Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

CDRSHNET: VARIANCE-GUIDED MULTISCALE AND SELF-ATTENTION FUSION WITH HYBRID LOSS FUNCTION TO RESTORE TRAFFIC-SIGN IMAGES CAPTURED IN ADVERSE CONDITIONS

Milind Vijay Parse[1] and Dhanya Pramod[2]

(Received: 10-Nov.-2023, Revised: 2-Jan.-2024 and 23-Jan.-2024, Accepted: 25-Jan.-2024)

ABSTRACT

In challenging weather conditions, various visual impediments such as raindrops, shadows, haze and distortions from dirty camera lenses and codec errors adversely affect the quality of traffic-sign images. Existing methods struggle to address these issues comprehensively, necessitating an innovative approach to restoration. This paper introduces the Codec Dirty Rainy Shadow Haze Network (CDRSHNet) architecture, integrating self-attention (SA) and variance-guided multiscale attention (VGMA) mechanisms. SA captures global dependencies, enabling focused processing of relevant image regions, while VGMA emphasizes informative channels and spatial locations for enhanced representation. A hybrid loss function, combining Gradient Magnitude Similarity Deviation (GMSD) and Charbonnier loss, boosts image quality. When trained on a diverse dataset, CDRSHNet attains a remarkable 99.3% restoration accuracy, yielding an average SSIM of 0.978 and an average PSNR of 39.58 on the Real Image Dataset (RID). On the Synthetic Image Dataset (SID), the average SSIM is 0.963 and the average PSNR is 39.46. The proposed model significantly improves image clarity and facilitates precise interpretation.

KEYWORDS

Image restoration, Challenging weather conditions, Variance-guided multiscale attention, Custom loss function.

1. INTRODUCTION

Computer-vision tasks are pivotal in various applications, relying heavily on accurately interpreting visual data. However, these tasks encounter significant challenges when images are captured under adverse conditions or influenced by codec errors, rain, haze, shadows and blurry images due to dirty camera lenses. These adverse conditions introduce visual distortions, which diminish image quality and hinder visual elements' precise detection and classification. Real-world applications like automated driving systems and traffic-sign recognition systems require image restoration to recognize traffic symbols successfully. Concerning these challenges, traffic-sign image restoration becomes vital to enhance image quality and enable precise classification. Image restoration targets the improvement of degraded or corrupted images affected by factors, like noise, blur or compression, ultimately aiming to create high-quality images closely resembling the originals [2]-[3].

The usage of deep-learning methods for image restoration has gained significant popularity, as they can effectively learn to model the complex relationships between degraded images and their corresponding clean versions. Autoencoders, Convolutional Neural Networks (CNNs), Deep Residual Networks (ResNets), Recursive Neural Networks (RNNs) and Generative Adversarial Networks (GANs) are some of the popular image-restoration techniques [2]-[3].

This study proposes a deep learning-based approach for restoring traffic-sign symbols under five challenging conditions: images captured with a dirty camera lens, images captured under rainy and hazy environments, shadow-influenced images and codec error images. These issues reduce image sharpness, resulting in compromised contrast, color accuracy and blurs. Codec errors impact visual artifacts, like pixelation and color distortion. The proposed CDRSHNet incorporates a fusion of attention blocks and group normalization to enhance the quality of the restored images. Attention blocks selectively highlight essential features in the image, improving the efficiency of the network to distinguish between different types of traffic signs. Group normalization decreases the internal covariate shift, improving network stability and performance. Furthermore, we introduce a custom loss function that combines the

1. M. V. Parse is with Symbiosis International (Deemed University) (SIU), Pune, India. Email: parsemv@gmail.com
2. D. Pramod is with Symbiosis Centre for Information Technology (SCIT), Symbiosis International (Deemed University), Pune, India. Email: director@scit.edu

Charbonnier loss and Gradient Magnitude Similarity Deviation (GMSD) to enhance the quality of the recovered images. The GMSD evaluates the structural similarity between the original and restored images, whereas the Charbonnier loss penalizes significant errors in the restored image. The proposed approach is assessed using the images in the CURE-TSR dataset. The performance of the proposed network is assessed by peak signal-to-noise ratio (PSNR), mean absolute error (MAE) and structural similarity index measure (SSIM). These metrics are extensively used in the image-processing community to gauge the quality of the restored images and their analysis provides a quantitative assessment of the recommended approach's effectiveness. The experiments showcase that the proposed method outperforms current practices in restoring traffic-sign images under challenging conditions. The findings underscore the potential of deep learning techniques to enhance object restoration in complex real-world scenarios.

The paper is structured into several sections. Section two presents a literature review of existing works focusing on image-restoration techniques for challenging conditions, such as codec error, dirty lenses, rain, haze and images with shadows. Section three discusses our proposed architecture, including attention blocks, group normalization and custom loss function. Section four provides an in-depth analysis of the results obtained from the model, including the model loss and accuracy for both the training and validation datasets. Additionally, this section reports on the efficacy of the proposed approach in restoring traffic-sign symbols in challenging conditions. The performance of the proposed model is then assessed in Section 5 using a variety of measures, including peak signal-to-noise ratio (PSNR), mean absolute error (MAE) and structural similarity index measure (SSIM). These metrics are extensively used in the image-processing community to gauge the quality of the restored images and their analysis provides a quantitative assessment of the recommended approach's effectiveness. Finally, Section 6 provides the concluding remarks on the research work.

1.1 Our Contributions

The primary objective of this study is to implement a novel deep learning-based method designed explicitly for restoring traffic-sign symbols in the presence of five challenging conditions. Our proposed CDRSHNet model utilizes a fusion of attention blocks and group normalization to improve the quality of restored images. The attention blocks selectively emphasize essential features, enhancing the network's ability to distinguish between traffic signs. Moreover, group normalization assists in mitigating internal covariate shifts, improving stability and overall performance. We introduce a custom loss function that combines the Charbonnier loss and Gradient Magnitude Similarity Deviation (GMSD) to further enhance image recovery. The GMSD evaluates the structural similarity between images, while the Charbonnier loss penalizes significant errors. Together, these components contribute to the overall improvement in image quality. In our evaluations of the CURE-TSR dataset, we employ metrics, such as peak signal-to-noise ratio (PSNR), mean absolute error (MAE) and structural similarity index measure (SSIM). These well-established metrics provide a quantitative assessment of the effectiveness of our approach, clearly demonstrating its superiority over existing methods in restoring traffic-sign images under challenging conditions.

2. LITERATURE REVIEW

Image restoration aims to restore the original image from a damaged or noisy image. Over the years, various techniques have been developed to address this problem, each with advantages and limitations [4]. Some commonly used image-restoration techniques are filtering, statistical and model-based approaches, such as the blind-deconvolution method, inverse filter, Wiener filter and constrained least squares filter. These methods can be categorized as linear or nonlinear and are designed to mitigate the effects of noise and blur in the image. The techniques aim to recover the original image from a degraded version by applying mathematical operations that enhance its quality. The method chosen depends on the image-restoration problem and the type of degradation present in the image [4]-[5].

