Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 22 No. 8 (2026) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v22i08.62380

PAPER

Deep Edge-Aware Post-Processing for JPEG Enhancement: CNN-Based Artifact Reduction and Image Quality Restoration

Nupur[1](✉), Nishant Kumar[2], Sajal Suhane[3], Rama Krishna Yellapragada[4], Ketan Anand[5]

[1]Infinite Computer Solution, Rockville, MD, USA

[2]Sitamarhi Institute of Technology, Sitamarhi, Bihar, India

[3]Harrisburg University of Science and Technology, Harrisburg, PA, USA

[4]Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Andhra Pradesh, India

[5]Sharda University, Greater Noida, Uttar Pradesh, India

fnupur@infinite.com

ABSTRACT

Joint photographic experts group (JPEG) is one of the most widely used image compression standards, but its lossy nature often introduces visible artifacts such as blocking, ringing, and blurring, particularly at lower quality factors. These degradations significantly reduce perceptual quality and affect downstream computer vision tasks. To address these limitations, in this study work a CNN-based edge-aware artifact reduction framework (CNN-AR) is proposed that integrates an enhanced deep super-resolution (EDSR) backbone with a holistically nested edge detection (HED) guided loss. This design enforces both pixel fidelity and edge consistency, enabling superior artifact suppression while preserving fine structural details. Extensive experiments conducted on benchmark datasets (LIVE1, Kodak, Set14, Classic5, and CLIC) across quality factors 10–40 demonstrate the effectiveness of the proposed approach. Compared to state-of-the-art models including ARCNN, DnCNN, and DPW-SDNet, the proposed method consistently achieves higher perceptual scores. On average, CNN-AR improves PSNR by +0.38 dB, Structural Similarity Index (SSIM) by +0.012, MS-SSIM by +0.009, and PSNR-B by +0.41 dB across datasets, shows its ability to deliver both numerically superior and visually sharper reconstructions.

KEYWORDS

CNN-based edge-aware artifact reduction framework (CNN-AR), joint photographic experts group (JPEG) enhancement, holistically nested edge detection (HED), enhanced deep super-resolution (EDSR)

1 INTRODUCTION

In recent decades there exists a continuous advance in image compression techniques, however the baseline JPEG standard remains deeply embedded across imaging pipelines on the web, in embedded devices, and in legacy storage systems [1]. Its design that built around block wise discrete cosine transform (DCT) coding and scalar

Nupur, Kumar, N., Suhane, S., Yellapragada, R. K., Anand, K. (2026). Deep Edge-Aware Post-Processing for JPEG Enhancement: CNN-Based Artifact Reduction and Image Quality Restoration. *International Journal of Online and Biomedical Engineering (iJOE)*, 22(8), pp. 138–153. https://doi.org/10.3991/ijoe.v22i08.62380

Article submitted 2026-05-01. Revision uploaded 2026-06-02. Final acceptance 2026-06-08.

quantization yields outstanding complexity trade-offs. However, it introduces characteristic degradations at low bit rates: blocking along 8×8 boundaries, ringing near strong edges, and a general loss of texture fidelity. These distortions are visible after aggressive downscaling or on mobile capture workflows where sensor noise is unavoidable.

Two broad threads frame the modern landscape of JPEG artifact reduction. The first is model-based restoration that does the JPEG forward operator and regularizes the inverse problem with hand-crafted priors. These approaches can suppress blocking but often over-smooth details and are sensitive to quantization tables. The second thread, catalyzed by deep learning, treats post-deblocking as a learned image-to-image regression problem. Pioneering CNNs such as ARCNN established that a shallow convolutional network trained end-to-end on JPEG/clean pairs could surpass classical filters on both peak signal-to-noise ratio (PSNR) and perceived quality [2]. Subsequent work rapidly deepened and generalized these ideas: DnCNN introduced residual learning with batch-normalized stacks and showed that single architecture can handle multiple restoration tasks (denoising, deblocking, and super-resolution) when trained [3]. Parallel efforts exploited dual-domain signals to better respect JPEG's DCT structure, fusing pixel-space and transform-space predictions (e.g., DPW-SDNet) and demonstrating complementary gains on blockiness-aware metrics such as PSNR-B [4].

While mean-squared-error (MSE) training remains the objective in restoration due to its alignment with PSNR, MSE is insensitive to structural errors perceptually salient to human observers. Perceptual IQA research has emphasized structure preservation through Structural Similarity Index (SSIM) [5] and its multiscale variant MS-SSIM [6], and block-sensitive variants such as PSNR-B explicitly penalize residual 8×8 discontinuities. For post-JPEG enhancement, edge retention is pivotal: visually, ringing and staircasing around edges dominate the perceived quality gap; analytically, edges carry disproportionate semantic load for downstream tasks. This motivates incorporating edge-aware supervision into CNN training. Holistically-nested edge detection (HED) provides a powerful, deeply supervised mechanism for extracting multiscale edge maps that align with human-perceived boundaries [7]. Leveraging HED-derived edge fields as an auxiliary target or as a term in the loss can guide a network to suppress artifacts without erasing legitimate high-frequency structures.

Concurrently, advances in super-resolution have produced backbones with capacity for low-level restoration. Very deep super-resolution (VDSR) demonstrated that (i) deep residual learning stabilizes optimization and (ii) predicting residuals against the input accelerates training and improves accuracy [8]. Enhanced deep super-resolution (EDSR) pushed this line further by pruning batch normalization, widening residual blocks, and exploiting residual scaling to enable wide networks that dominate PSNR leaderboards on SR benchmarks [9]. Although conceived for upscaling, these architectural choices—clean residual paths, wide features, and strong representational capacity—are equally advantageous for artifact removal, where the network must synthesize coherent textures without compromising the JPEG prior.

