VRAE: Vertical Residual Autoencoder for License Plate Denoising and Deblurring

Cuong Nguyen[1], Dung T. Tran[1,2], Hong Nguyen[3],
Xuan-Vu Phan[1], Nam-Phong Nguyen[1*]

[1]Ha Noi University of Science and Technology, Ha Noi, Vietnam.
[2]Center of Environmental Intelligence, VinUniversity, Hanoi, Vietnam.
[3]University of Southern California, Los Angeles, United States.

*Corresponding author(s). E-mail(s): phong.nguyennam@hust.edu.vn;
Contributing authors: cuong.nt227090@sis.hust.edu.vn;
dung.tt2@vinuni.edu.vn; hongn@usc.edu; vu.phanxuan@hust.edu.vn;

Abstract

In real-world traffic surveillance, vehicle images captured under adverse weather, poor lighting, or high-speed motion often suffer from severe noise and blur. Such degradations significantly reduce the accuracy of license plate recognition systems, especially when the plate occupies only a small region within the full vehicle image. Restoring these degraded images a fast realtime manner is thus a crucial pre-processing step to enhance recognition performance. In this work, we propose a Vertical Residual Autoencoder (VRAE) architecture designed for the image enhancement task in traffic surveillance. The method incorporates an enhancement strategy that employs an auxiliary block, which injects input-aware features at each encoding stage to guide the representation learning process, enabling better general information preservation throughout the network compared to conventional autoencoders. Experiments on a vehicle image dataset with visible license plates demonstrate that our method consistently outperforms Autoencoder (AE), Generative Adversarial Network (GAN), and Flow-Based (FB) approaches. Compared with AE at the same depth, it improves PSNR by about 20%, reduces NMSE by around 50%, and enhances SSIM by 1%, while requiring only a marginal increase of roughly 1% in parameters.

Keywords: License Plate Deblurring, License Plate Denoising, Lightweight, Generalize Preservation

1 Introdution & Related Work

In recent years, traffic scene understanding has become essential for intelligent transportation, autonomous driving, and surveillance. However, real-world traffic images are often degraded, reducing system reliability. Motion blur frequently occurs due to fast vehicle–camera motion and low-cost sensors with limited exposure, while advanced high-speed cameras remain costly [1–3]. Moreover, low-light conditions further introduce noise, glare, and low contrast, particularly in Southeast Asian cities with dense nighttime traffic and mixed lighting [4–6]. In addition, sensor limitations such as noise and dynamic range exacerbate the trade-off between blur and signal strength [7, 8]. As a result, these degradations severely impair downstream tasks like detection, recognition, and segmentation [9, 10]. To address this, image enhancement techniques, especially including denoiser and deblurrer, are widely studied. While traditionally treated separately, traffic images often suffer from both simultaneously. Deep learning methods have been explored including AE for compact representations, GAN for enhanced perceptual quality, and FB for explicit distribution learning that balances fidelity and diversity.

Traditional methods using AE have been widely adopted in such tasks due to their ability to learn compact latent representations and reconstruct images by filtering noise from signal [11]. Classical convolutional AE have shown promising results in removing additive noise, as demonstrated in DnCNN [7], which employed residual learning and batch normalization for Gaussian denoising. Similarly, convolutional encoder-decoder structures have been successfully applied in deblurring, such as DeblurGAN [2], which combined adversarial training with perceptual loss to recover sharp textures. This facilitates direct mapping from corrupted inputs to clean outputs, effectively addressing complex or spatially varying degradations. State-of-the-art deep generative methods, such as GAN and Diffusion, have been integrated into restoration pipelines. Specifically, DeblurGAN-v2 [12] introduced a multi-scale GAN framework that achieved both sharpness and stability in deblurring. Recently, FB models have emerged as a principled alternative for image restoration, capable of learning invertible mappings between degraded and clean domains while preserving density information [13].Notably, FB has been introduced as a scalable and stable training framework for conditional image generation, with its application to restoration explored in tasks demanding spatial consistency and smooth transitions, such as PnP-Flow [14, 15].