Diffusion-based methods use partial differential and variational restoration technology to propagate known information to the region to be repaired. This method works well for small-scale image damage, but cannot handle large missing areas or complex textures. Texture-based methods estimate information on corrupted regions using texture features in the original image and filling in the missing part with the best matching block. These methods are more suitable for severe damage in the image and can quickly

recover the texture details of the damaged image regions [6]. Researchers also use regularization-based methods to solve image-restoration problems. These methods aim to find a solution that satisfies constraints or regularization terms. Prior knowledge or assumptions about the underlying structure of the image are used to define the regularization terms [7]. Recently, various deep learning-based techniques have shown considerable success in image restoration. These methods use neural networks (CNNs) to learn the mapping between degraded images and their corresponding clean images. By training on large datasets consisting of degraded and clean images, these networks can learn to restore images accurately. A detailed discussion on deep-learning approaches to image restoration is given in [2].

Along with single neural network-based methods, generative adversarial networks (GANs) have also been used for image-restoration tasks. A discriminator and a generator are the two neural networks that comprise GANs. The generator network produces a restored image as an output from a degraded image as input. On the other hand, the discriminator network aims to differentiate between the cleaned and restored images. GANs can learn to generate highly realistic restored images by training these networks together. Overall, image restoration has advanced significantly over the years, with techniques ranging from linear filtering to deep learning-based methods [8].

G. Kwon et al. (2017) introduced a new and challenging dataset and comprehensively evaluated existing deep learning-based and traditional machine learning-based algorithms on this dataset. The CURE-TSR dataset includes images captured in real-world and simulated environments, including challenging weather, lighting and occlusions. The dataset includes various image degradations, such as blur, noise, shadows and codec errors, making traffic-sign recognition more difficult. Experimental results confirm that the performance of traffic-sign recognition algorithms varies significantly under different types of image degradations, with some algorithms being more robust than others [1]. S. Ahmed, U. Kamal et al. (2021) recommended a modular framework to detect and recognize traffic signs under difficult weather conditions. The offered solution implements a CNN for traffic-sign detection and recognition (TSDR) with prior image enhancement, comprising four modules: a challenge classifier, Enhance-Net (an image-enhancement module), a sign detection CNN and a classification CNN. Enhance-Net is trained to enhance traffic-sign regions specifically, enabling accurate detection instead of the entire image. The image enhancement component uses an encoder-decoder CNN architecture to augment input image quality by removing various image degradations, such as noise, blur and rain. The enhanced image is then fed into the region-based CNN detector for traffic-sign detection. The region-based CNN detector uses a two-stage approach, where, during the first stage, a set of candidate regions are generated, which are used during the second stage to classify a traffic sign or background. The image-enhancement block comprises five sub-blocks: rain, snow, haze, dirty lens and lens-blur removal blocks. Each block is designed to address a specific type of challenge. Each sub-block is applied to the input image only if the challenge classifier detects the corresponding type of challenge [9].

R. Huang et al. (2018) put forward an autoencoder-based architecture for restoring compressed images corrupted with codec errors. The autoencoder is first trained on a large set of clean images to learn a prior distribution of image patches. During the restoration process, the compressed image is first decompressed and then the autoencoder is used to predict the corrupted pixels in the decompressed image caused by the codec errors. To achieve this, the corrupted image is first divided into small overlapping patches and each patch is inpainted separately using the autoencoder. The autoencoder-based inpainting method uses the prior knowledge of the autoencoder to predict the missing pixels in each patch. Then, the patches are merged to produce the final restored image [10]. S. Jeon, H. Kim et al. (2019) proposed a solution for restoring compressed images distorted by the compression process. The authors advocated for an autoencoder-regularization approach to restore the images to their original quality. The proposed method involves training an autoencoder network on a large dataset of uncompressed images. The trained network is then used to regularize the restoration process for compressed images by imposing a constraint on the restored image to be similar to the output of the encoder part of the autoencoder [11].

K. Zhang et al. (2020) addressed the problem of concealing errors that occur during the compression of video streams. The authors introduced a two-stage method that uses deep-learning techniques to enhance the reconstructed video frames. The suggested method uses a CNN model to estimate the missing information in the corrupted video frames. This estimation is then used to generate a "confidence map" that indicates the reliability of the estimation at each pixel location. The next stage uses another CNN

to refine the estimated frames based on the confidence map [12]. M. Uricar et al. (2021) introduced a method for detecting image staining due to a camera lens in self-driving scenarios using a data-augmentation approach using a GAN. The recommended method uses a CycleGAN architecture to generate synthetic images that simulate different levels of camera-lens soiling with the help of two pairs of generator and discriminator networks to learn the mapping between clean and soiled image domains. The first generator inputs a clean image and outputs a corresponding soiled image. In contrast, the second generator inputs a soiled image and outputs a corresponding clean image. The two discriminator networks help ensure that the generated images are realistic and belong to their respective domains. The synthetic images generated by CycleGAN are then used to augment the original dataset of clean and soiled images. The augmented dataset trains a convolutional neural network (CNN) classifier that can accurately detect camera-lens soiling in autonomous driving scenarios. The authors use a dataset of 10,000 images from a camera attached to the front of a car. The dataset contains clean and soiled images, with varying levels of soiling caused by rain, snow and mud [13].

X. Li, B. Zhang et al. (2021) presented a solution for removing artifacts caused by contaminants (such as dust, dirt and moisture) on camera lenses in videos captured by moving cameras. The solution consists of three stages: detection, localization and removal. In the recognition stage, a deep CNN is trained to detect the presence of contaminants in each video frame. In the localization stage, a motion-analysis method is used to estimate the movement of the contaminants in each frame. In the removal stage, using the estimated motion information, a temporal filtering method is applied to remove the artifacts caused by the contaminants. To evaluate the suggested approach, the authors created a dataset of 30 video sequences with varying levels of contaminants captured by a moving camera. The dataset includes videos with rain, snow, dirt and clean reference videos. The authors additionally offer annotated ground truth and the extent of the contaminants in each frame. The method achieves PSNR of 35.37 and SSIM of 0.980 in stage one [14]. J. Mohd, S. M. Reyes et al. (2021) described a novel approach to detect dust particles in camera lenses mounted on moving robots. The suggested method also includes a technique for correcting the recorded or live image data by selectively removing the dust areas using an adaptive tiling-based approach. Dust particles are a significant issue for camera lenses in different disciplines, such as traffic-sign identification and geospatial data capture. The method aims to improve dust detection and correct image data by comparing consecutive frames captured by moving robots and removing the dust particles using the proposed technique while preserving the original data. Simulation results achieved 90-92% accuracy in removing the dust particles without affecting the actual data, which is a significant improvement [15].

