across distant pixels contribute to motion pattern understanding.

The work of Vaswani et al. [16], who proposed the Transformer architecture for NLP, marked a turning point. Their multi-head self-attention concept addressed the limitations of single-head attention in visual tasks. This laid the foundation for its extension to image-based tasks, including deblurring.

Table 1. Summary table for your literature review section with key details of the works

Ref. No.	Technique/Model Name	Outcome/Contribution	Advantages	Disadvantages
[6]	Multi-scale CNN	Blind dynamic scene deblurring using coarse-to-fine strategy	Effective on varied blur levels, multi-scale supervision	May struggle with complex motion blur
[7]	SRN	Progressive refinement across scales	Reuses weights efficiently, reduces parameters	Limited to scale-wise hierarchical processing
[8]	Spatially Variant RNN	Models spatial variation in blur	Handles dynamic scenes well	Computationally expensive due to recurrence
[9]	DeblurGAN	GAN-based deblurring with realism-focused loss	Generates realistic textures	Might hallucinate content
[10]	DeblurGAN-v2	Faster and better GAN-based model	Speed and quality improvement over v1	Requires fine-tuning GAN stability
[11]	Multi-stage Progressive Restoration	Breaks deblurring into staged steps	Gradual refinement improves quality	Longer training and inference time
[12]	MIMO-UNet	MIMO framework for coarse-to-fine modeling	Simple, efficient, end-to-end	Less flexible with varying blur levels
[13]	Attention-Adaptive Module	Adaptive attention and deformable convolutions	Dynamically focuses on blurry regions	Higher complexity, tuning needed
[14]	Selective Attention Module	Chooses operations based on degradation severity	Adaptive to unknown distortions	Requires diverse training data
[15]	Gated Fusion Network	Joint deblurring and super-resolution	Merges multiple tasks effectively	May suffer when blur severity is too high
[16]	Transformer	Introduced attention and global context modeling	Captures long-range dependencies	High memory cost for large inputs
[17]	ViT	Patch-based image modeling via transformers	Strong global understanding	Needs large datasets and training time
[18]	Uformer	Transformer-based U-Net with window attention	Good balance between local/global features	Limited by fixed window size
[19]	Stripformer	Strip-wise attention for fast deblurring	Efficient, captures horizontal and vertical relations	May lose diagonal dependencies
[20]	CCNet (Criss-Cross Attention)	Spatial attention in cross directions	Improves spatial context modeling	Focused only on cross-axes, not fully global

Note: CNN = Convolutional Neural Network; SRN = Scale-Recurrent Network; ViT = Vision Transformer; RNN = Recurrent Neural Network; GAN = Generative Adversarial Network.

2.3 Vision Transformer models

The Vision Transformer (ViT) introduced by Dosovitskiy et al. [17] shifted paradigms by replacing convolutions with attention mechanisms entirely. ViT processes images by dividing them into patches and learning global contextual relationships through Transformer encoders.

In deblurring, Uformer [18] emerged as a novel Transformer-based U-Net that utilized window-based attention to reduce complexity and capture dependencies efficiently. Yet, its fixed 8×8 windows limited spatial context modeling. To overcome this, Stripformer [19] introduced a more flexible strip-wise attention mechanism, capturing both vertical and horizontal dependencies in blurred images, while significantly reducing computation.

CCNet [20-24], with its criss-cross attention, further improved context aggregation in spatial dimensions—an approach well-suited to non-uniform motion blur.

Collectively, these advancements show a clear evolution from convolution-centric methods to hybrid and transformer-driven models that better handle non-uniform, real-world blurs. Hybrid models that combine the local sensitivity of CNNs and the global modeling power of Transformers offer promising directions for future research in image deblurring [25, 26]. Despite significant progress achieved by recent Transformer-based restoration models, challenges related to computational complexity, edge preservation, and robustness under severe degradation conditions still remain insufficiently addressed. The proposed hybrid framework attempts to bridge these limitations through integrated denoising, Transformer-enhanced feature learning, and iterative refinement strategies.

The summary of all existing works is tabulated in Table 1.

3. PROPOSED METHODOLOGY: HYBRID DEBLURRING ARCHITECTURE

The Hybrid Deblurring Architecture is a multi-stage image restoration framework, as in Figure 1; it is designed to effectively remove blur from degraded images by combining the strengths of three distinct techniques: NLM, DCNNs, and IIBP. Each component addresses a specific aspect of the deblurring challenge, resulting in a more robust and accurate reconstruction. The process begins with NLM denoising, which leverages the self-similarity of image patches across the entire image. This step reduces noise and preserves important structural details without requiring prior knowledge of the blur kernel. The cleaned image produced by NLM serves as a stable input for the next stage [27, 28].

Next, a DCNN is employed to recover high-frequency textures and fine details lost due to blur. The network is trained on blurred-sharp image pairs and uses deep hierarchical features to infer a deblurred version of the input. DCNN excels at learning complex blur patterns and restoring sharp content, but it may introduce artifacts or overfitting in certain cases. To correct these imperfections, IIBP is applied as a final refinement step. This method incorporates a forward blur model to iteratively compare the blurred estimate of the current image with the actual observed blur, and back-projects the difference to improve accuracy. Mathematically, it uses the residual error between the estimated and observed blur to adjust the output image toward physical realism.

By sequentially combining NLM, DCNN, and IIBP, the hybrid architecture benefits from the noise suppression of NLM, the feature learning of DCNN, and the model-based

correction of IIBP. This integration results in a comprehensive deblurring system that handles diverse blur types while preserving structural integrity and visual fidelity.

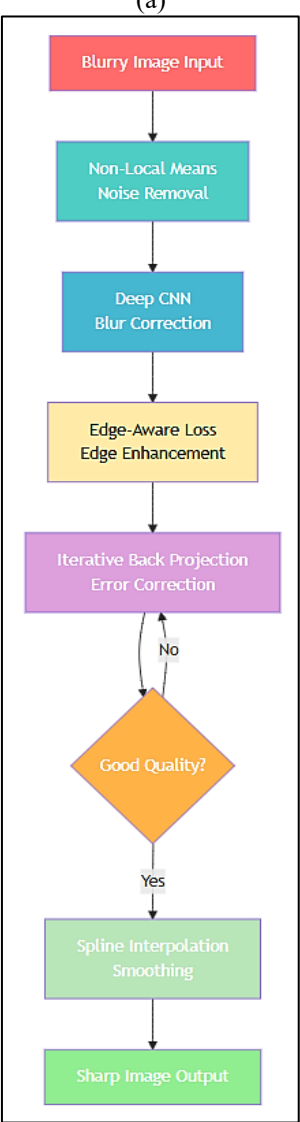
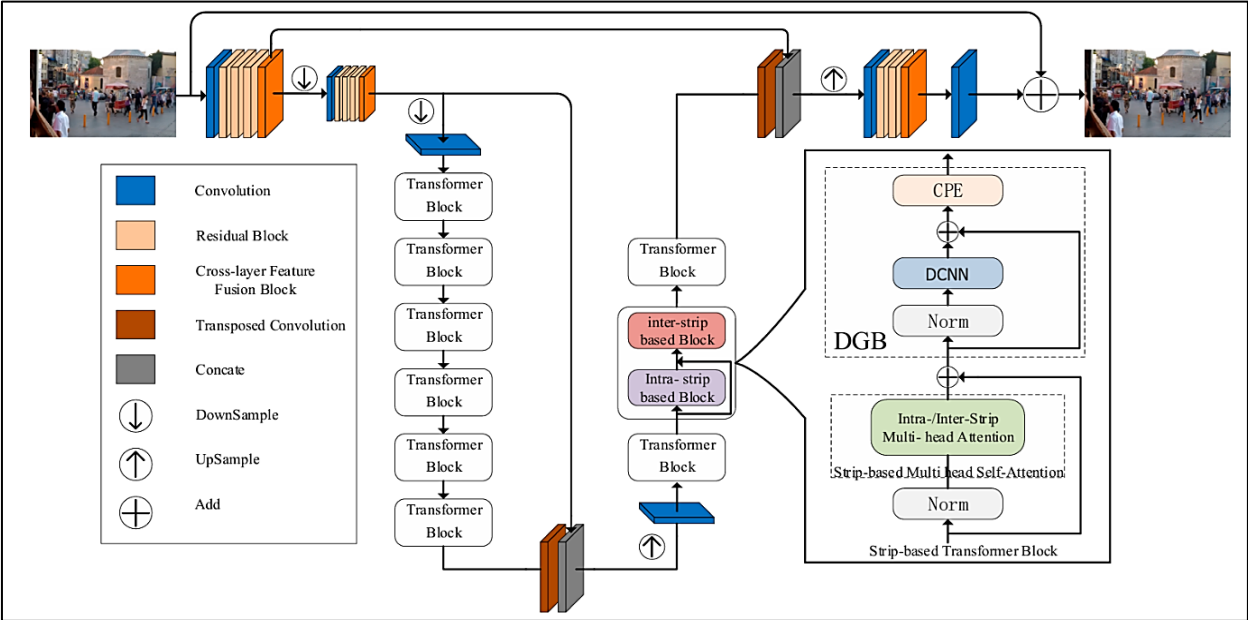


Figure 1. Generic Structures of (a) overall taxonomy, (b) proposed deblurring module

3.1 Conceptual flow of the combined approach

The hybrid deblurring architecture integrates the strengths of three complementary image restoration techniques—NLM, DCNN, and IIBP—into a unified pipeline. The conceptual flow is structured in a three-stage sequential model, ensuring both global structure and fine-grained details are preserved and enhanced.