Research Gap: AE have been employed in traffic image enhancement due to their lightweight design, which is suitable for resource-constrained monitoring systems [16–18]. However, shallow encoders typically capture only low-level features and fail to recover mid- and high-level structures that are critical for fine detail restoration [19, 20]. GAN improve perceptual realism with sharper edges and more natural textures, but often distort pixel-level fidelity, hallucinate numbers, limiting their reliability for traffic applications where accurate recovery of details such as license plates and road markings is essential [21, 22]. FB models preserve pixel-level accuracy by optimizing explicit likelihood functions, yet their high computational complexity and memory consumption make them impractical for real-time deployment on edge devices [13, 23]. Therefore, a clear research gap exists in developing architectures that balance

computational efficiency with accurate detail preservation for traffic image enhancement. For such reasons, the main contributions of this work can be summarized as follows:

- We propose the **Vertical Residual Autoencoder (VRAE)** with an auxiliary block that vertically aggregates and refines embeddings for enhanced feature representation. Experimental results show that VRAE achieves impressive improvements across PSNR, SSIM, and NMSE, while maintaining competitive performance even with reduced model parameters.
- Through extensive experiments, we show that our approach not only encourages general information preservation in the early encoder layers but also outperforms conventional autoencoders in terms of image compression, as evaluated by entropy change
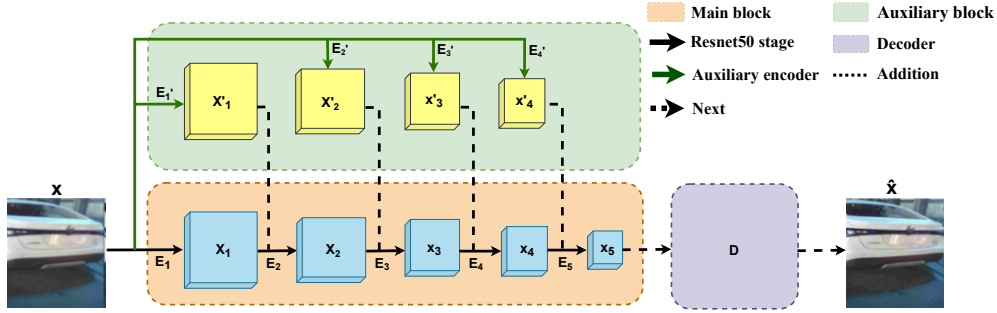

Fig. 1 VRAE aggregates the input embedding by passing it through the feature embedding block before forwarding it to the next encoder block.

2 Vertical Residual Autoencoder

Problem statement In traffic surveillance systems, vehicle images are often degraded by motion blur, sensor noise or adverse weather conditions, which obscure important visual details such as license plates, headlights, and road markings. Assume that all distortion is created by $\epsilon \sim N(0,1)$ and the task of vehicle image enhancement is defined as learning a one-to-one mapping $\mathbb{F}: \mathbf{x} \to \hat{\mathbf{x}}$ from a degraded input image $\mathbf{x} \in \mathbb{R}^{3 \times W \times H}$ to an enhanced output image $\hat{\mathbf{x}} \in \mathbb{R}^{3 \times W \times H}$, where $\mathbf{x} = \hat{\mathbf{x}} + \epsilon$.

2.1 Overall Architecture

VRAE comprises three components: (i) a parallel stack of *Auxiliary Encoders* $\{E_i\}_{i=1}^{N}$, each extracting hierarchical features directly from the input x; (ii) a continuous stack of *Main Encoders* $\{E_i'\}_{i=1}^{N-1}$ that further transforms the intermediate features and embeds them for residual injection into deeper stages; and (iii) a *Decoder D* that aggregates features from both streams to reconstruct the output $\hat{x}$. The two streams interact via element-wise additions at multiple stages, enabling vertical residual learning.