H. Wang et al. (2021) suggested a deep CNN-based solution, "SRNet," to remove rain from a single image, integrating a structural residual learning framework with a residual block and a multiscale structure-extraction network. The residual block is designed to distinguish between the input and output images, while the multiscale structure-extraction network is employed to extract structure information from input images. This method achieved satisfactory performance on rain removal in a single image with PSNR=35.31 and SSIM=0.9448 [16]. Rainy images can adversely impact multimedia and computer-vision applications. CNN-based solutions have been employed to address this issue and eliminate rain from a single image. S. Li, W. Ren et al. (2019) presented a novel multi-task learning architecture that enhances performance by reducing excessive mapping between ground truth and output images. This architecture features a decomposition network that separates the rainy image into a clear background and multiple layers for the main component. During training, the composition structure is reconstructed to enhance the image quality by integrating clean input images and rain-related information. Experimental results demonstrated that this approach produces high-quality image restoration for synthetic and real images and surpasses contemporary techniques. Furthermore, the technique can be applied to other tasks, such as dust abstraction. Their method achieves PSNR of 33.7508 and SSIM of 0.9412 on the Rain50 dataset [17].

By analyzing urban video scenes, vision-based traffic analytic systems can significantly benefit Intelligent Transportation Systems (ITSs). However, vehicle detection and tracking can be challenging due to moving cast shadows, resulting in inaccuracies. M. U. Arif et al. (2022) conducted a comprehensive analysis of traditional and cutting-edge shadow identification and removal algorithms for traffic scenes based on 70 research papers published over the past 30 years. The study emphasizes the need for a hybrid approach combining traditional and well-known shadow-detection and removal

techniques before applying CNN-based vehicle-detection methods. The study also recommends using Highway I, II and III datasets for comparative evaluations; despite many CNN-based techniques for vehicle detection, moving cast shadow is still a challenge, necessitating pre-processing steps for accurate vehicle detection in traffic scenes [18]. Eliminating shadows from images can improve their visual appeal and has numerous applications in computer vision. Currently, based on deep-learning techniques, CNN is deemed the most efficient way to remove shadows. These methods can be trained using paired data of both the shadowed and clean images. Training CNN on unpaired data is typically favored in practice owing to the simplicity of data collection. In 2021, Z. Liu et al. proposed a new approach to shadow removal, referred to as the Lightness-Guided Shadow Removal Network (LG-ShadowNet), which employs unpaired data for training. The method comprises two CNN modules, with the first compensating for lightness and the second removing the shadow based on the information obtained from the previous module. It also introduces a loss function that utilizes the color before the existing data. Comprehensive experiments were conducted on popular datasets, such as the Image Shadow Triplets Dataset (ISTD), adjusted ISTD and USR. The proposed method performs well compared to available methods trained on unpaired data, with PSNR=25.92 and SSIM=0.909 [19].

The limited availability of paired shadow and ground truth images hinders the development of robust and large-scale shadow extraction algorithms. This limitation restricts the variety and size of shadow-removal datasets, making it difficult to train such algorithms. H. V. Le et al. (2020) presented a new shadow-removal technique that uses shadow and non-shadow regions from images for training to address this challenge. The approach uses an adversarial framework that incorporates a physical shadow-formation model. The method is handy for video shadow removal and achieves good results compared to the existing works [20]. H. Fan et al. (2019) examined the difficulties that current shadow-removal methods face in image segmentation and target recognition. They recommended a deep CNN composed of an encoder-decoder and refinement network to address these issues. The network predicts the alpha shadow scale factor and generates sharper edge information. A new image database (RSDB) is built and tested against various databases to evaluate the algorithm. Compared to other methods, the suggested algorithm significantly improves PSNR and SSIM metrics, producing sharper and shadow-free images that retain the image's color and texture close to the original image [21]. X. Hu et al. (2019) introduced Mask-ShadowGAN, a novel technique for removing shadows using unpaired data. The approach uses a deep-learning framework to generate a shadow mask based on an input shadow image. The generated mask is then employed to guide the process of shadow generation, incorporating cycle-consistency constraints. The framework is designed to simultaneously learn the generation of shadow masks to optimize overall performance. The effectiveness of this approach was evaluated on an unpaired dataset for removing shadows; it exhibited promising results across various experiments [22].

A unique image-fusion technique was introduced in 2021 by L. Ren et al. It improves the guided filter for better decomposition and artifact reduction. Before fusion, the contrast of viewable pictures is improved to address low light and noise. The authors divide the visible and infrared images vertically into sub-images, separate them into base and detail layers and use two fusion techniques. They also suggest a gradient-brightness criterion for adaptive output. Compared to earlier fusion techniques, experimental results show more significant performance in maintaining visible image details and improving infrared object clarity [23].

Q. Yang et al. (2022) tackled the problem of small-object detection. Feature Pyramid Networks (FPNs) represent a revolutionary technique that the authors suggested for enhancing small item detection. Small-Object Feature Enhancement (SOFE) and Variance-guided Region of Interest Fusion (VRoIF) are the two modules that makeup SV-FPN. To extract small-object features, SOFE improves finer-resolution level feature maps. In addition, VRoIF uses the variation in RoI features to determine the degree to which various RoI characteristics from various layers are all present. Ablation tests demonstrate the efficiency of SV-FPN on three open datasets, which achieved mean Average Precision (mAP) values of 41.5%, 53.9% and 38.3% on the KITTI, PASCAL VOC 07+12 and MS COCO 2017 datasets, respectively [24]. X. Yang (2020) conducted a comprehensive review of the attention mechanisms in computer vision. It discusses several attention mechanisms: self-attention, channel attention and spatial attention. The author emphasizes using attention mechanisms in image synthesis, object detection and picture classification. The author also investigates various attention-incorporating models, such as Transformer-based models and Convolutional Neural Networks (CNNs) and evaluates their benefits and drawbacks. In computer vision, the importance of attention mechanisms in improving

visual perception and task performance is substantial [25].

DehazeFormer distinguishes itself by showcasing its superior performance across multiple datasets, highlighting its efficacy in image dehazing. The authors introduce several enhancements to the Swin Transformer architecture, incorporating modifications, such as replacing LayerNorm and Gaussian Error Linear Unit **(**GELU) with RescaleNorm and ReLU, respectively. Furthermore, they propose a shifted window-partitioning scheme and a spatial information-aggregation scheme, contributing to the model's resilience and efficiency in dealing with dehazing tasks. Importantly, these improvements transcend the scope of DehazeFormer, as they offer minor, yet impactful, enhancements that can be applied to other networks. The model's features are exceptionally noteworthy regarding its performance on a substantial remote-sensing image-dehazing dataset [26].

A task-related contrastive network for single-image dehazing is introduced by W. Yi et al. (2023). It focuses on a compact autoencoder-like architecture with FEM and AFM, utilizing contrastive learning for improved performance. The proposed strategy involves effective data augmentation and a task-friendly embedding network. TC-Net outperforms existing methods, but limitations include reliance on common data-augmentation approaches and increased GPU-memory usage due to dynamic parameter updates in the training process [27].