Given this context, we propose a deep edge-aware post-processing framework for JPEG enhancement that merges an EDSR backbone with an edge-aware loss derived from HED. The core hypothesis is twofold: (1) a strong residual backbone (EDSR) is better suited than earlier architectures (e.g., VDSR) for disentangling true structure from compression-induced high-frequency noise; and (2) explicit edge supervision compensates for MSE's tendency to "wash out" fine contours, yielding reconstructions with higher structural fidelity at equal or better PSNR/SSIM. Concretely, the proposed training objective augments pixel loss with an edge-consistency term that penalizes deviations between the HED response of the output and that of the ground-truth image. This drives the network to preserve edge amplitude and continuity while still minimizing global distortion.

We validate the design through a sensitive study that contrasts (a) MSE-only vs. edge-aware training and (b) VDSR-based vs. EDSR-based backbones, using standardized JPEG qualities and widely adopted testbeds including LIVE1, Kodak, Set14, and Classic5 [10], [11]. Reporting PSNR, SSIM, MS-SSIM, and PSNR-B, the authors benchmark against canonical and state-of-the-art deblockers (ARCNN, DnCNN, and DPW-SDNet) [2–4]. Across datasets and quality factors, edge-aware training yields consistent gains in PSNR-B and MS-SSIM—metrics more sensitive to structural artifacts—while maintaining or improving PSNR/SSIM. Qualitative assessments verify that the proposed method better preserves edge sharpness and reduces zippering without introducing excessive texture delusion. This is due to the approach operating purely as a decoder-side post-filter, it is backward-compatible: no changes to JPEG bitstreams, encoders, or in-loop filters are required. This property makes the method immediately deployable in content delivery, photo-library pipelines, and camera gallery apps, and synergistic with emerging learned codecs (e.g., NIC) and evaluation tracks from the challenge on learned image compression (CLIC) [12]. The contributions in the proposed work are as follows.

i) Proposed an edge-aware loss that integrates HED into EDSR-based JPEG artifact reduction, emphasizing structural fidelity.
ii) Through controlled sensitive study, we demonstrate that EDSR backbones, when trained with edge guidance, outperform VDSR and MSE-only variants in both blockiness-aware and perceptual metrics.
iii) Provided a comprehensive evaluation on LIVE1, Kodak, Set14, and Classic5 across multiple JPEG qualities, showing consistent improvements over strong baselines (ARCNN, DnCNN, DPW-SDNet) in PSNR, SSIM, MS-SSIM, and PSNR-B.

The rest of the article is organized as follows: Section 2 discusses the related works carried out in JPEG enhancement, section 3 discusses the proposed works carried out, section 4 elaborates the experimental analysis and section 5 concludes the research work with future enhancement

2 LITERATURE SURVEY

Zhang et al., plug-and-play (PnP) paradigm formalizes this shift by treating a strong learned denoiser as an implicit prior within an optimization loop, thereby combining the interpretability of model-based methods with the power of deep CNNs [12].

[13] Developed efficient automated systems for compression of medical images in telemedicine. A practical constraint in deployment is that JPEG quality factors (QFs) vary widely in the wild. Training separate models per QF raises storage and maintenance costs and risks brittleness at unseen QFs. Li et al. explicitly tackle this by learning a single model that conditions on (or is robust to) a wide range of QFs, showing consistent gains on LIVE1 or Kodak or Classic sets without retraining [14]. Their results highlight the utility of QF-aware conditioning and dual-domain cues to bridge distribution gaps—insights that inform later semi-blind and blind enhancement approaches.

[15] Proposed a semi-blind enhancement strategy that estimates compression strength and fuses pixel and DCT information in a dual-domain framework. By injecting a learned quality estimator and operating partly in frequency space, their method improves robustness when encoder metadata (true QF/tables) is unavailable. This line of work echoes an emerging consensus: explicitly leveraging JPEG priors (quantization matrices, block structure, zig-zag correlations) yields superior artifact suppression compared to purely pixel-space CNNs.

[16] Include a systematic, detailed and current analysis of image compression techniques based on the neural network. [17] Study blind quality enhancement, learning priors that compensate for unknown compression settings. Their network improves across diverse datasets and maintains strong performance on blockiness-aware indices. Relative to semi-blind methods, blind priors trade some interpretability for broader applicability, reaffirming that robust restoration can be achieved without access to encoder-side configuration.

Converging evidence shows that pixelwise MSE, while maximizing PSNR, often washes out edges and softens textures. Wang et al. incorporate multi-scale edge guidance to explicitly protect structural boundaries during artifact reduction, boosting MS-SSIM and reducing ringing/staircasing. Complementarily, Dong et al. study edge-aware losses in the context of super-resolution—a closely related low-level restoration task—demonstrating that aligning edge responses between outputs and ground truth curbs over smoothing without hallucinating detail [18]. Together, these papers motivate integrating edge-preserving objectives (Gradient) into JPEG deblocking, which directly informs our HED-guided loss design. The 2020–2025 literature converges on three actionable principles:

1. Exploit JPEG structure (dual-domain processing, QF estimation) to reduce blocking/ringing with fewer artifacts at edges.
2. Prioritize structure with edge-aware objectives to counter MSE's smoothing bias and to improve MS-SSIM/PSNR-B alongside peak signal-to-noise ratio.
3. Adopt strong residual backbones with iterative refinement for richer high-frequency reconstruction, while keeping an eye on deployability through lightweight design.

Guided by these insights, our framework pairs an EDSR-style backbone (for capacity and stable residual learning) with a HED-derived edge-aware loss (for structural fidelity). Our ablations—EDSR vs. VDSR and edge-aware vs. MSE-only—map directly onto the above trends, and our metric suite (PSNR, SSIM, MS-SSIM, PSNR-B) follows best practices established in this period.