3

2.2 Auxiliary Block

The auxiliary path $\{E'_i\}$ acts as a *feature embedding module* that repeatedly extracts complementary, input-conditioned features to be fused with the main stream. Each block follows a shallow Conv–Norm–Activation structure and optionally includes pooling to summarize global context:

$$\mathbf{x}'_i = E'_i(\mathbf{x}), \quad i = 1, \dots, 5. \tag{1}$$

As illustrated in Fig. 1, the auxiliary block operates in parallel with the main encoder and generates an additional feature hierarchy $\{\mathbf{x}'_i\}$ from the degraded input $\mathbf{x}$. These features are progressively injected into the corresponding layers of the main block through element-wise fusion, allowing the model to enrich low- and mid-level representations with complementary cues.

2.3 Main Block

In our implementation, each main encoder E_i is instantiated as the i-th stage of the ResNet-50 architecture [24]. ResNet-50 employs a bottleneck design consisting of three convolutional layers: a 1×1 convolution for dimensionality reduction, a 3×3 convolution for spatial processing, and another 1×1 convolution for restoring dimensions. A skip connection adds the input to the output of the block, enabling better gradient flow and alleviating the vanishing gradient problem. To ensure a fair comparison, the same ResNet-50–based encoder backbone is also adopted for the baseline Autoencoder (AE), GAN, and FB models. This guarantees that the observed performance improvement arises from the proposed vertical residual aggregation mechanism rather than architectural capacity differences.

At each stage, the feature maps from the main and auxiliary encoder paths are fused by element-wise addition before being passed into E_i. Formally:

$$\mathbf{x}_i = E_i\big(\mathbf{x}_{i-1} + \mathbf{x}'_{i-1}\big), \quad i = 2, \dots, 5, \tag{2}$$

where $\mathbf{x}_{i-1}$ is the output of the $(i-1)$-th main encoder block and $\mathbf{x}'_{i-1}$ is the output of the $(i-1)$-th auxiliary encoder block. This additive fusion serves as an embedding-level integration mechanism: it incorporates complementary information without increasing feature dimensionality, offering a parameter-efficient alternative to concatenation. Such design, inspired by residual networks [24] and multi-branch models [25], not only stabilizes gradient flow but also enhances representational power, allowing the main encoder to capture hierarchical abstractions while continuously integrating localized cues from the auxiliary encoder.

2.4 Decoder

The decoder D reconstructs $\hat{\mathbf{x}}$ from $\mathbf{x}_N$ using a cascade of *transposed convolutional layers* followed by *ReLU* activations. Each transposed convolution progressively

upsamples the feature maps, restoring the spatial resolution:

$$\mathbf{y}_k = \text{ReLU}(\text{ConvTranspose2d}_k(\mathbf{y}_{k-1})), \quad k = 1, \ldots, 5, \quad \mathbf{y}_0 = \mathbf{x}_N, \quad \mathbf{y}_K = \hat{\mathbf{x}}. \quad (3)$$

We train the network with the *mean squared error (MSE)* loss only:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{BCHW} \sum_{b=1}^{B} \sum_{c=1}^{C} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(\hat{x}_{b,c,i,j} - x_{b,c,i,j} \right)^2, \quad (4)$$

which averages the squared reconstruction error over the entire batch and all pixels/channels.

3 Experiments

3.1 Dataset Preparation

From the CCPD dataset [26], 3,036 high-resolution vehicle images were collected and resized to 256×256 pixels. The dataset was split into training, validation, and test sets in a 70%–15%–15% ratio. Data augmentation, including rotations, was applied only to the training set, increasing it to 7,000 images. Low-resolution inputs were generated by adding discrete noise (values in $[0, 9]$ scaled by 0.1) and applying a 3×3 average pooling filter iteratively 10 times, simulating low-quality surveillance footage. This degradation process is not identical but bears similarity to the synthetic degradations used in SRMD [20], where combinations of blurring and noise are employed to simulate multiple low-resolution conditions.

3.2 Baseline choices

In vehicle image restoration, GAN have been widely applied to inpainting, deblurring, super-resolution, and artifact removal, achieving visually plausible and high-quality results [12, 21, 27]. In contrast, flow-based models such as RealNVP [28] and Glow [13] provide exact invertibility between image and latent spaces, enabling better preservation of fine-grained details. Although still underexplored in this domain, these characteristics make flow-based approaches particularly relevant to vehicle image restoration. To ensure a fair and controlled comparison, in this work we select both GAN and FB methods as baselines and design them with main encoder and decoder blocks consistent with our proposed method.