To summarize, image-restoration techniques encompass various methodologies targeting specific image flaws. Diffusion methods excel at rectifying minor damages, while texture-based approaches effectively recover severe image flaws. Regularization methods rely on prior knowledge to meet image-structure constraints for restoration. Deep-learning techniques employing Convolutional Neural Networks (CNNs) efficiently restore images by learning from extensive datasets. GANs, with their generator and discriminator, produce realistic image restorations. Autoencoder-based methods predict and restore corrupted pixels in decompressed images, while autoencoder regularization aids in restoring compressed images. Deep-learning techniques efficiently enhance reconstructed video frames post-compression errors. GAN-based augmentation effectively detects lens staining in autonomous driving scenarios. Techniques for detecting and removing camera-lens contaminants in moving videos follow systematic stages. Rain removal relies on CNN-based approaches using structural learning and residual networks. Multi-task CNN architectures efficiently handle shadow removal with minimal mapping. Hybrid methods combine pre-processing with CNN-based detection for shadow detection. Innovations, like Lightness-Guided ShadowNet, use unpaired data for effective shadow removal. Techniques involving physical shadow models and CNN-based refinement effectively address shadow-removal issues. Mask-ShadowGAN, utilizing unpaired data, showcases a comprehensive approach to shadow removal. Image-fusion techniques employ guided filters and adaptive output for improved clarity in restorations. Collectively, these techniques contribute to the broad landscape of image restoration, each offering unique solutions for diverse image imperfections.

3. PROPOSED CDRSHNET ARCHITECTURE

The CDSRHNet architecture accepts colored noisy images with a resolution of 128x128x3 (height, width, RGB channels) and outputs a restored noiseless image. The network design has five levels with a cascade succession of convolutional operations. On each level, there are residual blocks. Two convolutional layers with Rectified Linear Unit (ReLU) activation and a group-normalization layer comprise each residual block. Small batch-size issue is addressed using group normalization. The feature map is normalized using group normalization, which divides the channels into groups and generates unique normalization statistics for each group based on the channel mean and variance. This strategy ensures independence from batch size when batch sizes are small. Padding is used throughout the model during convolutional operations to preserve the edge information of the image. The feature maps' spatial dimensions are preserved through padding, allowing the convolutional layers to capture important spatial data effectively. A skip connection is formed by element-wise summing the output of each residual block with its corresponding input. By propagating significant features from earlier blocks to later ones, this skip link aids in preserving those features. Figure 1 shows the proposed architecture of CDSRHNet.

The output of the final block in a level is passed as a skip connection to the self-attention module after the execution of all three residual blocks. The self-attention module captures global dependencies, so

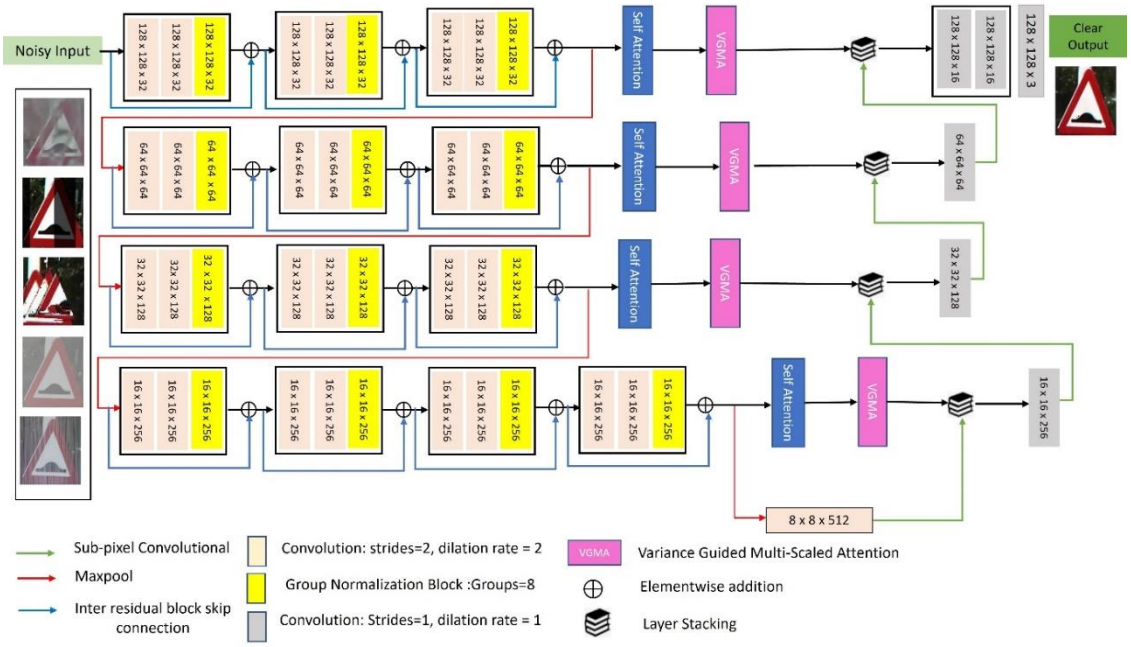


Figure 1. Proposed CDSRHNet model architecture.

that the network focuses on relevant image regions. The Variance Guided Multi-Scaled Attention (VGMA) module continues to process the output of the self-attention module and improves the image representation by emphasizing informative channels and spatial locations. Also, skip connections are used in deep neural networks to solve the vanishing-gradient problem, because the gradient signal has a shorter path to follow during backpropagation. The spatial dimensions of the feature maps produced at the end of the third residual block in each level are down-sampled using pooling layers. This downsampling reduces the parameter count in the network and increases its insensitivity to small spatial translations. Additionally, pooling layers increase the neurons' receptive field, which improves the network's ability to record spatially invariant data. Equation (1) determines the output down-sampled feature map of a tensor.

$$D = \frac{(I+2P-K)}{S} + 1 \tag{1}$$

where, I is the input feature map, P is the padding, K indicates convolutional kernel and S is the stride of convolutional operation.

The dilatation process introduces gaps or spaces between the kernel elements, allowing neurons to perceive a larger area of the input-feature map. The dilation rate determines the size of the gap in the kernel. The proposed method has a dilation rate of (2,2), which increases the receptive field without increasing the parameters. The mathematical formula for a dilated convolutional operation with dilation rate d can be expressed as follows with the help of Equation (2).

$$y=\sum_{k=0}^{k=k-1} [i + (k * d)] * w[k] \tag{2}$$

where, i is input feature map, y is output feature map, w is the kernel of size k and d is the dilation rate.

The spatial dimensions of output-feature map y are identical to those of input-feature map i, but a factor of d expands each neuron's receptive field. The final bottleneck layer is 8x8x512 in dimension and represents a compressed-input representation. The most important and pertinent data from the preceding layers is collected in this layer. The expansion path uses subpixel convolution to improve efficiency, capture global dependencies and retain spatial information. The outputs from the VGMA module and the preceding level's subpixel convolution layer are combined at each level of the expansion path. Concatenation enables the model to recover spatial data that was lost during down-sampling.