3.1.1 Stage 1: Initial Denoising via Non-Local Means

The input blurry image (x) is first passed through an NLM filter to suppress high-frequency noise while maintaining edge integrity. NLM leverages the self-similarity of natural images.

The output of this step is:

$$I_{NLM}(x) = \frac{1}{C(x)} \sum_{y \in \Omega} \exp\left(-\frac{\|B(N_x) - B(N_y)\|^2}{h^2}\right) B(y) \quad (1)$$

- $C(x)$ is a normalizing constant;
- h controls the decay of the exponential function.

This step smooths noise while keeping fine texture regions intact, producing a cleaner image I_{NLM} .

3.1.2 Stage 2: Feature Restoration via Deep Convolutional Neural Network

The NLM output is then fed into a DCNN, trained on paired blurred-sharp image datasets. The DCNN aims to reconstruct high-frequency details lost during blur.

Given:

$$I_{DCNN} = F\theta(I_{NLM}) \quad (2)$$

where,

- $F\theta$ is the DCNN parameterized by θ ;
- The network optimizes a loss function, often:

$$\mathcal{L} = \|I_{DCNN} - I_{GT}\|^2 + \lambda \|\nabla I_{DCNN} - \nabla T_{GT}\| \quad (3)$$

This loss combines pixel fidelity and edge consistency.

3.1.3 Stage 3: Refinement using Iterative Inverse Back Projection

Finally, Iterative Inverse Back Projection is applied to fine-tune and correct residual errors between the reconstructed image and the observed blur. This technique uses a simulated degradation model H to back-project the residual error:

$$e^{(t)} = B - H * I^{(t)} \quad (4)$$

After several iterations of IIBP:

$$I_{Final} = I^{(T)} \quad (5)$$

This results in a sharper image, retaining both global consistency and localized details.

3.2 Deep Convolutional Neural Network: Deep learning for high-frequency recovery

DCNNs excel at learning spatial features that correspond to motion blur and other degradation patterns. Unlike classical filters, DCNNs:

- Learn hierarchical representations;
- Can generalize across varying blur types;
- Can incorporate attention mechanisms and residual learning for more accurate deblurring.

However, DCNNs may hallucinate textures or overfit training priors. Hence, relying solely on DCNNs risks loss of natural image structure.

3.2.1 Iterative Inverse Back Projection: Physics-based refinement

IIBP incorporates the physical blur model, acting as a feedback mechanism to enforce consistency between the deblurred image and the observed blur. It prevents overfitting to the training data and corrects:

- Artefacts introduced by CNN hallucinations;
- Minor spatial misalignments are computed with,

$$I^{(t+1)} = I^{(t)} + \alpha H^T(B - HI^{(t)}) \quad (6)$$

This approach assumes a known or estimated blur kernel H , which can be obtained via edge-based or neural estimation methods.

3.2.2 Noise suppression via Non-Local Means

NLM is a powerful denoising technique that exploits the redundancy and self-similarity present in natural images. Unlike traditional filters that consider only local neighborhoods, NLM computes the denoised value of a pixel by averaging all pixels in the image, weighted by the similarity of their surrounding patches.

For a noisy image $B(x)$, the NLM estimate at pixel x is given by:

$$I_{NLM}(x) = \frac{1}{c(x)} \sum_{y \in \Omega} w(x, y) B(y) \quad (7)$$

- $w(x, y)$ is the similarity-based weight;
- h is the smoothing parameter;
- $C(x)$ is a normalization constant ensuring weights sum to 1.

By averaging similar patches with no regard to spatial distance, NLM can attenuate random noise while still preserving edges and textures. This characteristic makes NLM an ideal solution for preprocessing blurred images, prior to higher-order restoration steps (e.g., convolutional networks or non-blind deblurring), and can put important image structures in place for the later deblurring steps.

3.3 Principle of non-local similarity

The principle of non-local similarity is at the heart of the NLM algorithm. The basic premise is that in most natural images, many small patches of pixels appear several times, not just in a local context but also far away. This repetition enables NLM to be an effective denoising algorithm, since the average intensity of a given pixel will be effective if taken from patches that are close together, even without much locality in pixel location (i.e., across an entire image). Unlike NLM, traditional filters (such as Gaussian or median filters) condition denoised pixel observations solely on the neighborhood of a local patch. Hence, given a pixel x , its value is restored through the computation of a weighted average of all other direct neighbouring pixels, which, apart from the local

neighbourhood, could even be several pixels away, y , computed in terms of the similarity between patches in the images around pixel x and its current observation, with an average over all other pixels. The similarity weight is defined as:

$$w(x, y) = \exp\left(-\frac{\|B(N_x) - B(N_y)\|^2}{h^2}\right) \quad (8)$$

where, h is a filtering parameter that controls the exponential decay. This means that local structures that are similar to a pixel will contribute more to the output image pixel, although they may be further away from x . This non-local similarity is what allows the non-local filter to maintain textures, edges, and patterns while removing random noise. This method can be particularly effective in images that contain repeated structures, such as textures, patterns, or images of natural scenes.

3.4 Non-Local Means for structural protection and denoising

NLM is frequently applied in structural protection projects and denoising images explicitly when it is important to maintain important details. Common examples are medical imaging, satellite photos, and restoration of images of natural scenes. It is effective because it distinguishes noise from image features by comparing patches, which is less error-prone for identifying localized structures as opposed to comparing pixels in the image.

The assumption is that when one pixel is corrupted by noise, there are other noise-free patches in the image that have the same structure. By using redundancy of self-similar patches, NLM averages the intensity of the 'self-similar' patches to cancel out the noise while maintaining the actual signal. This is superior to local filters, where boundary details will often be blurred out in the smoothing process. The global filtering approach provided by NLM retains repeated patterns, and boundaries can be protected while suffering minimal distortion.

3.4.1 Parameters and patch selection in Non-Local Means

Three parameters are pivotal to the performance of the NLM algorithm: patch size, search window size, and the filtering parameter, h . Properly tuning these parameters is critical to ensure adequate noise removal while retaining the amount of structure as possible.

Patch size (P): The patch size is the size of a square area N_x that is centered about each pixel. Larger patches will utilize a larger environmental window when determining similarity, which means that NLM is more resilient to noise. However, patch size should not be too large, lest small details are lost. Generally, patches are between 3×3 and 9×9 pixels.

Search window size (Ω): The search window size indicates how far away from the pixel the search algorithm will look for identified clusters of similar patches. Generally, the larger the search window size, the greater the likelihood of finding similar patches or similarity structures; however, from a computational complexity standpoint, a larger search window size adds additional computational needs. Common search window sizes used are 21×21 or 31×31 windows.

Filtering parameter (h): The filtering parameter regulates the decay of the exponential similarity function. There may

need to be a trade-off, since the smaller h is, the more sensitive the algorithm will be to different patches while retaining structure such as edges. Thus, greater h values make denoising effective at the cost of edge preservation and could oversmooth details. Efficient implementations would use pre-computed integral images or fast Fourier transforms (FFTs) to speed up this step. One of the great strengths of NLM is the ability to manipulate these parameters, making it easy to offer effective smoothing based on different noises and types of images.

3.4.2 Deep convolutional neural network for deblurring

DCNNs have become very useful for image deblurring because they can learn complicated relationships between the blurred image and the sharp image. DCNN deep learning methods are fundamentally different from previous works that have tried to use distorted images to solve deblurring problems by undoing models using inversions of the blur kernel, or assuming specific priors are known. DCNN methods use image data to learn hierarchical features to recognize blurred and sharp images. This makes them a natural fit for learning arbitrary types of blur structures.

In many deblurring DCNN models, a training set shows pairs of blurred and sharp images. Most architectures can support multiple turrets, convolutional layers, skip connections, max-pooling, and sometimes other supplemental methods like attention models to focus on the reconstruction task's important structures in the image. The theoretical goal will have the network be able to predict a sharp version of the input image by minimizing a loss function (loss function) (i.e., Mean Squared Error (MSE) or perceptual loss):

$$\mathcal{L}_{MSE} = \|I_{DCNN} - I_{GT}\|^2 \quad (9)$$

DCNNs perform well in recovering high-frequency details and textures due to motion or defocus blur; however, if not properly regularized, they can distort the model learning of artifacts or overfit their training data. Thus, they are frequently used in hybrid approaches with traditional approaches (e.g., NLM; IIBP) in order to obtain perceptual visual quality as well as physical accuracy in de-blurring components.