3.3 Settings

All models were trained using the Adam optimizer with an initial learning rate of 1×10^{-4} and a batch size of 16. Each model was trained for 100 epochs on an NVIDIA RTX 4070 GPU.

- **AE, FB, VRAE**: These methods utilize the MSE loss between the predicted and ground-truth HR images

5

Table 1 Quantitative comparison of GAN, FB, AE (2–5 sub encoders), and the proposed VRAE (2–5 sub-encoders). All metrics are evaluated on the **test set** and averaged across all samples. The best results are in bold, and the second-best are underlined.

Model	PSNR ($\uparrow$)	NMSE ($\downarrow$)	SSIM ($\uparrow$)	Params	FPS
GAN (Gen)	28.743	0.006	0.842	14.59M	188
FB (Gen)	27.093	0.008	0.831	25.87M	35
AE2	27.787	0.007	0.840	**0.375M**[1]	**411**
AE3	26.159	0.011	0.802	2.77M	289
AE4	29.404	0.005	0.864	14.59M	189
AE5	27.243	0.008	0.793	48.43M	119
VRAE2	<u>30.319</u>	<u>0.004</u>	<u>0.884</u>	<u>0.376M</u>	<u>399</u>
VRAE3	**31.052**	**0.003**	**0.898**	2.78M	194
VRAE4	30.246	0.004	0.880	14.61M	90
VRAE5	30.032	0.003	0.860	48.48M	45

[1]M denotes millions, the unit of the number of parameters.

- **GAN**: The GAN generator is trained with a hybrid loss combining pixel-wise MSE and adversarial BCE. While MSE ensures closeness to the ground truth, it often leads to overly smooth outputs. The adversarial loss encourages perceptual realism by pushing the generator to fool the discriminator. The overall generator loss is thus:

$$\mathcal{L}_{\mathrm{GEN}} = \mathcal{L}_{\mathrm{MSE}} + \alpha \cdot \mathcal{L}_{\mathrm{BCE}}$$

where $\alpha = 10^{-3}$ is a small scalar balancing the adversarial loss contribution, following the original setting proposed in [21].

4 Result

4.1 Comparison with State-of-the-Art

Table 1 presents the quantitative comparison among GAN, FB, AE (2–5 stacks), and the proposed VRAE (2–5 sub-encoders). Conventional GAN-based and FB-based approaches show moderate perceptual quality but achieve relatively lower PSNR and SSIM values compared to AE-based and VRAE-based models. Although stacked Autoencoders (e.g., AE4) can provide competitive results (PSNR = 29.404 dB, SSIM = 0.864), their performance is inconsistent across different stack depths, and some configurations (e.g., AE3 and AE5) degrade significantly. In contrast, the proposed VRAE consistently outperforms both AE and GAN variants in terms of reconstruction quality. Notably, VRAE3 achieves the highest PSNR (31.052 dB) and SSIM (0.898), while also maintaining a low NMSE (0.003). VRAE2 also delivers strong results (PSNR = 30.319 dB, SSIM = 0.884), ranking second overall. Despite slightly higher parameter counts than AE2, VRAE models demonstrate a favorable trade-off between accuracy and efficiency, with FPS values remaining competitive (e.g., VRAE2 at 399 FPS versus
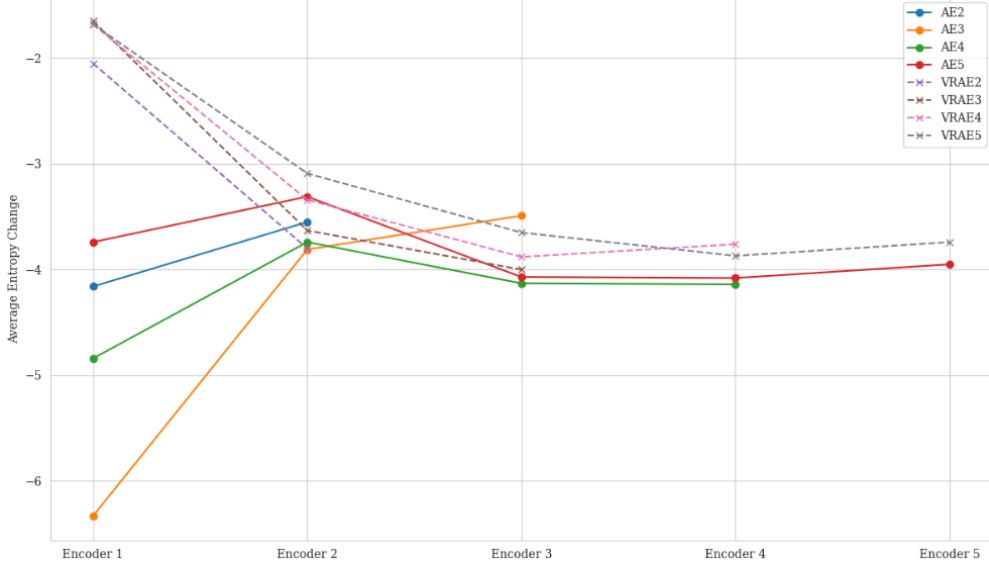
Fig. 2 Average entropy change across encoder layers of AE and VRAE. In the proposed architecture, entropy is computed from synthesized embeddings aggregated across encoder stages, capturing both local and global contextual dependencies.