3.1 Self-attention

The self-attention module, illustrated in Figure 2, captures long-range dependencies and selectively attends to relevant information within the input sequence or feature map. It requires three parallel 1x1 convolutions, to produce the query(Q), key(K) and value(V) vectors. While the key vectors represent all the other positions or elements, the query vector represents the current position or element being

attended to. The attention mechanism establishes position weights through the element-wise dot product of query and key vectors, followed by obtaining attention weights using a Softmax function.

The value vectors (V) containing features related to each position are multiplied by the respective attention weights. The self-attended feature representation is obtained by combining the resulting weighted value vectors using a summation operation or weighted average. The self-attended characteristics are then refined and transformed using a 1x1 convolution. Self-attention effectively captures the intra-correlation of an input matrix X = [x1, x2,⋯, xn]$\in \mathbb{R}^{(d\times n)}$, where d is the dimensionality of the input vectors and n is the number of vectors in the sequence. In this self-attention, Q, K, V and X are kept equal and the self-attention is computed as in Equation (3).

$$\text{SA} = Softmax\left(\frac{X^T X}{\sqrt{d}}\right) * X \tag{3}$$

here, SA is the self-attended feature representation calculated based on the attention mechanism which captures the intra-correlation or dependencies between each pair of input vectors, reflecting the significance of each vector in understanding or representing other vectors in the sequence.

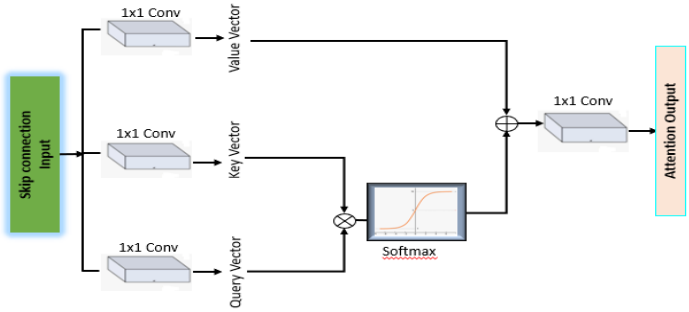


Figure 2. Self-attention architecture.

3.2 Variance-guided Multiscale Attention

The proposed approach combines channel attention with variance and spatial-attention modules to improve image-reconstruction quality while reducing network depth and computational complexity. By emphasizing channels with high variance, which indicates the presence of important or distinctive features, the channel-attention mechanism based on variance enables us to capture channel variability. The network can adjust its weighting based on variance by prioritizing informative channels and suppressing noisy or less informative channels. Each spatial position in the feature maps is taught to receive weight from the spatial-attention module, reflecting the significance of that region during feature fusion. This allows the network to adaptively adjust the contribution of each spatial location based on its significance, enhancing the representation of local and global details.

We perform an element-wise summation between the weighted feature maps obtained from the channel-attention mechanism and the spatial-attention weights to fuse the outputs of the channel attention and spatial-attention modules. By combining the informative channel-level weights with the spatially adaptable weights, this fusion process maximizes the benefits of both methods. This novel approach presents practical solutions to address the specific challenges outlined in this paper, offering advanced capabilities for various image-enhancement tasks. Figure 3 shows the architecture in more detail.

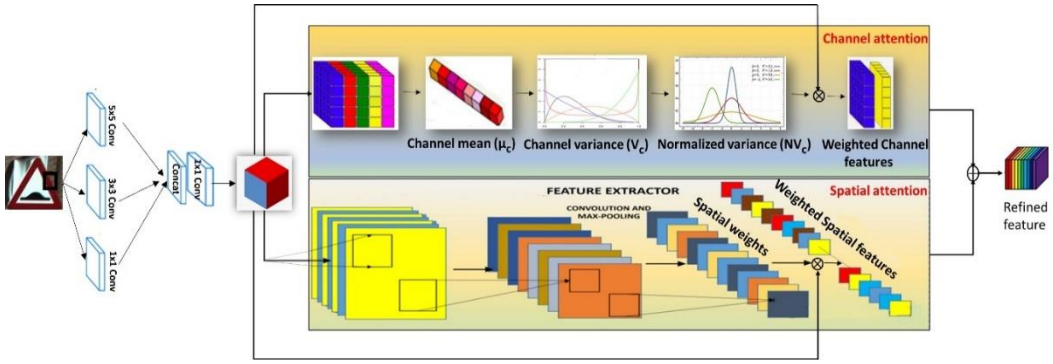


Figure 3. Variance-guided multiscale attention architecture.

3.2.1 Channel Attention

The channel-attention mechanism offers several benefits when variance is used to adjust the weights of all channels. First, emphasizing channels with high variance, which denotes the presence of significant or distinctive features, enables the capture of channel variability. The network can prioritize channels that contribute more to the overall task by using this adaptive weighting based on variance, which enhances performance. Incorporating variance-based weights further improves channel-noise robustness by minimizing the impact of noisy or uninformative low-variance channels. In addition to increasing the network's computing performance, this reduces noise and redundant data. The following are the steps involved in extracting significant channel-level features.

3.2.1.1 Variance Computation: The variance (Vc) for each channel (c) is calculated for a feature map (F) with dimensions H x W x C, where H represents height, W signifies width and C denotes the number of channels. The formula for Vc is expressed as follows:

$$\text{Vc} = \frac{1}{(H*W)} * \sum_{i=1}^{H}\sum_{j=1}^{W}(F(i,j,c) - \mu_c)^2 \quad (4)$$

Here, H and W denote the height and width of the feature map, respectively; i and j are indices representing spatial positions within the feature map; F(i, j, c) is the value of the feature map at position (i, j) in channel c and μ_c represents the mean value of channel c along a particular dimension in the input-feature map.

3.2.1.2 Variance Normalization: Channel-specific variances are normalized to obtain weights that sum up to 1. The calculation of Normalized Variance (NVc) is articulated as follows:

$$\text{NVc} = \frac{Vc}{\sum Vc} \quad (5)$$

3.2.1.3 Weight Adjustment: To adjust the importance of each channel, normalized variances are applied as weights. The weight assigned to a channel increases with its variance. Equation (6) calculates the adjusted weighted feature map (W_{oc}) for spatial position (i, j) and channel c:

$$W_{oc}\ (i, j, c) = \text{NVc} * F\ (i, j, c) \quad (6)$$

Each member of the matrix F is represented by F (i, j, c), denoting the value at the position (i, j) in channel c of the feature map.

3.2.2 Spatial Attention

Integrating spatial attention with variance-based channel attention offers a comprehensive approach to identify spatial dependencies and channel-wise variability, improving feature representation. Spatial attention improves feature representation by considering the spatial dependencies within the feature maps, whereas variance-based channel attention concentrates on capturing inter-channel relationships and significant features. Spatial attention captures contextual information and fine-grained features by modeling the relationships between spatial locations and allocating weights accordingly. This leads to improved feature discrimination and improved model performance.