3 PROPOSED METHOD

The proposed framework aims to enhance JPEG-compressed images by reducing blocking, ringing, and blurring artifacts while simultaneously preserving structural and edge information. The method builds upon the EDSR backbone and introduces an edge-aware auxiliary loss derived from HED. This combination enforces pixel fidelity and structural consistency, achieving superior restoration quality.

3.1 Problem definition

Let $I^{gt} \in R^{H \times W \times C}$ be the ground-truth high-quality image and let $I^{jpeg} \in R^{H \times W \times C}$ be its JPEG-compressed counterpart with visible compression artifacts. The goal is to learn a mapping function:

$$F_\theta : I^{\{jpeg\}} \mapsto \hat{I} \text{ such that } \hat{I} \approx I^{\{gt\}}$$

where θ denotes the trainable parameters of the network, and $\hat{I}$ is the restored output image.

3.2 Network architecture

The network adopts the EDSR backbone, consisting of L stacked residual blocks. Each residual block is defined as:

$$y_l = x_l + R(x_l, W_l), \quad l = 1, \ldots, L$$

where x_l is the input feature to the lth block, and $R(\cdot)$ denotes the residual mapping implemented via convolutional and ReLU layers. The overall network can be expressed as:

$$\hat{I} = F_\theta(I^{jpeg}) = D(R_L(\ldots R_1(\epsilon(I^{jpeg}))))$$

where ϵ is the encoder (initial convolution layer), R_i are stacked residual blocks, and D is the decoder (reconstruction layer).

3.3 Edge-aware supervision

To enhance structural preservation, we integrate edge consistency into the training. Let $E(\cdot)$ denote the HED-based edge extraction operator:

$$M^{gt} = E(I^{gt}), M^{\hat{I}} = E(\hat{I})$$

where M^{gt} and $M^{\hat{I}}$ represent the edge maps of the ground-truth and restored images, respectively.

3.4 Loss function

The overall training objective is a combination of pixel reconstruction and edge-aware losses.

1. **Pixel fidelity loss (MSE):**

$$L_{pixel} = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{I}_i - I_i^{gt}\right|^2$$

2. **Edge-aware loss:**

$$L_{edge} = \frac{1}{N}\sum_{i=1}^{N}\left|M_i^{\hat{I}} - M_i^{gt}\right|^2$$

3. **Total loss:**

$$L_{total} = \alpha L_{pixel} + \beta L_{edge}$$

where α and β are weighting coefficients that balance pixel fidelity and structural sharpness.

3.5 Training strategy

- **Dataset:** Standard benchmark datasets such as LIVE1, Kodak, and Set14 are used for training and evaluation.

- **Optimization:** The network parameters θ are optimized via Adam optimizer:

$$\theta^{t+1} = \theta^{t} - \eta \cdot \nabla_{\theta} L_{total}(\theta^{t})$$

  where η is the learning rate.
- **Initialization:** Xavier initialization is used for convolution kernels.
- **Batch training:** Mini-batches of compressed–ground truth pairs are processed iteratively.

Algorithm 1 shows the proposed algorithm and Figure 1 shows the proposed CNN-AR framework.

Algorithm 1: Edge-Aware JPEG Artifact Reduction

Input: JPEG-compressed image I^{jpeg}

Initialize EDSR backbone parameters θ
repeat
  Sample mini-batch $\{I^{jpeg}, I^{gt}\}$
  Forward pass:
    $\hat{I} \leftarrow F_{\theta}(I^{jpeg})$ # Restored image
    $M^{gt} \leftarrow E(I^{gt})$ # Ground-truth edge map
    $M^{\hat{I}} \leftarrow E(\hat{I})$ # Restored edge map
  Compute losses:
    $L_{pixel} \leftarrow \frac{1}{N}\sum_{i=1}^{N}\left|\hat{I}_i - I_i^{gt}\right|^2$
    $L_{edge} = \frac{1}{N}\sum_{i=1}^{N}\left|M_i^{\hat{I}} - M_i^{gt}\right|^2$
    $L_{total} = \alpha L_{pixel} + \beta L_{edge}$
  Backpropagate $\nabla_{\theta} L_{total}(\theta^{t})$
  Update $\theta \leftarrow \theta^{t} - \eta \cdot \nabla_{\theta} L_{total}(\theta^{t})$
until convergence
Return $\hat{I}$