AE2 at 411 FPS). When compared with AE at the same stack depth, VRAE improves PSNR by approximately 20%, reduces NMSE by around 50%, and enhances SSIM by about 1%, while introducing only a marginal increase of roughly 1% in the number of parameters. These results indicate that the VRAE architecture not only stabilizes performance across different depths but also surpasses conventional baselines in both reconstruction fidelity and computational efficiency.

4.2 Informative content preservation

According to [29], encouraging entropy preservation in the early layers promotes better generalization. This is consistent with our hypothesis Figure 4.2, where we observe that models preserving higher entropy in the first encoder layers tend to achieve better stability across subsequent layers. We first define the feature maps at encoder layer l as

$$F_l \in \mathbb{R}^{C_l \times H_l \times W_l}, \tag{5}$$

where C_l is the number of channels, and H_l, W_l are the spatial height and width of the feature maps. Formally, the entropy change between consecutive layers l and $l+1$ is defined as

$$\Delta H_l = H(F_{l+1}) - H(F_l), \tag{6}$$

7

where $H(F_l)$ denotes the Shannon entropy of the feature maps F_l. The Shannon entropy is defined as

$$H(X) = -\sum_i p(x_i) \log p(x_i), \tag{7}$$

where X is a random variable, x_i are possible outcomes (here corresponding to activation values in feature maps), and $p(x_i)$ is the empirical probability of observing x_i. According to the proof in [29], this entropy change can be approximated by the following proxy formulation:

$$\Delta H_l = (h - p + 1)(w - q + 1) \cdot \log |c_{11}|, \tag{8}$$

where h and w denote the height and width of the input feature map, p and q represent the height and width of the convolutional kernel, and c_{11} refers to the top-left coefficient of the convolution filter. This proxy captures how convolutional kernels affect the representational diversity of feature maps. A positive ΔH_l implies increased representational diversity, while a negative ΔH_l indicates information compression. In this work, we extend the formulation to operate at the encoder–block level instead of individual convolutional layers. Given an encoder block B_k containing m convolutional layers, we compute the average entropy change as:

$$\Delta H_{\text{avg}}^{(B_k)} = \frac{1}{m} \sum_{l \in B_k} \left[H(F_{l+1}) - H(F_l) \right]. \tag{9}$$

This block-wise aggregation smooths local fluctuations and enables a more stable comparison across architectures and depths. We now provide a deeper analysis of the entropy change patterns observed in Figure 4.2.

The first encoder layer exhibits the largest entropy discrepancy between AE and VRAE models. Specifically, AE3 shows a sharp entropy drop at Encoder 1, implying significant information loss at the start, whereas VRAE models begin with higher entropy change, indicating richer information preservation and supporting the hypothesis that maintaining early entropy prevents premature collapse.