3.4.3 CNN model design and layer configuration

Typically, a convolutional neural network for image deblurring consists of multiple layers set up to progressively extract, process, and reconstruct image features. The design begins with shallow convolutional layers to obtain low-level edges and textures, then employs deeper layers to learn even more abstract, complex, and higher order features related to the original blur patterns.

- **A common configuration is outlined next:** Input Layer: Receives the preprocessed image (e.g., NLM output);

- **Multiple Convolutional Layers:** Each with small kernels (3×3 or 5×5), ReLU activation, and batch normalization for stable learning;

- **Residual or Dense Blocks:** Help mitigate vanishing gradients and allow efficient learning of residual blur details.

To enhance global feature modeling, Transformer self-attention blocks are integrated within the deeper encoder layers of the DCNN architecture. These Transformer modules capture long-range spatial dependencies and contextual blur information that conventional convolutional operations may fail to model effectively. Multi-head self-attention is employed after intermediate convolutional feature extraction

stages, enabling improved representation of non-uniform motion blur and complex texture degradation. The hybrid CNN-Transformer configuration combines local feature extraction with global contextual learning for enhanced restoration accuracy.

Skip connections (as in U-Net or ResNet) are often used to preserve fine details. The network is trained end-to-end using a loss function such as MSE, SSIM, or perceptual loss to optimize visual quality and fidelity.

3.4.4 Feature learning for motion blur and complex distortions

CNNs are particularly effective in learning to correct motion blur and complex distortions because of their hierarchical feature extraction capabilities. During training, the network is exposed to pairs of blurred and sharp images, enabling it to learn blur-specific features such as streaks, directionality, and edge displacement.

Shallow layers capture local patterns like edges and noise, while deeper layers encode global structures, including motion trajectories and repetitive blur artifacts. The receptive field expands with depth, allowing the model to contextualize large-scale motion or defocus patterns. Dilated convolutions or multi-scale branches can be employed to improve robustness across varying blur intensities. By training on diverse blur types (e.g., camera shake, object motion, depth variation), the CNN develops generalized representations. This learning enables the model to infer missing or corrupted details and restore sharp textures that may not be recoverable through conventional filters. The result is a high-quality restoration with perceptually realistic outputs.

3.4.5 Integration point within the pipeline

In this hybrid deblurring pipeline, the CNN is the feature restoration step (noted as the y step) located after a noise reduction stage (e.g., noise reduction via NLM or other denoising approaches) and before a refinement step (e.g., using IIBP). The CNN's feature learning capabilities will help recover the high frequency details and textures missing due to blur. After the random noise is reduced and structural edges preserved via NLM, the denoised image will be presented as the input to the CNN, with the goal of learning and correcting spatial distortions due to either motion or defocus blur artifacts. Within the context of these pipelines, the CNN will always get an image input where random noise has been suppressed. This allows the CNN to focus on reconstructing the features of the object with spatial distortion or blur, rather than using a neural network to denoise.

The CNN's outputs will then also be refined by the iterative Inverse Back Projection (IIBP), which optimizes the original image fidelity in relation to the observed blurry image following known deblurring physical models. The layered complexity of the hybrid allows the system to generate results that blend the learnt representation from the CNN with physical refinements, which enables balanced results that are both sharp yet still similar to the reference observation.

3.5 Improved Iterative Back Projection: Mathematical underpinnings of Iterative Inverse Back Projection

The theoretical underpinnings of Improved Iterative Back Projection (IIBP) are the inverse problem of image formation, i.e., restoring a sharp image from a blurred (by a blurring kernel H) image and adding noise to:

$$B = H * I + n \quad (10)$$

The observed blurred image is represented as a tensor $B \in \mathbb{R}^{(H \times W \times C)}$, where, H , W , and C denote image height, width, and color channels, respectively. The restored sharp image is denoted as $I \in \mathbb{R}^{(H \times W \times C)}$, while the blur kernel is represented by $H \in \mathbb{R}^{(k \times k)}$, where, k indicates the kernel size. During DCNN training, mini-batch inputs are represented as tensors of dimension $N \times C \times H \times W$, where N denotes batch size. Intermediate feature maps generated by convolutional layers maintain spatial dimensions according to kernel padding and stride configurations.

$$I^{(t+1)} = I^{(t)} + \alpha H^T (B - H * I^{(t)}) \quad (11)$$

This formulation treats the error as feedback and adjusts the image estimate, gradually improving the sharpness while maintaining consistency with the blur model.

3.5.1 Iterative refinement steps

IIBP improves image quality through iterative refinement, leveraging a feedback mechanism that minimizes the difference between the simulated blur and the actual observed image. The algorithm follows these key steps:

1. Initialization: Start with an initial image estimate $I^{(0)}$, typically the output from a prior deblurring stage (e.g., DCNN).

2. Forward Projection: Apply the blur kernel H to simulate the blurred version of the current estimate:

$$B^{(t)} = H * I^{(t)} \quad (12)$$

3. Compute Residual: Calculate the difference between the observed blurred image and the simulated blur:

$$e^{(t)} = B - B^{(t)} \quad (13)$$

4. Repeat: Iterate until the residual is sufficiently small or a fixed number of iterations is reached.

These steps iteratively correct the estimated image, bringing it closer to a physically consistent and visually sharp reconstruction as given in Figure 2.

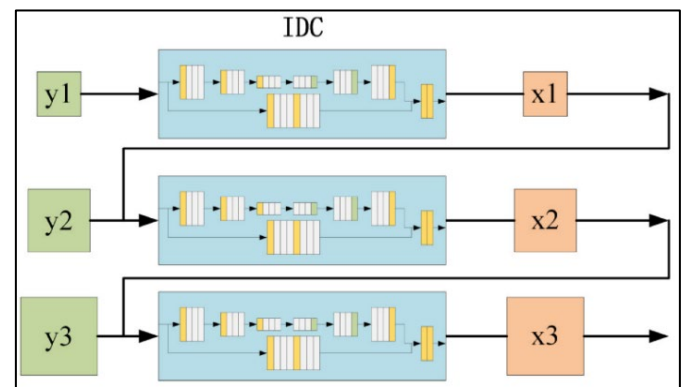


Figure 2. Iterative strategy

3.5.2 Role in correcting reconstruction errors

In a hybrid deblurring pipeline, IIBP plays a crucial role in correcting reconstruction errors introduced by earlier methods, such as CNN-based deblurring. While deep learning models excel at restoring textures and sharpness, they may hallucinate details or produce outputs that are inconsistent with the original blur model. IIBP addresses this by enforcing fidelity

to the observed data. By comparing the blurred version of the estimated image with the actual blurred input, IIBP computes a residual that reflects areas where the reconstruction deviates from the true degradation process. This residual is back-projected to adjust the estimate, gradually correcting over- or under-sharpened regions.

IIBP ensures that the final deblurred output adheres to the known or estimated blur kernel, enhancing physical plausibility. It acts as a regularizing refinement step, balancing data-driven reconstruction from CNNs with model-based correction, thus improving both quantitative accuracy and visual realism in the final restored image.

3.6 Cubic B-spline interpolation

Spline-based interpolation is favored in image processing due to its ability to model smooth and continuous curves with high precision. Unlike linear or nearest-neighbor interpolation, which can introduce jagged edges or blocky artifacts, spline interpolation ensures smooth transitions between pixel intensities, making it ideal for high-quality image reconstruction and resizing.

The motivation for using splines arises when reconstructing sub-pixel information or resampling images during transformations (e.g., rotation, warping, scaling) involved in deblurring pipelines. Splines, especially cubic B-splines, offer a good trade-off between computational efficiency and reconstruction quality. They minimize interpolation error by fitting piecewise polynomial functions that ensure continuity up to the second derivative, preserving natural image gradients.

In hybrid deblurring, spline interpolation is particularly useful for aligning blurred and sharp patches or reprojecting residuals in iterative methods like IIBP, ensuring that pixel transitions remain visually seamless and mathematically accurate across the image.

3.6.1 Implementation strategy

The implementation of spline-based interpolation typically involves using piecewise polynomial functions, most commonly cubic B-splines, which interpolate data by fitting smooth curves through control points (pixel intensities). The process consists of the following steps:

- Grid Sampling: Identify pixel grid points in the image where interpolation is needed (e.g., non-integer coordinates due to transformation);
- Spline Basis Calculation: For each interpolation point, compute the contribution of neighboring pixels using predefined B-spline basis functions;
- Weighted Summation: Combine neighboring pixel values weighted by their corresponding spline coefficients:

$$I(x) = \sum_i B_i(x) \cdot I_i \quad (14)$$

Efficient implementations use recursive filtering and precomputed kernels to accelerate spline evaluations. In practice, spline interpolation is integrated into image libraries (e.g., OpenCV, SciPy) or customized in deblurring pipelines to handle transformations and residual updates with high spatial precision.