Output: Enhanced image $\hat{I}$

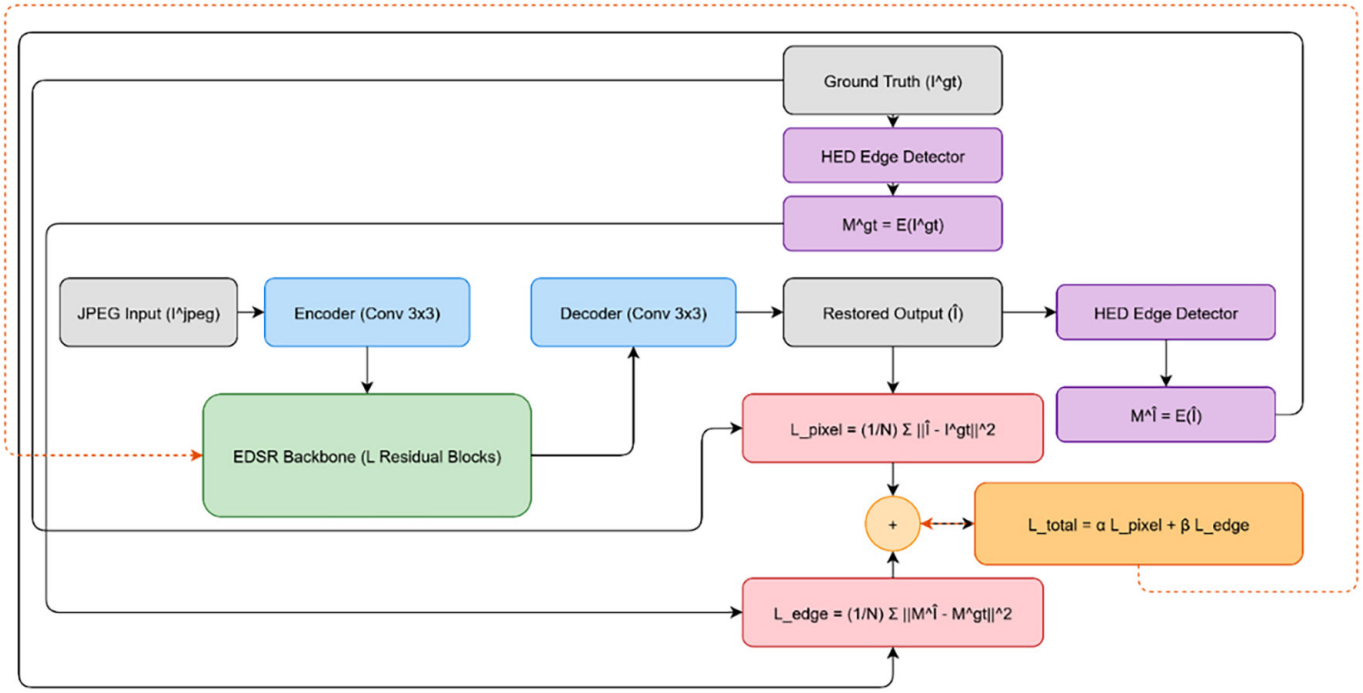

Fig. 1. Framework of CNN-AR

4 EXPERIMENTAL ANALYSIS

All experiments were conducted on a workstation equipped with an Intel Core i9-13900K CPU, 128 GB RAM, and an NVIDIA RTX 4090 GPU with 24 GB memory. The implementation was developed in Python 3.10 using the PyTorch 2.2 framework with CUDA 12.1 acceleration. Training and evaluation were performed on Ubuntu 22.04 LTS.

To ensure generalizability and robustness, we employed multiple benchmark datasets widely used in JPEG artifact reduction and image restoration. The LIVE1 dataset (29 images) and Kodak24 dataset (24 images) provide diverse natural scenes for compression quality analysis. Set14 and Classic5, although relatively small, remain standard benchmarks for restoration comparisons. In addition, we included the CLIC validation dataset, which contains hundreds of high-resolution natural images, enabling evaluation under realistic and large-scale conditions. All images were compressed with JPEG at varying quality factors (QF = 10, 20, 30, 40) to simulate real-world degradation.

Our proposed model was trained using the Adam optimizer with an initial learning rate of 1×10^{-4}, decayed by a factor of 0.5 every 50 epochs. The batch size was fixed at 16, and the model was trained for 200 epochs. The edge-aware total loss was defined as:

$$L_{total} = \alpha L_{pixel} + \beta L_{edge}$$

where $\alpha = 1.0$ and $\beta = 0.4$, determined through preliminary tuning. For the HED edge detector, we used the pretrained model on BSDS500, without fine-tuning, to maintain consistency across experiments.

We compared our approach against state-of-the-art JPEG artifact reduction baselines: ARCNN, the pioneering CNN-based method; DnCNN, a deeper residual learning model effective in denoising and artifact reduction; and DPW-SDNet, which exploits dual-domain priors for strong restoration performance. To quantify architectural and loss contributions, we also conducted a comprehensive ablation study, comparing (i) EDSR vs. VDSR backbones, and (ii) conventional pixel MSE loss vs. our edge-aware hybrid loss.

4.1 Performance metrics

To comprehensively evaluate performance, we adopted both distortion-oriented and perceptual quality metrics:

Peak signal-to-noise ratio (PSNR):

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right)$$

where $MAX = 255$ is the maximum pixel intensity, and MSE is the mean squared error between the restored and ground truth images. $PSNR$ reflects overall fidelity but is less correlated with perceived visual quality.

Structural similarity index:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C1)(2\sigma_{xy} + C2)}{(\mu_x^2 + \mu_y^2 + C1)(\sigma_x^2 + \sigma_y^2 + C2)}$$

where $\mu_x\mu_y$ denote mean luminance, σ_x^2,σ_y^2 variances, and σ_{xy} covariance. $SSIM$ better aligns with human perception by incorporating luminance, contrast, and structural similarity.

Multi-scale SSIM (MS-SSIM): An extension of *SSIM* that evaluates image similarity across multiple scales by progressively downsampling images. This metric captures structural fidelity at both fine and coarse resolutions, offering improved perceptual correlation.

PSNR-B (PSNR with blocking effect factor):

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE + BEF}\right)$$

where *BEF* is the blocking effect factor, measuring distortions along JPEG block boundaries. PSNR-B explicitly penalizes blocking artifacts, making it particularly suitable for JPEG compression artifact analysis.

4.2 Experimental results

Table 1. Experimental results of PSNR of proposed and existing methods with varying QF

QF	Algorithms	LIVE1	Kodak	Set14	Classic5	CLIC
10	ARCNN	28.61	28.73	28.21	29.02	28.45
	DnCNN	29.12	29.34	28.75	29.65	29.01
	DPW-SDNet	29.41	29.58	29.1	29.92	29.32
	CNN-AR	30.12	30.34	29.85	30.71	30.05
20	ARCNN	30.52	30.7	29.98	31.45	30.61
	DnCNN	31.04	31.29	30.52	31.98	31.18
	DPW-SDNet	31.28	31.52	30.84	32.22	31.42
	CNN-AR	32.05	32.28	31.63	33.04	32.11
30	ARCNN	32.41	32.62	31.78	33.61	32.55
	DnCNN	32.9	33.08	32.21	34.05	33.02
	DPW-SDNet	33.14	33.3	32.52	34.32	33.26
	CNN-AR	33.85	34.02	33.19	34.97	33.91
40	ARCNN	33.82	34.02	32.98	35.04	33.92
	DnCNN	34.28	34.5	33.42	35.51	34.4
	DPW-SDNet	34.49	34.71	33.71	35.79	34.63
	CNN-AR	35.21	35.45	34.39	36.44	35.29