As the encoding proceeds, AE models often display stronger and more irregular entropy fluctuations across layers, while VRAEs maintain a smoother and more consistent decline. For example, VRAE3 decreases gradually from –1.5 to –3.7 (approximately 2.2 units), compared to AE3's sharper drop from –6.3 to –3.5 (about 2.8 units). This smoother progression suggests that VRAEs handle information compression more effectively, mitigating the risk of losing key features too early.

In deeper layers (Encoders 4–5), both architectures tend to converge to similar entropy levels; however, AEs usually stabilize at lower entropy, reflecting more aggressive compression. VRAEs, in contrast, retain slightly higher entropy, indicating better preservation of representational richness. Overall, these results reinforce that sustaining entropy diversity in early stages contributes to stronger representation learning in VRAE.
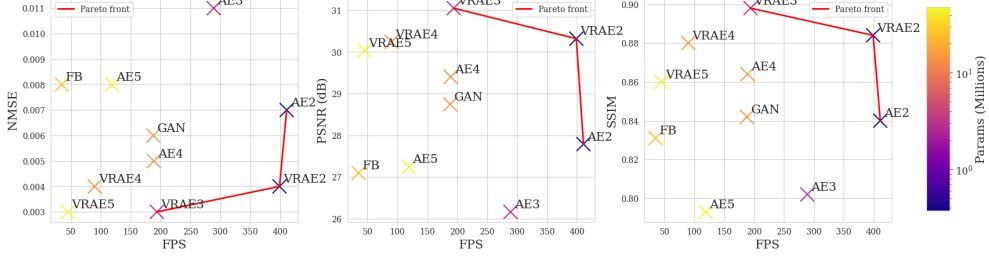
8

Fig. 3 Visualization of model performance trade-offs using the Pareto front. Each point denotes a model evaluated by reconstruction quality (NMSE, PSNR, SSIM) versus inference speed (FPS), with color indicating model size, i.e., the number of parameters. The red line represents the Pareto front, i.e., the set of models that cannot be improved in one objective without sacrificing another. This highlights the optimal balance between accuracy and efficiency, helping to identify the most suitable models depending on whether speed, quality, or a balanced trade-off is prioritized.

4.3 Comparative analysis of model trade-offs

To evaluate the efficiency of different models in vehicle image restoration, we compare their performance in terms of quality metrics (PSNR, SSIM, and NMSE) against inference speed (FPS). Figure 3 shows the distribution of models with the Pareto front highlighted in red, while the color scale represents the number of parameters in millions.

In the PSNR–FPS and SSIM–FPS plots, we observe that VRAE-based models (e.g., VRAE3 and VRAE2) consistently lie on the Pareto front, striking a balance between image quality and computational speed. In contrast, shallow autoencoders (e.g., AE2 and AE3) achieve high FPS but suffer from significantly lower quality scores, indicating that prioritizing speed alone comes at the cost of detail preservation. Meanwhile, GAN and flow-based (FB) models demonstrate moderate performance, but their relatively high parameter counts make them less efficient.

The NMSE–FPS trade-off further highlights these limitations: although AE2 reaches the highest FPS, it exhibits considerably larger reconstruction error, while deeper VRAE variants (VRAE3, VRAE2) maintain low NMSE values with acceptable inference speeds. These results underline the importance of designing models that preserve sufficient depth of encoding to capture meaningful representations while avoiding excessive parameter growth.

Overall, the Pareto front demonstrates that VRAE architectures achieve a favorable trade-off across accuracy, speed, and parameter efficiency, making them strong candidates for real-time vehicle image restoration on edge devices.

4.4 Limitations

While the proposed VRAE models demonstrate superior reconstruction quality across PSNR, NMSE, and SSIM, several limitations remain. As shown in Table 1, deeper variants (e.g., VRAE4 and VRAE5) yield consistent gains but incur substantially higher parameter counts and slower inference, revealing a clear trade-off between accuracy and efficiency. In addition, this work employs only base autoencoder and U-Net

architectures for clarity and accessibility, without exploring newer or more specialized designs that could further enhance performance. Finally, the evaluation is conducted on a single dataset, which may limit the generalizability of the findings. Future work will address these aspects by extending to more efficient architectures and diverse datasets.