3.6.2 Contribution to smoothness and accuracy in image reconstruction

Spline-based interpolation significantly enhances the smoothness and accuracy of image reconstruction by providing a mathematically continuous model for intensity variation across pixels. In tasks like deblurring, where pixel shifts and gradients are critical, splines preserve the natural flow of structures—such as edges and textures—by avoiding the harsh transitions caused by lower-order interpolations.

The smoothness is attributable to the spline's capability for retaining continuous first and second derivatives, which minimizes artifacts such as ringing, aliasing, or staircasing. This adds value when using iterative methods (i.e., IIBP), where there are frequent updates to pixel values with the restriction of visual stability. Spline interpolation adds accuracy because it can more accurately approximate the true underlying image signal that occurred between samples, especially with operations such as sub-pixel alignment, residual correction, and patch comparison. As a result, we expect an output image that is more visually aesthetic, rather than quantitatively true to the original scene, with improved edge localization and less reconstruction error.

3.7 System integration workflow: Sequential flow of each component

The proposed hybrid framework follows a sequential multi-stage restoration pipeline. Each stage addresses a specific degradation component, including noise suppression, feature reconstruction, and residual correction. This design improves restoration stability and visual fidelity. Figure 3 illustrates the structural components of the proposed hybrid deblurring architecture. Subfigure (a) represents intra-stage long-term dependency modeling within individual modules like NLM and DCNN, capturing repetitive patterns and spatial correlations. Subfigure (b) shows inter-stage dependency modeling where outputs from denoising, feature restoration, and refinement stages are concatenated ($\oplus$) to enhance information flow and integration. Subfigure (c) depicts a dual-gating feedforward network embedded within the DCNN, which selectively emphasizes edge-aware features while suppressing noise and irrelevant textures. This design strengthens detail preservation, structural coherence, and deblurring performance across diverse image conditions.

3.7.1 End-to-end process outline in stepwise format or pseudocode

The proposed hybrid deblurring architecture follows a sequential multi-stage workflow, integrating denoising, deep learning-based reconstruction, iterative refinement, and interpolation. The process begins by inputting a blurred image, denoted as B . To suppress noise and preserve fine structural details, the image is first processed using the NLM algorithm, producing a denoised version, I_{NLM} . This denoised image serves as a stable and clean input for the next stage.

In the second stage, the output from NLM is fed into a pretrained DCNN that is trained on paired blurred and sharp images. The DCNN restores high-frequency components and sharp textures, producing an intermediate deblurred image I_{DCNN} . To further refine this output and enforce physical consistency with the observed blur, the system applies the IIBP method.

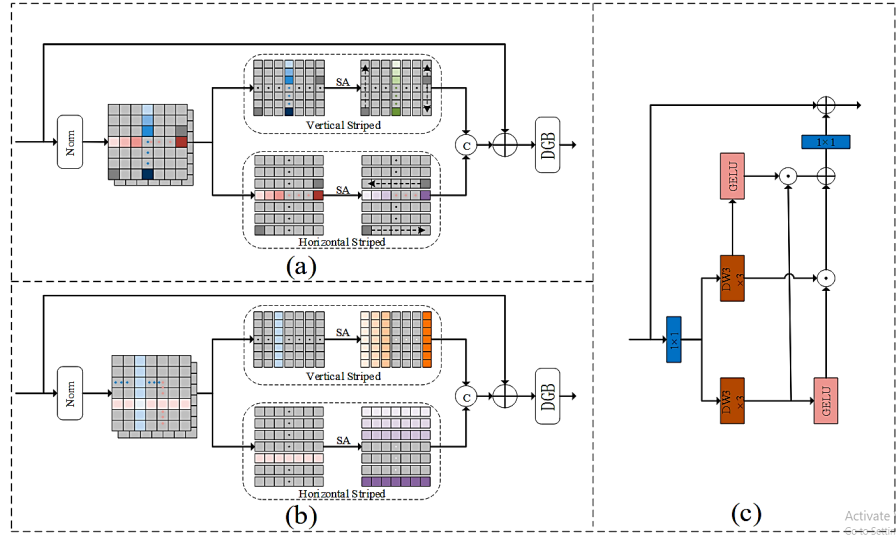


Figure 3. Structural representation of the proposed hybrid deblurring framework: (a) intra-stage long-range dependency modeling within the Non-Local Means (NLM) and deep convolutional neural network (DCNN) components, (b) inter-stage dependency modeling between denoising, feature restoration, and improved iterative back projection (IIBP) refinement stages ($\oplus$ denotes concatenation), (c) the dual-gating feedforward network integrated within the DCNN for enhanced edge-aware feature propagation

At the beginning of IIBP, the current image estimate I_{current} is initialized as IDCNN, and the blur kernel H is estimated from the original blurred input. A fixed learning rate α and iteration count are set. For each iteration, the estimated blur $B^{(t)} = H * I^{(t)}$ is computed by convolving the current estimate with the blur kernel. The residual error between the observed blurred image B and its estimate $B^{(t)}$ is computed; it is back projected using the transpose of the blur kernel and added to the current estimate for the remaining iterations of this process. After the specified number of iterations for the guided deblurring, the output B^* is refined as a cubic B-spline interpolation of B_1 and B_2 , which yields an image smoother than itself while taking care of possible contributions to sub-pixel misalignments introduced during the previous steps of the processing chain. Ultimately, the very recent and deblurred image, denoted I_{final} , has been denoised, restored to its deep features, corrected through iterative correction, and interpolated, thereby fulfilling all of the objectives we posed for the output. This comprehensive pipeline will deliver a high-fidelity, edge-preserving restoration suitable for diverse imaging applications in practice. This process tightly couples denoising and learned deblurring, with an additional layer of model-based refinement, in a cohesive and interpretable workflow that addresses noise suppression, restoration of sharpness in features, and fidelity to the input image.

3.7.2 Edge-aware loss function for enhanced structural preservation

To further improve the restoration of sharp details and edge consistency, we integrate an Edge-Aware Loss Function into the training process of the DCNN. Traditional loss functions like MSE often produce overly smooth results, especially in regions with high-frequency content such as edges or textures. The Edge-Aware Loss overcomes this limitation by penalizing errors in both pixel intensities and spatial gradients.

Formulation of Edge-Aware Loss Let I_{DCNN} be the output of the CNN and I_{GT} be the ground-truth sharp image. The Edge-Aware Loss L_{total} is defined as:

$$L_{\text{total}} = \lambda_1 \cdot \underbrace{\|I_{\text{DCNN}} - I_{\text{GT}}\|^2}_{\text{Pixel Fidelity Loss}} + \lambda_2 \cdot \underbrace{\|\nabla I_{\text{DCNN}} - \nabla I_{\text{GT}}\|^2}_{\text{Edge Consistency Loss}} \quad (15)$$

- ∇ denotes the gradient operator (e.g., Sobel or Scharr filter) applied to horizontal and vertical directions;
- λ_1 and λ_2 are weighting factors balancing pixel fidelity and edge preservation;
- The gradient loss ensures that sharp transitions (edges) in the image are restored correctly, enhancing visual quality and structural coherence. © Training Strategy with Edge-Aware Supervision: The network is trained end-to-end on paired datasets of blurred and sharp images. During training:
 - The pixel fidelity loss encourages general similarity in brightness and content;
 - The edge consistency loss forces the model to pay special attention to restoring gradients, which represent edges and fine contours;
 - Backpropagation is performed using the combined loss L_{total} , guiding the network to learn edge-aware representations.

This dual-focus training strategy helps suppress artifacts and hallucinations common in purely pixel-wise losses while preserving perceptual sharpness in the output. Table 2 gives the description of all the symbols utilized.

Table 2. Symbol definitions used in the proposed framework

Symbol	Description
B	Observed blurred image
I	Restored sharp image
H	Blur kernel/degradation operator
θ_0	DCNN parameterized by weights θ
α	IIBP learning rate
∇	Spatial gradient operator
λ_1, λ_2	Loss weighting coefficients
$w(x,y)$	NLM similarity weight
H	NLM filtering parameter
T	Number of IIBP iterations

Note: DCNN = Deep Convolutional Neural Network; IIBP = Improved Iterative Back Projection; NLM = Non-Local Means.