Consider $\int(\cdot)$ being the function to compute the spatial-attention weights denoted by SW(i,j), for an original feature map represented by O(i,j) at each spatial location (i, j). This involves a sequence of mathematical operations applied to the original feature map. After these operations, a nonlinear activation function is applied in conjunction with a weighted convolution. The spatial weights SW(i,j) at spatial location (i,j) are given by Equation (7).

$$\text{SW(i,j)} = \int(\text{O(i,j)}) \quad (7)$$

3.2.3 Adjusting Contribution by Feature Fusion

The weighted feature maps from the channel-attention mechanism created from Equation (6) are multiplied element-wise with the spatial-attention weights acquired from Equation (7) in the previous step to adjust the contribution of each spatial location during feature fusion.

$$\text{Woc}_{\text{Adjusted}}\ (i, j) = \text{Woc}(i, j) * \text{SW}(i,j) \quad (8)$$

$\text{Woc}_{\text{Adjusted}}$ (i, j) in Equation (8) represents the adaptively adjusted channel weights at each spatial location (i,j).

3.3 Hybrid Loss Function

This study proposes a novel custom loss function that combines the GMSD (Gradient Magnitude Similarity Deviation) and Charbonnier loss functions. Compared to the conventional single loss function, which can only handle one type of issue, the proposed loss function can simultaneously address noise, blurriness and sharpness problems. X. Zhu et al. [28] employed an integrated loss function made up of MSE, VGG19-based perceptual loss and Novel Quality Loss inspired by IQA (Image Quality Assessment) metric and GMSD to create an image that is in line with human vision, utilizing the GAN network.

The Charbonnier loss and the VGG16 perceptual loss were both employed by B. Wu et al. [29]. Their results outperformed previous-study methods in producing images with intricate features and sharp edges. In order to reduce noise in CT-scan images, B. Gajera et al. [30] proposed an enhanced GAN that combined Charbonnier loss and VGG19 perceptual loss. The authors found that this network significantly improves denoising performance by bringing soft-tissue noise levels closer to those of a Normal Dose CT (NDCT) scan. In Equation (9), L_M shows the final loss function of the proposed model. The weights for each loss term are α and β, respectively.

$$L_M = \alpha L_{GMSD} + \beta L_{Charbonnier} \tag{9}$$

Each weight has an initial value of 0.5 at the beginning of the training. The weights are dynamically modified after each epoch throughout the training based on their relative contribution to the validation loss. This method optimizes the model to minimize loss functions based on their relative importance in the overall validation loss, improving overall performance and achieving a better balance between reconstruction accuracy and perceptual quality.

3.3.1 Gradient Magnitude Similarity Deviation (GMSD)

To measure the similarity between two images, the Low-Resolution Image (LRI) (a distorted image) and the High-Resolution Image (HRI) (a ground truth image), W. Xue et al. created the Gradient Magnitude Similarity Deviation (GMSD) in 2014. GMSD evaluates how the global variation of the gradient-based local-quality map is used. GMSD is highly consistent with how people perceive the quality of an image and is computationally efficient. GMSD is robust to various visual artifacts, such as noise, blur and compression [31].

GMSD is calculated using the following steps:

a. Gradient Magnitude Calculation: Use the Prewitt filter to compute horizontal and vertical gradient magnitudes for the Low-Resolution Image (LRI) and the High-Resolution Image (HRI).
b. Gradient Magnitude Similarity (GMS): Calculate the Gradient Magnitude Similarity (GMS) between LRI and HRI by considering the relationship between their gradient magnitudes.
c. Mean GMS: Compute the mean GMS values across the entire image for LRI and HRI.
d. GMSD Calculation: Determine the GMSD, representing the deviation in gradient magnitude similarity, by assessing the differences between individual GMS values and their mean values.

3.3.2 Charbonnier Loss

The Charbonnier loss is a smooth approximation of the Huber loss that preserves its robustness while being more differentiable and simpler to optimize. Huber loss has a non-smooth quadratic to linear transition, which makes gradient-based optimization challenging. The epsilon parameter in the Charbonnier loss regulates the smoothness of the transition from the quadratic to linear region. The Charbonnier loss acts like the L2 loss (MSE) when the epsilon is small and the L1 loss (MAE) when the epsilon is large. For this experiment, the epsilon was set to 0.001. For image-restoration tasks, like denoising, deblurring and super-resolution, Charbonnier loss is better suited, because it penalizes substantial errors less severely than the MSE loss. Compared to MSE loss, Charbonnier-loss outcomes are more visually pleasing. MSE loss function may produce fuzzy edges. This occurs, because the Maximum Likelihood Estimator (MLE) for MSE is the arithmetic mean, while the edges in images typically have two unique modes or values, making them bimodal. Charbonnier loss is expressed by Equation (10).

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

$$L(y, y') = \frac{1}{n}\sum\sqrt{(y - y')^2} + \varepsilon^2 \quad (10)$$

where, y and y' are the original and predicated images, respectively, while Ɛ is a small constant.

For minor errors, the square root term in the formula behaves like an L1-norm penalty; for more enormous errors, it behaves like an L2-norm penalty.

3.3.3 Optimizer

"Reduce LR on Plateau" technique at run time dynamically decreases the learning rate by a predetermined factor when validation loss of the model stops improving after a certain number of iterations. For this study, initial learning rate was set to 1e-2 and learning rate reduction factor was 10% with a waiting of 3 epochs. At around 73% accuracy, the model started to show prolonged improvement, after which the learning rate was changed to 1e-3; finally, at 98.16% accuracy, it was changed to 1e-4.

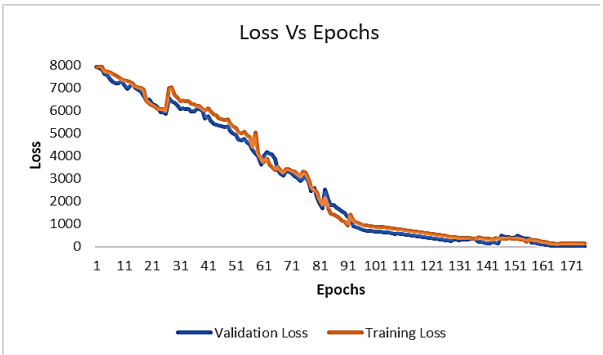
3.4 Dataset Preparation

The dataset used for this research is derived from the CURE-TSR dataset [1]. The CURE-TSR dataset contains over two million images of traffic-sign symbols cropped from the CURE-TSD video dataset. The CURE-TSR dataset is introduced to analyze and evaluate the efficiency of algorithms under challenging conditions. The CURE TSR dataset contains 14 different sign types and 12 challenging conditions. For this study, we created two subsets of the CURE TSR dataset. The first subset is the real-image dataset (RID) and the second is the synthesized-image dataset (SID). The real-image dataset (RID) contains 84K images with five challenging conditions and 14 sign types. The SID dataset contains 16800 images and is mainly used to evaluate the model and compare the results with the RID test dataset.