5 Conclusion and future work

VRAE was proposed the Variational Residual Autoencoder (VRAE) for vehicle image restoration, which combines an auxiliary encoder and feature aggregation to enhance local detail and hierarchical representation. VRAE outperforms conventional Autoencoders, GANs, and flow-based models in PSNR, SSIM, and NMSE, while offering competitive inference speed. However, deeper VRAE variants improve quality at the cost of higher complexity, limiting edge deployment. Future work includes exploring lightweight designs, compression methods, e.g., pruning, quantization, distillation, and adaptive-depth architectures. VRAE can be further extended to real-time video restoration and other intelligent transportation tasks in future work.

References

[1] Pan, J., Sun, D., Pfister, H., Yang, M.-H.: Blind image deblurring using dark channel prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1628–1636. IEEE (2016)

[2] Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 8183–8192. IEEE (2018)

[3] Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 257–265. IEEE (2017)

[4] Chen, C., Chen, Q., Xu, J., Koltun, V.: Learning to see in the dark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018). CVPR 2018

[5] Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., Yang, J., Zhou, P.: Enlightengan: Deep light enhancement without paired supervision. arXiv preprint arXiv:1906.06972 (2019). arXiv 2019

[6] Zamir, S.W., Arora, A., Long, C., Sohail, A., Khan, S., Bhulla, M., Khan, F.S., Yang, M.-H., Khan, M.A.: Learning enriched features for real image restoration and enhancement. In: European Conference on Computer Vision (ECCV) (2020). ECCV 2020

[7] Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE Transactions on Image Processing **26**(7), 3142–3155 (2017)

[8] Anwar, S., Barnes, N.: Real image denoising with feature attention. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops (2020). ICCV Workshops 2020

[9] Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018). arXiv 2018

[10] He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017). ICCV 2017

[11] Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders, 1096–1103 (2008). ACM

[12] Kupyn, O., Martyniuk, T., Wu, J., Wang, Z.: Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8878–8887 (2019)

[13] Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. In: Advances in Neural Information Processing Systems, pp. 10215–10224 (2018)

[14] Martin, S., Gagneux, A., Hagemann, P., Steidl, G.: Pnp-flow: Plug-and-play image restoration with flow matching. International Conference on Learning Representations (ICLR) (2025)

[15] Zhu, Y., Zhao, W., Li, A., Tang, Y., Zhou, J., Lu, J.: Flowie: Efficient image enhancement via rectified flow. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13–22 (2024)

[16] Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning, pp. 1096–1103 (2008). ACM

[17] Xu, J., Li, Z., Guo, H., He, H.: Lightweight image super-resolution with information multi-distillation network. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 1237–1245 (2020)

[18] Jiang, K., Wang, Z., Yi, P., Wang, G., Lu, T., Jiang, J.: Deep distillation recursive network for remote sensing image super-resolution. Remote Sensing **10**(11), 1700 (2018)

[19] Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (2013)

[20] Zhang, K., Zuo, W., Zhang, L.: Learning a single convolutional super-resolution network for multiple degradations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3262–3271 (2018)

[21] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4681–4690 (2017)

[22] Wang, X., Yu, K., Dong, C., Loy, C.C.: Esrgan: Enhanced super-resolution generative adversarial networks. Proceedings of the ECCV Workshops (2018)

[23] Zhang, K., Van Gool, L., Timofte, R.: Invertible image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10085–10094 (2021)

[24] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

[25] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)

[26] Xu, Z., Yang, W., Meng, L., Lu, W., Huang, J., Zhou, J.: Towards end-to-end license plate detection and recognition: A large dataset and baseline. European Conference on Computer Vision (ECCV) Workshops, 255–271 (2018)

[27] Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1125–1134 (2017)

[28] Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. In: ICLR (2017)

[29] Meni, M.J., White, R.T., Mayo, M.L., Pilkiewicz, K.R.: Entropy-based guidance of deep neural networks for accelerated convergence and improved performance. Information Sciences **681**, 121239 (2024) https://doi.org/10.1016/j.ins.2024.121239