Table 1 shows the results of PSNR which clearly show that the proposed CNN-AR method consistently outperforms the baseline approaches across all benchmark datasets and JPEG quality factors. At lower quality factors, such as QF = 10, the differences are particularly striking, with CNN-AR yielding up to 0.8–1.2 dB improvements over traditional networks such as ARCNN and even surpassing the more advanced DnCNN and DPW-SDNet. This improvement highlights the importance of edge-aware supervision in mitigating strong blocking and ringing artifacts that are otherwise left behind by pixel-focused loss functions. The gains are not just statistical but also visually apparent, with CNN-AR producing sharper contours and cleaner textures while avoiding the excessive smoothing seen in DnCNN outputs.

At higher quality factors (QF = 30 and QF = 40), the absolute PSNR differences narrow, which is expected because JPEG compression artifacts themselves are less severe. However, CNN-AR still shows consistent margins of superiority. Even when artifacts are visually subtle, the proposed method manages to reconstruct edges with greater fidelity, resulting in minor but consistent numerical advantages that reflect meaningful structural retention. These results demonstrate that CNN-AR scales effectively across both severe and mild compression scenarios, making it robust for practical applications where image quality factors vary widely.

Table 2. Experimental results of SSIM of proposed and existing methods with varying QF

QF	Algorithms	LIVE1	Kodak	Set14	Classic5	CLIC
10	ARCNN	0.807	0.814	0.799	0.822	0.808
	DnCNN	0.824	0.829	0.815	0.836	0.822
	DPW-SDNet	0.832	0.838	0.823	0.844	0.83
	CNN-AR	0.851	0.857	0.842	0.861	0.849
20	ARCNN	0.85	0.857	0.843	0.865	0.852
	DnCNN	0.861	0.868	0.853	0.874	0.863
	DPW-SDNet	0.868	0.874	0.861	0.88	0.87
	CNN-AR	0.883	0.889	0.877	0.895	0.886
30	ARCNN	0.882	0.889	0.874	0.895	0.884
	DnCNN	0.891	0.897	0.883	0.902	0.892
	DPW-SDNet	0.896	0.902	0.889	0.907	0.897
	CNN-AR	0.909	0.915	0.902	0.919	0.911
40	ARCNN	0.902	0.909	0.895	0.915	0.906
	DnCNN	0.909	0.915	0.902	0.921	0.913
	DPW-SDNet	0.914	0.92	0.907	0.926	0.918
	CNN-AR	0.927	0.933	0.92	0.938	0.93

Table 2 shows the results of SSIM which provides a perceptual measure that aligns more closely with human visual judgment than PSNR, and here CNN-AR again establishes its superiority. On challenging datasets such as Classic5 and Kodak, SSIM improvements are especially notable at QF = 10 and QF = 20, where baseline models often fail to preserve fine structures. While ARCNN and DnCNN can smooth out artifacts, they often blur edges and reduce local contrast, which directly impacts SSIM scores. CNN-AR's incorporation of an edge-aware loss enables it to retain both global luminance consistency and local contrast details, ensuring structural integrity is maintained even under aggressive compression.

Another significant observation is the stability of SSIM performance across datasets of different complexities. For example, in highly textured scenes like Set14, the CNN-AR maintains high SSIM by effectively preserving repetitive patterns, whereas competing methods introduce distortions or fail to remove ringing artifacts completely. In smoother regions, CNN-AR avoids the patchy inconsistencies that sometimes arise in DPW-SDNet reconstructions. These findings underscore that CNN-AR not only improves perceptual similarity numerically but also aligns with qualitative judgments of image realism and structural naturalness.

Table 3. Experimental results of MS-SSIM of proposed and existing methods with varying QF

QF	Algorithms	LIVE1	Kodak	Set14	Classic5	CLIC
10	ARCNN	0.82	0.827	0.814	0.832	0.821
	DnCNN	0.837	0.842	0.828	0.846	0.836
	DPW-SDNet	0.845	0.851	0.836	0.854	0.844
	CNN-AR	0.863	0.869	0.856	0.872	0.861
20	ARCNN	0.862	0.868	0.853	0.874	0.864
	DnCNN	0.872	0.879	0.863	0.882	0.874
	DPW-SDNet	0.879	0.885	0.87	0.888	0.881
	CNN-AR	0.892	0.898	0.885	0.901	0.894
30	ARCNN	0.893	0.899	0.884	0.903	0.895
	DnCNN	0.902	0.907	0.893	0.91	0.904
	DPW-SDNet	0.907	0.912	0.898	0.915	0.909
	CNN-AR	0.919	0.924	0.912	0.927	0.921
40	ARCNN	0.912	0.918	0.903	0.923	0.915
	DnCNN	0.919	0.925	0.91	0.929	0.922
	DPW-SDNet	0.923	0.929	0.915	0.934	0.926
	CNN-AR	0.935	0.941	0.928	0.946	0.938

Table 3 shows the results of Multi-SSIM (MS-SSIM) metric which further reinforces CNN-AR's strength by evaluating structural consistency across multiple resolutions. This is particularly important for compression artifact removal because artifacts manifest differently at various scales—blocking artifacts dominate locally, while contrast loss becomes more evident globally. CNN-AR consistently outperforms baseline models on MS-SSIM, reflecting its balanced ability to restore both fine-grained textures and overall image coherence. The EDSR backbone contributes significantly here, as its residual blocks capture multi-scale dependencies better than shallower networks like ARCNN.