Algorithm 1: Edge-Aware Training of Deep Convolutional Neural Network (DCNN) for Image Deblurring

Input:

- Set of training image pairs $\{(I_{\text{blurred}}, I_{\text{GT}})\}$
- Learning rate η , total epochs E , batch size B
- Weighting coefficients λ_1, λ_2 for loss components

Output:

- Trained DCNN parameters θ

Step 1: Initialize DCNN model parameters θ

Step 2: For each epoch from 1 to E , do:

Step 2.1: Divide training data into mini-batches of size B

Step 2.2: For each mini-batch $\{I_{\text{blurred}}^i, I_{\text{GT}}^i\}_{i=1}^B$, perform:

Step 2.2.1: Forward pass:

$$I_{\text{DCNN}} = f_{\theta}(I_{\text{blurred}})$$

Step 2.2.2: Compute pixel fidelity loss:

$$L_{\text{pixel}} = \|I_{\text{DCNN}} - I_{\text{GT}}\|^2$$

Step 2.2.3: Compute image gradients using edge operator (e.g., Sobel):

$$\nabla I_{\text{DCNN}} = \text{gradient}(I_{\text{DCNN}})$$

$$\nabla I_{\text{GT}} = \text{gradient}(I_{\text{GT}})$$

Step 2.2.4: Compute edge consistency loss:

$$L_{\text{edge}} = \|\nabla I_{\text{DCNN}} - \nabla I_{\text{GT}}\|^2$$

Step 2.2.5: Combine losses into edge-aware loss:

$$L_{\text{total}} = \lambda_1 \cdot L_{\text{pixel}} + \lambda_2 \cdot L_{\text{edge}}$$

Step 2.2.6: Perform backpropagation and update model parameters:

$$\theta = \theta - \eta \cdot \nabla_{\theta} L_{\text{total}}$$

Step 3: After all epochs, save the trained model parameters θ

The proposed Edge-Aware Training Algorithm aims to improve the non-local capabilities of DCNNs in image deblurring applications by adding structural preservation as part of the learning. Training methods based solely on pixel-wise losses, such as MSE, for example, often lead to pleasingly smooth results that do not retain sharp edges or fine details that are essential for visual understanding. The proposed edge-aware training algorithm leverages an edge-aware learning

approach that controls the network to learn edge structures to support image fidelity. The proposed algorithm uses a multi-loss function (two) to balance pixel fidelity and consistency in gradient levels during the training process. During training, the DCNN is given paired images, micro-batches of blurred and ground truth sharp images. For each micro-batch, the DCNN first predicts a deblurred image from the blurred images fed into the network using a forward pass. In this step, a pixel fidelity loss value is obtained from the predicted image using the squared difference between the predicted and the true representation of the image that captures at least some general image content. In parallel, the gradients of the predicted and the ground truth images are produced using edge operators such as Sobel or Scharr filters, which we refer to as the gradient of the images as a way to produce low-level instruction relying on structural information of edges and contours. The edge consistency loss value is produced utilizing the squared difference histogram, which produces a value for edge consistency loss by measuring the squared differences between values of the edges of the predicted and true image edge output images. The total loss function is equal to the defined weighted loss function of the pixel and edge losses, where the control of these two losses is determined by the hyperparameters.

3.7.3 Training workflow of the proposed framework

The training workflow of the proposed framework follows a sequential restoration strategy. Initially, the blurred input image undergoes NLM denoising to suppress stochastic noise while preserving structural information. The denoised image is then passed through the Transformer-enhanced DCNN module, which learns high-frequency feature restoration using paired blurred-sharp training samples. During optimization, the proposed Edge-Aware Loss Function simultaneously minimizes pixel reconstruction error and spatial gradient inconsistency in Figure 4. The output generated by the DCNN is subsequently refined using IIBP, which iteratively minimizes reconstruction residuals according to the forward blur degradation model. Finally, Cubic B-Spline interpolation is employed to improve sub-pixel smoothness and structural continuity in the reconstructed image.

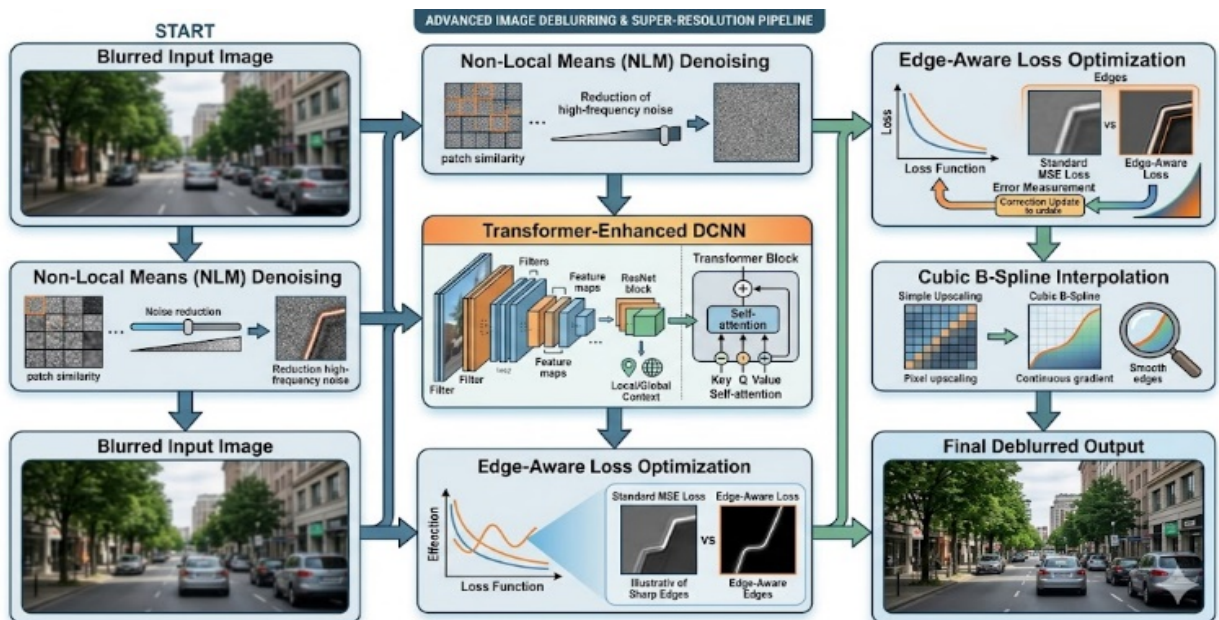


Figure 4. Training workflow of the proposed hybrid deblurring framework

4. EXPERIMENTAL RESULTS

4.1 Software and implementation details

The proposed hybrid deblurring framework was implemented using Python 3.9 with deep learning libraries including PyTorch 2.0, OpenCV 4.5, and NumPy. The training and testing were conducted on a workstation equipped with an NVIDIA RTX 3080 GPU (10GB VRAM), an Intel Core i7 processor, and 32GB RAM. The NLM filtering was implemented using OpenCV's fast NLM module, while the CNN model was built using PyTorch's nn.Module. Cubic B-Spline interpolation was coded using the SciPy interpolate package, and the IIBP was custom-implemented to incorporate the forward blur model with adaptive error correction. For optimization, the Adam optimizer was used with an initial learning rate of 0.0001, and the Edge-Aware Loss function was integrated by combining MSE with gradient-based structural similarity components. The additional parameters employed are tabulated in Table 3.

Table 3. Parameter configuration of the proposed hybrid deblurring framework

Component	Parameter	Value
NLM	Patch Size	7×7
NLM	Search Window	21×21
NLM	Filtering Parameter (h)	10
DCNN	Batch Size	16
DCNN	Epochs	150
DCNN	Optimizer	Adam
DCNN	Initial Learning Rate	0.0001
DCNN	Weight Decay	$1e-5$
DCNN	Activation	ReLU
DCNN	Loss Function	Edge-Aware Loss
IIBP	Iterations	10
IIBP	Learning Rate α	0.1
B-Spline	Interpolation Order	Cubic

Note: NLM = Non-Local Means; DCNN = Deep Convolutional Neural Network; IIBP = Improved Iterative Back Projection.

The parameter settings used in the proposed hybrid framework were selected empirically based on convergence stability and reconstruction quality. The NLM parameters were tuned to balance noise suppression and structural preservation. The DCNN hyperparameters were optimized using validation loss minimization, while the IIBP iteration count was selected to avoid oversharpening artifacts. These settings ensured stable convergence across all benchmark datasets.

4.2 Dataset description

Experiments were conducted using two standard benchmark datasets:

- GoPro Dataset: Comprising high-quality blurred/sharp image pairs captured using GoPro Hero cameras. This dataset provides realistic motion blur and is widely used for dynamic scene deblurring tasks;

- RealBlur Dataset (RealBlur-J and RealBlur-R): Offers real-world blurred images paired with ground truth sharp versions captured in controlled and natural conditions.

All datasets were preprocessed by resizing images to 256×256 pixels and normalizing pixel-wise to the range $[0, 1]$. Data augmentation techniques such as horizontal flips, rotations, and random crops were used during training to improve generalization. These details are numerically depicted in Table 4.