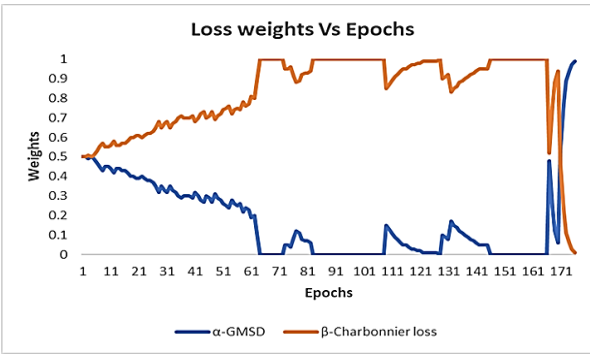
The RID dataset is partitioned into Training RID and Testing RID sub-sets with a ratio of 80:20. The training dataset contains 67200 images and the testing dataset contains 16800 images. Similarly, the test SID dataset contains 16800 synthesized images. This study considered 14 traffic signs with five visual challenges: codec error, dirty lens, rain, shadow and haze images.

4. TRAINING AND VALIDATION LOSS

Graph 1 shows the training *versus* validation loss for the proposed model developed to restore traffic-sign images. The x-axis signifies the number of epochs (i.e., the number of times the model has been trained on the entire dataset). At the same time, the y-axis represents the loss, which measures the distinction between the predicted output of the model and the actual output. The training loss and validation loss were relatively high during the initial training process, indicating that the model was not accurately predicting the restored images. However, as the number of epochs increased, the training loss and validation loss decreased gradually. This suggests that the model was learning to restore the images better over time.



Graph 1. Training loss and validation loss.



Graph 2. Values of alpha and beta.

Initially, the training loss was 7935 and the validation loss was 7922. After 178 epochs, the training loss and validation loss dropped to 186 and 139, respectively, which shows that the model has learned the weights much more accurately. Further, there was no improvement in the loss. This encouraging outcome indicates that the model can restore traffic-sign images with high accuracy and could be used for practical applications.

4.1 Loss Tuning Parameters-Alpha and Beta

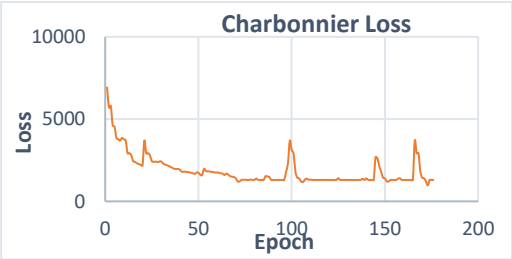
As shown in Equation (9), the alpha and beta weights are the weights of the GMSD and Charbonnier losses, respectively. Updating these weights gives more weight to the more critical loss function and less weight to the less important loss function. Suppose that the ratio of the validation loss to the training loss decreases. In that case, the model performs better on the validation sub-set than on the training sub-set, which may indicate that the Charbonnier loss contributes more to the total loss. In this case, the beta weight is increased and the alpha weight is decreased; so, the model gives more weight to the Charbonnier loss.

On the other hand, if the ratio of the validation loss to the training loss increases, this indicates that the model performs worse on the validation sub-set than on the training sub-set. This could happen if the model overfitted to the training sub-set, which means that it fits the noise in the training sub-set rather than the underlying pattern. In such cases, the GMSD loss, which measures the structural similarity between the ground truth and the predicted image, becomes more critical, as it encourages the model to generate visually similar images to the ground truth rather than just fitting the training-set noise. Therefore, increasing the alpha weight and decreasing the beta weight ensure that the model gives more weight to the GMSD loss and tries to generate visually similar images to the ground truth rather than just fitting the training-set noise.

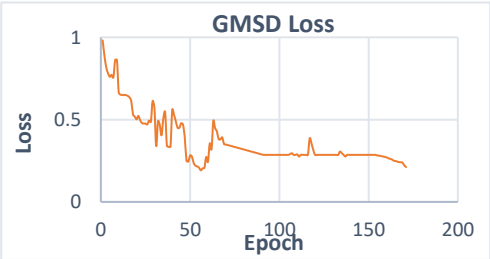
The decay parameter controls the rate at which the weights are updated, which increases or decreases the weights in each epoch. The decay parameter is a ratio of validation loss to training loss. A small constant (epsilon) is added to prevent alpha and beta from becoming absolute zero. If either alpha or beta becomes zero, the corresponding loss function will not contribute to the overall loss and the model will not be able to learn any further from that loss function. The alpha and beta factors are clamped between a minimum value (epsilon) and a maximum value (1.0). The sum of alpha and beta is fixed at 1.0; therefore, the distribution graph of alpha and beta looks like a mirror image of each other, as shown in Graph 2.

4.2 Comparing Hybrid Loss Function and Single Loss Functions

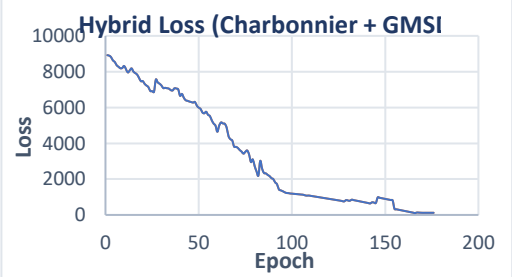
As discussed in sub-section 3.2, to demonstrate the effectiveness of our novel custom loss function that combines the GMSD (Gradient Magnitude Similarity Deviation) and Charbonnier loss functions, we conducted ablation experiments, as shown in Graphs 3-6; the proposed hybrid loss has optimal convergence of 139 after 178 epochs. Charbonnier, GMSD and MSE have optimal losses of 1206, 1238 and 0.211, respectively. The proposed hybrid loss also has smooth progression compared to other individual loss functions.



Graph 3. Charbonnier loss.



Graph 4. GMSD loss.



Graph 5. Hybrid loss (Proposed).

Graph 6. MSE loss.

During the experiment, we also captured quantitative matrices PSNR, SSIM and MAE of the model with each of the loss functions; as shown in Table 1. The proposed hybrid function exceeds all the parameters except for RMSE, where MSE loss has the least value of 105.651 against 106.151.

Table 1. Loss-function comparison.

Loss Function	PSNR	SSIM	MAE
Hybrid	41.024	0.9726	0.040
MSE	31.940	0.9169	0.090
Charbonnier	29.091	0.8628	0.127
GMSD	28.315	0.8390	0.164

5. MODEL PERFORMANCE EVALUATION

The model's performance has been evaluated based on the image-processing domain's three most frequent evaluation metrics. These are: The Mean Absolute Error (MAE), Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM).

5.1 Mean Absolute Error (MAE)

MAE measures the average absolute difference between the restored and ground truth images. Lower MAE indicates a better resemblance of the input image with the ground truth image. Equation (11) is used to calculate MAE.

$$\text{MAE} = \frac{1}{N} * \sum |y_i - x_i| \tag{11}$$

where, N represents the total number of pixels in the restored and ground truth images, y_i and x_i are the pixel values at the corresponding positions in the restored and ground truth images. The summation is taken over all pixels in the images. The absolute value (|.|) ensures that the differences are positive and the average is calculated by dividing the sum by N. Table 2 shows the MAE values when the model is tested on the total test images in individual challenging conditions.

Table 2. MAE for each challenging condition in the test dataset.