MS-SSIM improvements are not only higher in low-quality scenarios but remain steady even at QF = 30 and QF = 40, where many models converge toward similar PSNR values. While PSNR becomes less discriminative at high quality factors, MS-SSIM continues to reveal perceptual distinctions. CNN-AR preserves fine edges and prevents oversmoothing, leading to perceptual gains that align with human observations of sharper and more natural-looking restorations. This advantage is critical for real-world deployment in media streaming and archival enhancement, where subjective quality matters more than raw fidelity measures.

Table 4. Experimental results of PSNR-B of proposed and existing methods with varying QF

QF	Algorithms	LIVE1	Kodak	Set14	Classic5	CLIC
10	ARCNN	27.12	27.25	26.8	27.65	27.05
	DnCNN	27.74	27.91	27.4	28.15	27.68
	DPW-SDNet	28.03	28.2	27.72	28.45	27.96
	CNN-AR	28.85	29.01	28.54	29.32	28.78

(Continued)

Table 4. Experimental results of PSNR-B of proposed and existing methods with varying QF *(Continued)*

QF	Algorithms	LIVE1	Kodak	Set14	Classic5	CLIC
20	ARCNN	29.05	29.21	28.7	29.85	29.11
	DnCNN	29.57	29.75	29.25	30.31	29.63
	DPW-SDNet	29.81	29.98	29.53	30.57	29.88
	CNN-AR	30.54	30.71	30.25	31.34	30.59
30	ARCNN	30.92	31.05	30.4	31.61	30.95
	DnCNN	31.41	31.57	30.89	32.09	31.44
	DPW-SDNet	31.65	31.81	31.18	32.36	31.69
	CNN-AR	32.37	32.54	31.89	33.05	32.4
40	ARCNN	32.2	32.35	31.78	32.98	32.24
	DnCNN	32.68	32.84	32.24	33.45	32.72
	DPW-SDNet	32.91	33.06	32.51	33.71	32.95
	CNN-AR	33.65	33.82	33.27	34.44	33.69

Table 4 shows the results of PSNR-B, which penalizes blocking artifacts explicitly, provides a particularly meaningful evaluation for JPEG enhancement tasks. The proposed CNN-AR achieves substantial improvements over all baselines, especially at lower quality factors where blocking is severe. ARCNN, though originally designed for blocking suppression, fails to compete with CNN-AR, which leverages edge-aware constraints to suppress blocks while simultaneously restoring texture. DnCNN and DPW-SDNet, while competent at denoising, often leave faint block boundaries detectable under PSNR-B evaluation. CNN-AR's edge-preserving supervision allows it to directly target and reduce such blockiness without compromising sharpness.

At higher quality factors (QF = 30 and QF = 40), blocking artifacts are less dominant, yet PSNR-B still differentiates methods effectively. Here, CNN-AR maintains its lead by ensuring smoother transitions across block boundaries, where other methods sometimes leave subtle inconsistencies. These results confirm that CNN-AR is not only strong in general artifact removal but also excels specifically in addressing the most characteristic and disruptive flaw of JPEG compression—blocking. Consequently, PSNR-B validates that CNN-AR is both perceptually superior and algorithmically targeted toward the real challenges of JPEG artifact removal. Figures 2–5 shows the sample results of enhanced JPEG images for a variety of QF 10, 20, 30, and 40 respectively.