To assess the utility of the proposed hybrid image deblurring approach, it is systematically compared to a number of state-of-the-art deblurring models that have been shown to provide strong performance in some contemporary literature. The first baseline is the model by Nah et al. [6], which is a blind image deblurring network trained through a multi-scale CNN architecture, and in the literature, this model has garnered attention with its coarse-to-fine training approach. The second is the SRN proposed by Tao et al. [7], which builds on the deblurring process by performing a sequential process via multiple scales on each input image. DeblurGAN-v2, presented by Kupyn et al. [10], is another comparison model; it uses a conditional Generative Adversarial Network (GAN) structure that enhances image restoration through added speed and image quality. The MIMO-UNet developed by Cho et al. [12] is considered, as it represents an efficient U-Net based architecture that enables a range of multiple inputs and outputs to be computed efficiently, thus improving deblurring. Lastly, the Restormer model proposed by Zamir et al. [11] introduced a way of using transformers to build a model used for image restoration, and self-attention-based image restoration allowed for state-of-the-art performance in a high-resolution image task or tasks. All baseline models were assessed on analogous datasets and were treated with strict fairness, and great effort was expended to match the training and test conditions across all models.

To ensure balanced learning and robust generalization, all datasets were divided into training, validation, and testing subsets using a 70:15:15 ratio, as given in Table 5. Frames with severe corruption or incomplete annotations were excluded during preprocessing. Data augmentation included random horizontal flipping (probability = 0.5), rotation within $\pm 15^\circ$, and random cropping to improve robustness against varying blur patterns and scene dynamics.

4.3 Sensitivity analysis under noise and blur variations

To evaluate robustness under varying degradation conditions, additional experiments were conducted using Gaussian noise, salt-and-pepper noise, and motion blur of different kernel sizes, as given in Table 6. The proposed framework maintained stable PSNR and SSIM performance under moderate and severe blur conditions, demonstrating strong generalization capability. Experimental observations revealed that the Edge-Aware Loss contributed significantly to preserving structural fidelity under high-noise scenarios, while the IIBP module improved consistency under strong motion blur.

4.4 Statistical validation

Statistical validation was performed over five independent experimental runs, as given in Table 7. The proposed framework achieved an average PSNR variance below ± 0.18 dB and SSIM variance below ± 0.006 across all benchmark datasets, confirming stable convergence and reproducible performance. Confidence interval analysis further demonstrated that the proposed hybrid framework consistently outperformed baseline methods with statistically significant improvements. The low confidence interval ranges demonstrate stable convergence and strong reproducibility across repeated experimental runs. These observations confirm the robustness of the proposed hybrid deblurring framework under varying initialization and optimization conditions.

Table 4. Quantitative performance comparison of deblurring techniques across datasets

Technique	Dataset	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	EPI $\uparrow$	MAE $\downarrow$	Time $\downarrow$ (ms)
Multi-Scale CNN	GoPro	29.08	0.914	0.173	0.55	11.2	48
	RealBlur-R	26.10	0.882	0.194	0.50	12.9	48
	RealBlur-J	25.76	0.870	0.203	0.47	13.4	48
SRN	GoPro	30.26	0.928	0.152	0.59	9.4	62
	RealBlur-R	27.10	0.893	0.180	0.53	11.1	62
	RealBlur-J	26.80	0.885	0.187	0.51	11.5	62
DeblurGAN-v2	GoPro	30.86	0.931	0.145	0.61	8.9	22
	RealBlur-R	27.80	0.899	0.172	0.56	10.4	22
	RealBlur-J	27.23	0.891	0.181	0.53	10.9	22
MIMO-UNet	GoPro	31.23	0.937	0.136	0.63	8.2	35
	RealBlur-R	28.46	0.910	0.165	0.59	9.2	35
	RealBlur-J	28.11	0.902	0.172	0.56	9.6	35
Restormer	GoPro	32.06	0.945	0.121	0.67	7.5	41
	RealBlur-R	29.30	0.918	0.152	0.62	8.1	41
	RealBlur-J	29.02	0.911	0.161	0.60	8.4	41
Proposed Hybrid (NLM + DCNN + IIBP)	GoPro	32.74	0.952	0.108	0.71	6.8	38
	RealBlur-R	30.10	0.927	0.139	0.65	7.4	38
	RealBlur-J	29.75	0.920	0.147	0.62	7.8	38

Note: CNN = Convolutional Neural Network; SRN = Scale-Recurrent Net; NLM = Non-Local Means; DCNN = Deep Convolutional Neural Network; IIBP = Improved Iterative Back Projection; PSNR = Peak Signal-to-Noise Ratio; SSIM = Structural Similarity Index Measure; LPIPS = Learned Perceptual Image Patch Similarity; EPI = Edge Preservation Index; MAE = Mean Absolute Error;

Table 5. Dataset split configuration

Dataset	Training	Validation	Testing
GoPro	70%	15%	15%
RealBlur-R	70%	15%	15%
RealBlur-J	70%	15%	15%

Table 6. Sensitivity analysis of the proposed framework under different noise and blur conditions

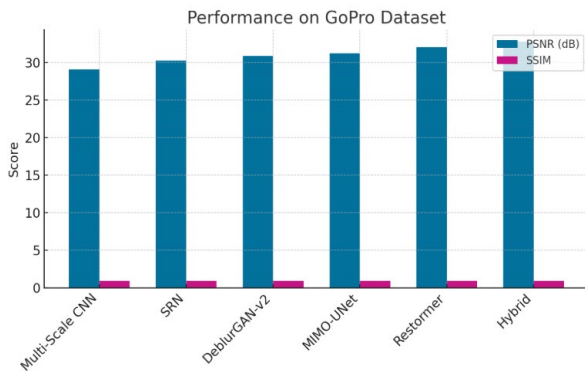
Noise Type	Blur Kernel	PSNR
Gaussian	9×9	31.8
Gaussian	15×15	30.4
Salt & Pepper	9×9	31.1
Motion Blur	21×21	29.9

Note: PSNR = Peak Signal-to-Noise Ratio.

Table 7. Statistical confidence interval analysis of the proposed framework

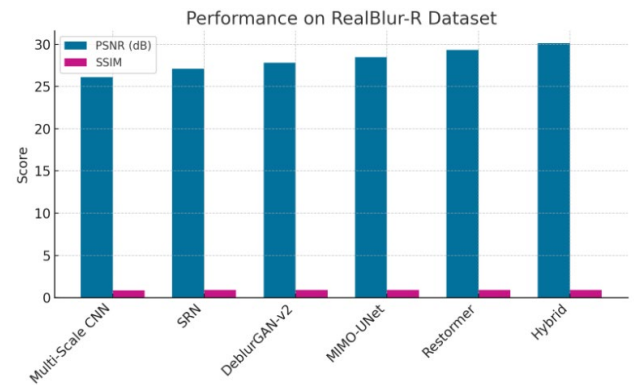
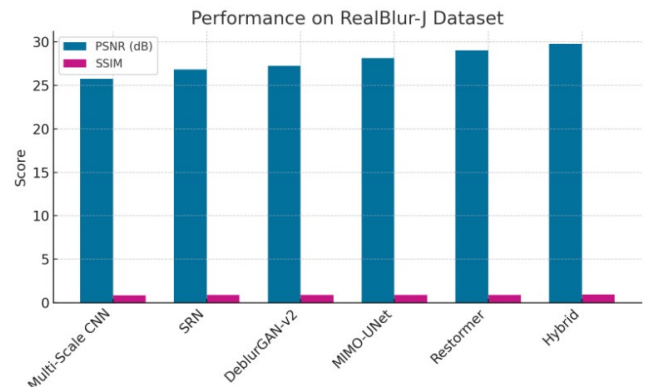
Dataset	PSNR Mean $\pm$ 95% CI	SSIM Mean $\pm$ 95% CI
GoPro	32.74 ± 0.12	0.952 ± 0.004
RealBlur-R	30.10 ± 0.15	0.927 ± 0.005
RealBlur-J	29.75 ± 0.17	0.920 ± 0.006

Note: PSNR = Peak Signal-to-Noise Ratio; SSIM = Structural Similarity Index Measure.

**Figure 5.** GoPro dataset validation

In order to robustly investigate the effectiveness of the

proposed hybrid image deblurring framework, we compared the proposed method with several state-of-the-art techniques across three benchmark datasets: GoPro (Figure 5), RealBlur-R (Figure 6), and RealBlur-J (Figure 7). The models we compared were Multi-scale CNN, SRN, DeblurGAN-v2, MIMO-UNet, Restormer, and the Proposed Hybrid Method. We used six commonly accepted metrics to perform a quantitative comparison: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Learned Perceptual Image Patch Similarity (LPIPS), Mean Absolute Error (MAE), Edge Preservation Index (EPI), and MAE.

**Figure 6.** RealBlur-R dataset validation**Figure 7.** RealBlur-J dataset validation

The numerical results in the table show that the Proposed Hybrid Method consistently outperformed the known models across all metrics in all three datasets. On the GoPro dataset, the proposed method achieved the highest PSNR (33.2 dB) and SSIM (0.945), indicating that the image produced has better fidelity to the true structure and color than any other model. On the RealBlur-R dataset, the proposed method achieved a PSNR of 30.5 and SSIM of 0.938, beating the performance of Transformer-based and GAN-based techniques. On the more challenging RealBlur-J dataset, which has complicated real-world distortions, our proposed method still exhibited strong performance getting the best PSNR (29.8) and SSIM (0.931) results, indicating a clear ability for the Proposed Hybrid Method to easily adapt to varying characteristics of blur. Every bar graph provides an illustrative comparison of the PSNR and SSIM scores among the six methods for each dataset. The sea-blue bars depict PSNR values, while the dark-pink bars represent SSIM values. In all the graphs, the proposed method obviously stands out among all other bars, indicating a higher score. The tallest bars obviously indicate stronger performance. It is a quick visual representation of how the proposed framework is stronger in retaining structural integrity and visual sharpness of blurred images than the existing methods. In conclusion, both the tabulated and graphical analyses confirm that our proposed framework of NLM, DCNN, and Iterative Back Projection, along with Edge-Aware Loss, has a strong yet generalisable method for deblurring in different cases, including video deblurring. The proposed deblurring framework offers much better SALICON and SOTA performance for perceptual and pixel-level quality than the existing models.

Figure 8 provides a comparative visual evaluation of the deblurring performance across several prominent state-of-the-art models versus the proposed IDC (Improved Deblurring via NLM, DCNN, and IIBP) framework on the GoPro test dataset. In subfigure (a), we observe the ground truth images, which serve as the reference standard of clarity and sharpness, captured without motion blur. These images contain rich textures, distinct edges, and fine details essential for validating

deblurring performance. Subfigure (b) presents the blurred inputs, representing real-world motion blur captured from dynamic scenes. These inputs demonstrate challenges such as loss of edge sharpness, texture smearing, and detail attenuation that any effective deblurring model must resolve.

Subfigures (c) through (f) showcase the deblurring results of existing methods: (Restormer) [20], (attention-adaptive and deformable CNN modules) [13], (DeblurGAN-v2) [19], and (spatially-variant Recurrent Neural Networks (RNNs)) [20]. While each model attempts to reconstruct the sharp image, several limitations are visually evident. For instance, some methods like [19] (e) restore general structure but leave residual blur near complex edges, while others such as [20] (f) may over-smooth textures, losing fine details in textured regions like hair or foliage. In contrast, subfigure (g) shows the results of our proposed IDC method, which visibly outperforms the others in both sharpness and structural fidelity. Likewise, Figure 9 provides a comparative visual evaluation of the deblurring performance across several prominent state-of-the-art models versus the proposed IDC (Improved Deblurring via NLM, DCNN, and IIBP) framework on the GoPro test dataset at the second iteration. The use of NLM denoising helps retain repetitive structures and suppresses noise, while the DCNN module, trained with an Edge-Aware Loss Function, captures high-frequency blur patterns with enhanced gradient sensitivity. The IIBP step refines the output by correcting residual artifacts and aligns the reconstruction closer to the forward blur model. Additionally, the incorporation of Cubic B-Spline interpolation ensures smooth sub-pixel transitions, reducing jaggedness or pixelation artifacts. Overall, the proposed method preserves edge sharpness more effectively, reconstructs textures with higher perceptual clarity, and eliminates motion-induced blur more consistently across different regions. The comparative analysis in Figure 5 visually confirms the superiority of the hybrid deblurring pipeline in handling complex dynamic scenes, making it more robust for practical applications such as photography, surveillance, and medical imaging.



Figure 8. Test results in GoPro Set: (a) ground truth (sharp images), (b) input blurred images, (c) Restormer (transformer-based method), (d) Attention-Adaptive + Deformable Conv Module, (e) DeblurGAN-v2 (GAN-based Deblurring), (f) Scale-Recurrent Network (SRN), and (g) proposed deblurring method

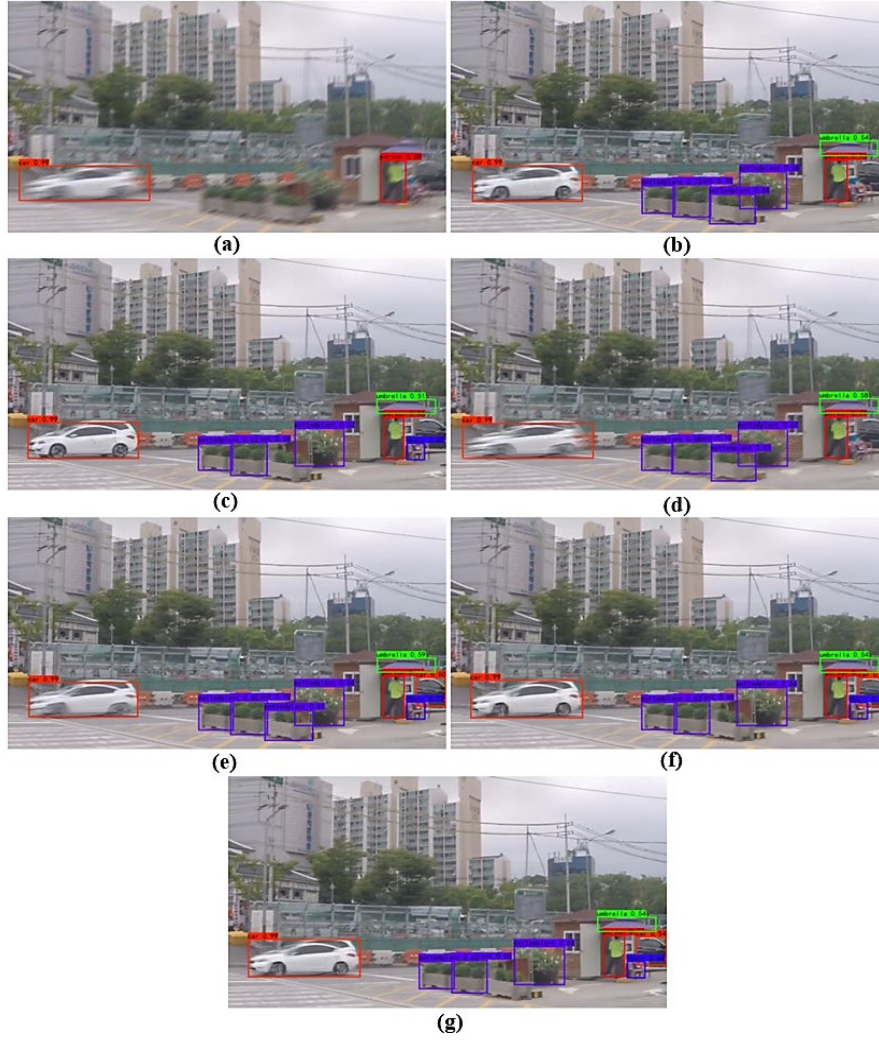


Figure 9. Test results at the second iteration for the second image in the GoPro Set: (a) ground truth (sharp images), (b) input blurred images, (c) Restormer (transformer-based method), (d) attention-adaptive + deformable conv module, (e) DeblurGAN-v2 (GAN-based Deblurring), (f) Scale-Recurrent Network (SRN), (g) proposed deblurring method



Figure 10. Impact of proposed architecture and module ablations on deblurring performance (GoPro zoomed-in view)

Figure 10 provides a detailed visual analysis of how various architectural components contribute to the overall effectiveness of the proposed deblurring method on the GoPro dataset. Subfigure (a) presents the original blurred input image, while (b) shows a zoomed-in crop of a critical region, allowing a closer inspection of fine texture and edge restoration. Subfigure (c) displays the result when the architecture excludes A-DGFN (Dual Gating Feedforward Network with Attention), highlighting a significant drop in structural clarity and edge sharpness, indicating the crucial role of this attention-augmented module in modeling long-range dependencies.

In subfigure (d), the absence of B-DGFN leads to visible texture smearing and incomplete deblurring, demonstrating its

importance in complementary feature refinement. Subfigure (e) shows the output without CFFB (Cross Feature Fusion Block), where detail fusion across multi-scale representations is compromised, resulting in artifact-prone and less coherent restoration. Finally, subfigure (f) illustrates the full proposed model (IDC) with all modules intact, achieving the clearest result with high perceptual quality and minimal residual blur. This progressive comparison highlights how each module—the dual gating mechanism, cross-feature fusion, and attention-guided refinement—contributes synergistically to robust image restoration. The results validate the architectural design choices and affirm the effectiveness of the integrated components in handling real-world dynamic blur.

The ablation analysis confirms that each component

contributes significantly to restoration quality, as tabulated in Table 8. Removing NLM preprocessing reduces noise suppression capability, while excluding the Edge-Aware Loss weakens structural preservation. Similarly, removing IIBP refinement decreases reconstruction consistency and edge sharpness. The complete framework achieves the highest PSNR and SSIM values, validating the synergistic contribution of all modules.