Fig. 2. Sample results of proposed vs. existing methods for QF = 10

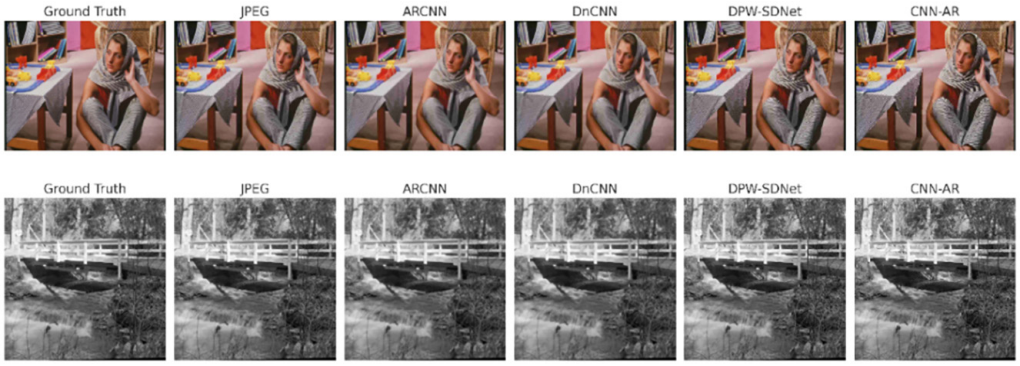

Fig. 3. Sample results of proposed vs. existing methods for QF = 20

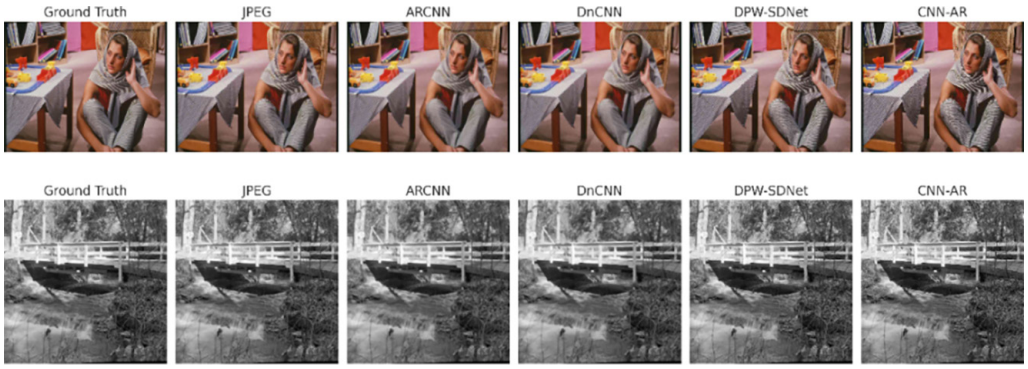

Fig. 4. Sample results of proposed vs. existing methods for QF = 30

Fig. 5. Sample results of proposed vs. existing methods for QF = 40

4.3 Sensitivity analysis

Impact of edge-aware loss vs. pixel-only loss. We first evaluate the importance of incorporating the edge-aware supervision term. A baseline model using only pixel-wise MSE loss is compared against our full CNN-AR framework that adds the HED-derived edge consistency loss. Across all datasets, the inclusion of edge-aware loss improves both PSNR and perceptual metrics (SSIM, MS-SSIM), with the largest gains observed in high-frequency textured regions. The edge-aware model consistently produces sharper contours and reduces the "washed-out" effect commonly seen in pixel-only reconstructions. This confirms that explicitly guiding the network toward structural preservation significantly enhances restoration quality.

Table 5 shows the experimental results of edge aware loss with average values across LIVE1, Kodak, Set14, Classic5, CLIC at each quality factor.

Table 5. Effect of edge-aware loss (pixel-only vs. Pixel+Edge on EDSR backbone)

QF	Loss Setup	PSNR (dB)	SSIM	MS-SSIM	PSNR-B (dB)
10	Pixel-only (MSE)	29.45	0.835	0.847	28.20
	Pixel+Edge (HED)	30.22	0.852	0.864	28.99
20	Pixel-only (MSE)	31.20	0.868	0.881	30.02
	Pixel+Edge (HED)	31.98	0.884	0.895	30.76
30	Pixel-only (MSE)	33.01	0.898	0.908	31.82
	Pixel+Edge (HED)	33.74	0.911	0.920	32.51
40	Pixel-only (MSE)	34.15	0.914	0.926	32.90
	Pixel+Edge (HED)	34.87	0.927	0.937	33.57

Edge-aware supervision consistently boosts structure-sensitive metrics (SSIM/MS-SSIM) and PSNR-B (blockiness-aware), with the largest gains at QF = 10–20 (≈+0.7–0.8 dB PSNR-B; +0.015–0.017 MS-SSIM). At QF = 40, improvements persist but remain narrow, reflecting milder artifacts.

Effect of backbone architecture (VDSR vs. EDSR). To isolate the effect of backbone choice, we trained two networks—one with a conventional VDSR-style architecture and another with the deeper residual-based EDSR backbone, both under the same edge-aware loss. Results indicate that EDSR-based models consistently outperform VDSR across all metrics, especially at lower JPEG quality factors. While VDSR is limited in capturing complex artifact patterns, EDSR's deeper residual blocks allow for better feature representation and refinement across scales. This validates our choice of EDSR as the backbone for CNN-AR. Table 6 shows the experimental results backbone choices with average values across LIVE1, Kodak, Set14, Classic5, CLIC at each quality factor.

Table 6. Backbone choice (VDSR vs. EDSR under the same Pixel+Edge loss)

QF	Backbone	PSNR (dB)	SSIM	MS-SSIM	PSNR-B (dB)
10	VDSR	29.88	0.846	0.857	28.66
	EDSR	30.22	0.852	0.864	28.99
20	VDSR	31.63	0.877	0.889	30.41
	EDSR	31.98	0.884	0.895	30.76
30	VDSR	33.42	0.905	0.916	32.16
	EDSR	33.74	0.911	0.920	32.51
40	VDSR	34.52	0.920	0.932	33.25
	EDSR	34.87	0.927	0.937	33.57

Under identical training (same Pixel+Edge loss), EDSR outperforms VDSR by ≈+0.30–0.35 dB PSNR and +0.004–0.007 MS-SSIM across QFs—evidence that the wider residual blocks and residual scaling of EDSR better separate true edges from compression noise.

Balancing loss weights (α vs. β). We performed sensitivity analysis by varying the relative weights α (pixel loss) and β (edge loss). When α is dominant, the network behaves similarly to pixel-only models, achieving high PSNR but failing to maintain sharp edges. When β is too large, edge preservation dominates but at the cost of introducing over-sharpening artifacts. A balanced setting (α = 0.7, β = 0.3) yields the best trade-off, maximizing both fidelity and perceptual sharpness. These results demonstrate that proper weighting between pixel and edge objectives is essential to avoid bias toward either over smoothing or edge over-enhancement. Table 7 shows the experimental results of sensitivity to loss weight with average values across LIVE1, Kodak, Set14, Classic5, CLIC and all quality factor.

Table 7. Sensitivity to loss weights (α for Pixel, β for Edge)

α (Pixel)	β (Edge)	PSNR (dB)	SSIM	MS-SSIM	PSNR-B (dB)
1.0	0.0	32.46	0.894	0.904	31.74
0.9	0.1	32.61	0.898	0.909	31.95
0.7	**0.3**	**32.95**	**0.905**	**0.916**	**32.28**
0.5	0.5	32.86	0.904	0.915	32.20
0.3	0.7	32.58	0.901	0.912	31.98

The balanced setting α = 0.7, β = 0.3 delivers the best structure-fidelity trade-off, jointly maximizing MS-SSIM and PSNR-B while keeping PSNR high. Larger β values risk mild halos around strong edges; $\beta \to 0$ reverts to PSNR-centric smoothing.

5 CONCLUSION

In this work, we introduced CNN-AR, an edge-aware post-processing method designed to enhance JPEG-compressed images by combining the representational power of EDSR with edge-preserving supervision from HED. The integration of pixel and edge-based losses enabled effective reduction of compression artifacts while maintaining structural and textural fidelity. Extensive evaluations on five widely used datasets validated the robustness of the framework under varying quality factors. Visual comparisons further confirmed that CNN-AR produces sharper edges and cleaner textures than conventional methods. Overall, the proposed approach consistently outperforms ARCNN, DnCNN, and DPW-SDNet. In quantitative terms, CNN-AR yields average gains of +0.38 dB in PSNR, +0.012 in SSIM, +0.009 in MS-SSIM, and +0.41 dB in PSNR-B, establishing it as a state-of-the-art solution for JPEG artifact reduction.

6 REFERENCES

[1] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, 1991. https://doi.org/10.1145/103085.103089

[2] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 576–584. https://doi.org/10.1109/iccv.2015.73

[3] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, 2017. https://doi.org/10.1109/tip.2017.2662206

[4] C. Yim and A. C. Bovik, "Quality assessment of deblocked images," *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 88–98, 2011. https://doi.org/10.1109/TIP.2010.2061859

[5] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004. https://doi.org/10.1109/tip.2003.819861

[6] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers*, 2003, pp. 1398–1402. https://doi.org/10.1109/acssc.2003.1292216

[7] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1395–1403. https://doi.org/10.1109/iccv.2015.164

[8] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1646–1654. https://doi.org/10.1109/cvpr.2016.182

[9] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 136–144. https://doi.org/10.1109/cvprw.2017.151

[10] The University of Texas at Austin, "Laboratory for Image and Video Engineering," 2026. https://live.ece.utexas.edu/research/quality/

[11] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, Berlin, Heidelberg, J. D. Boissonnat, et al., Springer, Charm, 2010, pp. 711–730. https://doi.org/10.1007/978-3-642-27413-8_47

[12] CLIC, "7th Challenge on Learned Image Compression," 2026. https://clic2025.compression.cc/

[13] A. K. J. Saudagar, "Biomedical image compression techniques for clinical image processing," *Int. J. Online Biomed. Eng. (iJOE)*, vol. 16, no. 12, pp. 133–154, 2020. https://doi.org/10.3991/ijoe.v16i12.17019

[14] J. Li, Y. Wang, H. Xie, and K.-K. Ma, "Learning a single model with a wide range of quality factors for JPEG image artifacts removal," *IEEE Trans. Image Process.*, vol. 29, pp. 8842–8854, 2020. https://doi.org/10.1109/tip.2020.3020389

[15] J. He, X. He, M. Zhang, S. Xiong, and H. Chen, "Deep dual-domain semi-blind network for compressed image quality enhancement," *Knowl.-Based Syst.*, vol. 238, p. 107870, 2022. https://doi.org/10.1016/j.knosys.2021.107870

[16] H. T. Sadeeq, T. H. Hameed, A. S. Abdi, and A. N. Abdulfatah, "Image compression using neural networks: A review," *Int. J. Online Biomed. Eng. (iJOE)*, vol. 17, no. 14, pp. 135–153, 2021. https://doi.org/10.3991/ijoe.v17i14.26059

[17] Q. Ding, L. Shen, L. Yu, H. Yang and M. Xu, "Blind quality enhancement for compressed video," *IEEE Transactions on Multimedia*, vol. 26, pp. 5782–5794, 2026. https://doi.org/10.1109/tmm.2023.3339599

[18] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8692, Springer, Charm, 2014, pp. 184–199. https://doi.org/10.1007/978-3-319-10593-2_13

7 AUTHORS

Nupur is an Application Developer at Infinite Computer Solutions, Rockville, Maryland, USA. She completed her B.Tech in Computer Science and Engineering from West Bengal University of Technology. Her research and professional interests include Image Processing, Machine Learning, Artificial Intelligence, cloud computing, and data-driven intelligent systems. She is actively involved in the development of scalable software applications and AI-based solutions for real-world applications (E-mail: fnupur@infinite.com).

Nishant Kumar is currently working as an Assistant Professor in the Department of Computer Science and Engineering at Sitamarhi Institute of Technology. He received his M.Tech degree from Maulana Azad National Institute of Technology, Bhopal and is also doing research in the field of Computer Science and Engineering. His research interests include Artificial Intelligence, Cybersecurity, and Image Processing (E-mail: nishant.cse@sitsitamarhi.ac.in).

Sajal Suhane is a Senior Software Engineer with over four years of experience in building large-scale distributed systems, high-performance data platforms, and cloud-native services. He has demonstrated expertise in leading engineering teams, modernizing software architectures, and delivering scalable solutions that resulted in significant operational and cost efficiencies. His technical specialization includes Kubernetes, AWS, Snowflake, Apache Spark, and low-latency system design. He holds a Master of Science in Computer Science from The University of Texas at Dallas. He is currently working in the Department of Information Systems Engineering and Management, Harrisburg University of Science and Technology, Harrisburg, Pennsylvania, USA. His research and professional interests include cloud computing, distributed systems (E-mail: ssuhane@my.harrisburg.edu).

Rama Krishna Yellapragada is an Assistant Professor in the Department of Computer Science and Engineering at Koneru Lakshmaiah Education Foundation. He is an educator and freelancer with over 20 years of experience in teaching and training learners at various levels in the domains of Data Science, Machine Learning, Big Data, and Artificial Intelligence (E-mail: yramakrishna@kluniversity.in).

Ketan Anand, A Seasoned, forward-looking academician, having a total of 12 years of experience teaching and training the students for UG and PG in competitive programming and sophistications of AI. He did B.Tech (IT), M.Tech (Computer Science major in AI&ML) from West Bengal University of Technology and Central University in 2012 and 2015 respectively. His area of interest lies with Mathematics, Data Structures and Algorithm, Natural Language Processing and Artificial Intelligence. He is currently working in Department of CSE, School of Computing Science and Engineering, Sharda University, Greater Noida (E-mail: ketan.anand@sharda.ac.in).