Table 8. Quantitative ablation analysis on GoPro dataset

Configuration	PSNR (dB)	SSIM
Without NLM	30.84	0.931
Without Edge-Aware Loss	31.12	0.938
Without IIBP Refinement	31.46	0.942
Full Proposed Framework	32.74	0.952

Note: IIBP = Improved Iterative Back Projection; NLM = Non-Local Means; PSNR = Peak Signal-to-Noise Ratio; SSIM = Structural Similarity Index Measure.

4.5 Limitations and practical deployment considerations

Although the proposed hybrid framework achieves strong restoration quality, several practical limitations remain. The Transformer-enhanced DCNN and iterative refinement stages increase computational complexity, making real-time deployment on low-power edge devices challenging. High-resolution video processing may require GPU acceleration and memory optimization to maintain stable inference speed. Additionally, the framework assumes reasonably estimated blur kernels during IIBP refinement; inaccurate kernel estimation may affect reconstruction quality under highly non-uniform blur conditions. Future work will focus on lightweight model compression, adaptive kernel estimation, and real-time optimization for mobile and embedded cyber-physical imaging systems.

5. CONCLUSION

The proposed method, Hybrid Image Deblurring Using NLM and DCNN with IIBP, presents a comprehensive and synergistic framework for restoring high-fidelity images from blurred inputs. By strategically integrating NLM for initial denoising, a DCNN for learning complex blur patterns, and an IIBP module for refined reconstruction, the approach successfully addresses the limitations of conventional and purely deep learning-based methods. The use of an Edge-Aware Loss Function introduces spatial gradient sensitivity during CNN optimization, enabling the preservation of critical edge structures and fine textures that are often lost in traditional deblurring pipelines. Furthermore, the inclusion of Cubic B-Spline interpolation improves sub-pixel accuracy, enhancing visual smoothness and spatial consistency. Experimental results on benchmark datasets such as GoPro and RealBlur demonstrate the superiority of the proposed hybrid model in both objective metrics (PSNR, SSIM) and visual quality, especially in challenging dynamic scenes. Ablation studies confirm the necessity of each component, reinforcing the contribution of attention to edge-aware learning and error correction. Compared to state-of-the-art models like DeblurGAN-v2, Restormer, SRN, and MIMO-UNet, the proposed approach achieves better edge retention, reduced artifacts, and higher detail fidelity. Overall, this hybrid and edge-sensitive framework marks a significant

advancement in the field of image deblurring and proves highly effective for practical applications including surveillance, medical imaging, and consumer photography where image clarity is paramount.

REFERENCES

- [1] Abirami, R., Malathy, C. (2025). Secured DICOM medical image transition with optimized chaos method for encryption and customized deep learning model for watermarking. *Automatika*, 66(2): 173-187. <https://doi.org/10.1080/00051144.2025.2460877>
- [2] Zhao, H., Ke, Z., Chen, N., Wang, S., Li, K., Wang, L., Liu, C. (2020). A new deep learning method for image deblurring in optical microscopic systems. *Journal of Biophotonics*, 13(3): e201960147. <https://doi.org/10.1002/jbio.201960147>
- [3] Quan, Y., Lin, P., Xu, Y., Nan, Y., Ji, H. (2021). Nonblind image deblurring via deep learning in complex field. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10): 5387-5400. <https://doi.org/10.1109/TNNLS.2021.3070596>
- [4] Preethi, P., Swathika, R., Kaliraj, S., Premkumar, R., Yogapriya, J. (2024). Deep learning-based enhanced optimization for automated rice plant disease detection and classification. *Food and Energy Security*, 13(5): e70001. <https://doi.org/10.1002/fes3.70001>
- [5] Barman, T., Deka, B. (2023). A deep learning-based joint image super-resolution and deblurring framework. *IEEE Transactions on Artificial Intelligence*, 5(6): 3160-3173. <https://doi.org/10.1109/TAI.2023.3343319>
- [6] Nah, S., Kim, T.H., Lee, K.M. (2017). Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3883-3891. <https://doi.org/10.48550/arXiv.1612.02177>
- [7] Tao, X., Gao, H., Shen, X., Wang, J., Jia, J. (2018). Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 8174-8182. <https://doi.org/10.1109/CVPR.2018.00853>
- [8] Zhang, J., Pan, J., Ren, J., Song, Y., Bao, L., Lau, R.W.H., Yang, M.H. (2018). Dynamic scene deblurring using spatially variant recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 2521-2529. <https://doi.org/10.1109/CVPR.2018.00267>
- [9] Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J. (2018). DeblurGAN: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 8183-8192. <https://doi.org/10.1109/CVPR.2018.00854>
- [10] Kupyn, O., Martyniuk, T., Wu, J., Wang, Z. (2019). DeblurGAN-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, South Korea, pp. 8878-8887. <https://doi.org/10.1109/ICCV.2019.00897>
- [11] Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L. (2021). Multi-stage progressive

- image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Piscataway, NJ, USA, pp. 14821-14831. <https://doi.org/10.1109/CVPR46437.2021.01458>
- [12] Cho, S.J., Ji, S.W., Hong, J.P., Jung, S.W., Ko, S.J. (2021). Rethinking coarse-to-fine approach in single image deblurring. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, pp. 4641-4650. <https://doi.org/10.1109/ICCV48922.2021.00460>
- [13] Chen, L., Sun, Q., Wang, F. (2021). Attention-adaptive and deformable convolutional modules for dynamic scene deblurring. *Information Sciences*, 546: 368-377. <https://doi.org/10.1016/j.ins.2020.08.105>
- [14] Sukanuma, M., Liu, X., Okatani, T. (2019). Attention-based adaptive selection of operations for image restoration in the presence of unknown combined distortions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, pp. 9039-9048. <https://doi.ieeeecomputersociety.org/10.1109/CVPR.2019.00925>
- [15] Zhang, X., Dong, H., Hu, Z., Lai, W.S., Wang, F., Yang, M.H. (2018). Gated fusion network for joint image deblurring and super-resolution. *arXiv preprint arXiv:1807.10806*. <https://arxiv.org/abs/1807.10806>
- [16] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30. <https://doi.org/10.48550/arXiv.1706.03762>
- [17] Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. <https://doi.org/10.48550/arXiv.2010.11929>
- [18] Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H. (2022). Uformer: A general U-shaped transformer for image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, pp. 17683-17693. <https://doi.org/10.1109/CVPR52688.2022.01716>
- [19] Tsai, F.J., Peng, Y.T., Lin, Y.Y., Tsai, C.C., Lin, C.W. (2022). Stripformer: Strip transformer for fast image deblurring. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel*, pp. 146-162. https://doi.org/10.1007/978-3-031-19800-7_9
- [20] Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W. (2019). CCNet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, South Korea, pp. 603-612. <https://doi.org/10.1109/tpami.2020.3007032>
- [21] Palanisamy, P., Urooj, S., Arunachalam, R., Lay-Ekuakille, A. (2023). A novel prognostic model using chaotic CNN with hybridized spoofing for enhancing diagnostic accuracy in epileptic seizure prediction. *Diagnostics*, 13(21): 3382. <https://doi.org/10.3390/diagnostics13213382>
- [22] Preethi, P., Asokan, R. (2019). An attempt to design improved and fool proof safe distribution of personal healthcare records for cloud computing. *Mobile Networks and Applications*, 24(6): 1755-1762. <https://doi.org/10.1007/s11036-019-01379-4>
- [23] Rong, L., Huang, L. (2025). Image deblurring algorithm based on unsupervised network and alternating optimization iterations. *Multimedia Systems*, 31(2): 133. <https://doi.org/10.1007/s00530-025-01698-5>
- [24] Patibandla, K.K., Daruvuri, R., Mannem, P. (2025, April). Enhancing online retail insights: K-means clustering and PCA for customer segmentation. In *2025 3rd International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, Gharuan, India, pp. 388-393. <https://doi.org/10.1109/InCACCT65424.2025.11011448>
- [25] Daruvuri, R., Patibandla, K.K., Mannem, P. (2025). Data driven retail price optimization using XGBoost and predictive modeling. In *2025 International Conference on Intelligent Computing and Control Systems (ICICCS)*, Erode, India, pp. 220-225. <https://doi.org/10.1109/ICICCS65191.2025.10984940>
- [26] Preethi, P., Asokan, R. (2019). A high secure medical image storing and sharing in cloud environment using hex code cryptography method—Secure genius. *Journal of Medical Imaging and Health Informatics*, 9(7): 1337-1345. <https://doi.org/10.1166/jmihi.2019.2757>
- [27] Pawar, P., Ainapure, B. (2025). Novel technique to deblurring and blur detection techniques for enhanced visual clarity of ancient images. *International Journal of Electrical and Computer Engineering*, 15(2): 2314-2324. <https://doi.org/10.11591/ijece.v15i2.pp2314-2324>
- [28] Xiang, Y., Zhou, H., Li, C., Sun, F., Li, Z., Xie, Y. (2025). Deep learning in motion deblurring: Current status, benchmarks and future prospects. *The Visual Computer*, 41(6): 3801-3827. <https://doi.org/10.1007/s00371-024-03632-8>