Challenging Condition	No. of Signs	Number of Images	MAE value for Test RID			MAE value for Test SID		
			Min.	Max.	Avg.	Min.	Max.	Avg.
Codec Error	14	2800	0.0140	0.046	0.022	0.0146	0.048	0.022
Dirty Lens	14	2800	0.0026	0.038	0.018	0.0028	0.044	0.018
Rainy Images	14	2800	0.0034	0.042	0.020	0.0036	0.042	0.020
Shadow Images	14	2800	0.0012	0.032	0.016	0.0014	0.036	0.018
Haze	14	2800	0.0038	0.042	0.026	0.0038	0.042	0.026

5.2 Peak Signal-to-Noise Ratio (PSNR)

PSNR calculates the relationship between the signal's highest possible power and the noise's power, which affects its accuracy in representing the signal. The equation that defines PSNR is Equation (12).

$$\text{PSNR} = 10 \log \left(\frac{R^2}{MSE}\right) \tag{12}$$

where, R is the maximum pixel value of the image, which is 255 for a coloured image. MSE is the mean squared error between the original image and the restored image. Equation (13) describes the calculation of MSE between the y-original image and y' – restored image for n-data points.

$$\text{MSE} = \frac{1}{n}\sum_1^n (y - y')^2 \tag{13}$$

Table 3 shows the PSNR values when the model is tested on the total test images in individual challenging conditions.

Table 3. PSNR for each challenging condition in the test dataset.

Challenging Condition	No. of Signs	Number of Images	PSNR value for Test RID			PSNR value for Test SID		
			Min.	Max.	Avg.	Min.	Max.	Avg.
Codec Error	14	2800	36.45	39.89	38.17	34.56	40.02	37.86
Dirty Lens	14	2800	38.23	41.57	39.90	37.95	40.86	38.54
Rainy Images	14	2800	39.78	42.58	41.18	39.89	43.53	39.44
Shadow Images	14	2800	42.68	44.34	43.52	42.34	45.46	43.34
Haze	14	2800	32.84	36.74	34.79	31.08	37.98	34.86

5.3 Structural Similarity Index Measure (SSIM)

SSIM is a metric used to measure the similarity between two images. SSIM is commonly used to evaluate how well a restored image matches the original image. The mathematical formula for SSIM is given in Equation (14).

$$\mathrm{SSIM}(x,y) = \frac{(2 * \mu_x * \mu_y + C1) * (2 * \sigma_{xy} + C2)}{(\mu_x{}^2 + \mu_y{}^2 + C1) * (\sigma_x{}^2 + \sigma_y{}^2 + C2)} \quad (14)$$

where, x and y are the two images being compared, μ represents the mean of an image, σ represents the standard deviation and σ_{xy} represents the covariance of the two images. C1 and C2 are constants that stabilize the division by the weak denominator. Table 4 shows the SSIM values when the model is tested on test images in individual challenging conditions of both test datasets.

Table 4. SSIM for each challenging condition in the test dataset.

Challenging Condition	No. of Signs	Number of Images	SSIM value for Test RID			SSIM value for Test SID		
			Min.	Max.	Avg.	Min.	Max.	Avg.
Codec Error	14	2800	0.716	0.980	0.977	0.782	0.979	0.963
Dirty Lens	14	2800	0.834	0.982	0.979	0.824	0.972	0.968
Rainy Images	14	2800	0.886	0.987	0.976	0.842	0.982	0.976
Shadow Images	14	2800	0.898	0.988	0.982	0.912	0.984	0.978
Haze	14	2800	0.742	0.926	0.934	0.788	0.965	0.924

Overall, the SSIM values suggest that both datasets contain restored images that are highly similar to the original images, with some variations in the levels of similarity across different restored images in each dataset. Thus, the proposed method provides better image restoration even if the traffic sign is captured in challenging conditions.

5.4 Performance Enhancement with Self-attention and Variance-guided Multiscale Attention

An ablation experiment has been performed to evaluate the contribution of self-attention and VGMA in the proposed architecture. Results have been gathered using a real image dataset (RID), as shown in Table 5. Initially, the model was built without any attention mechanism where average PNSR and SSIM were obtained as 30.35 and 0.4525, respectively. In the next iteration, we added only a self-attention module and achieved an average PNSR of 34.46 and an average SSIM of 0.6642. Later on, the model was created using only VGMA and got an average PNSR of 38.42 and an average SSIM of 0.8288. Finally, both self-attention and VGMA modules were added to the model to obtain an average PNSR of 41.024 and an average SSIM of 0.9726. This experiment demonstrates that the proposed model performs better after including both attention modules.

Table 5. Contribution of self-attention and VGMA.

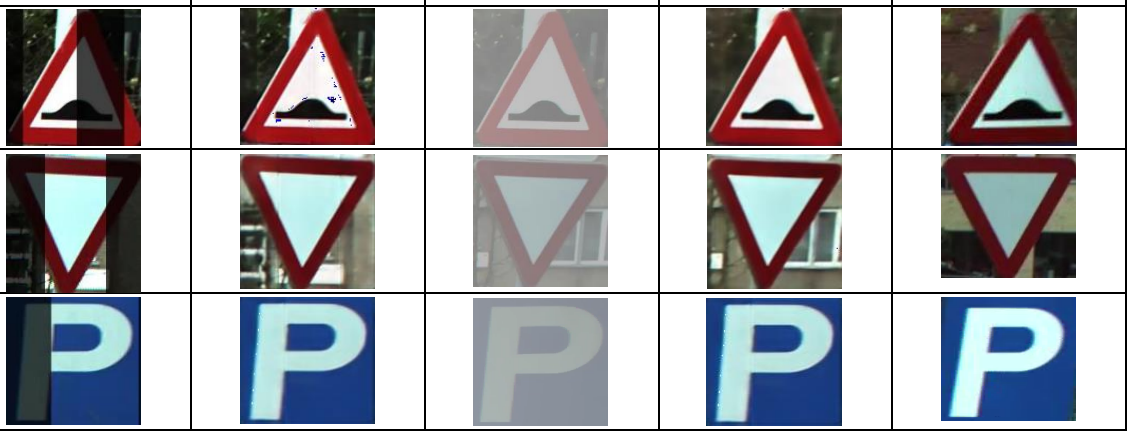
	No attention		Self-attention		VGMA		Self + VGMA	
Challenge Type	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Codec Error	26.36	0.1235	29.53	0.8109	34.47	0.8094	39.89	0.9800
Dirty Lens	31.65	0.2833	38.98	0.4724	42.45	0.8012	41.57	0.9820
Rain	34.58	0.6285	37.87	0.5572	39.71	0.7243	42.58	0.9870
Shadow	28.88	0.5300	38.76	0.8613	42.94	0.8918	44.34	0.9880
Haze	30.28	0.6970	27.16	0.6191	32.55	0.9175	36.74	0.926
Average	**30.35**	**0.4525**	**34.46**	**0.6642**	**38.42**	**0.8288**	**41.024**	**0.9726**

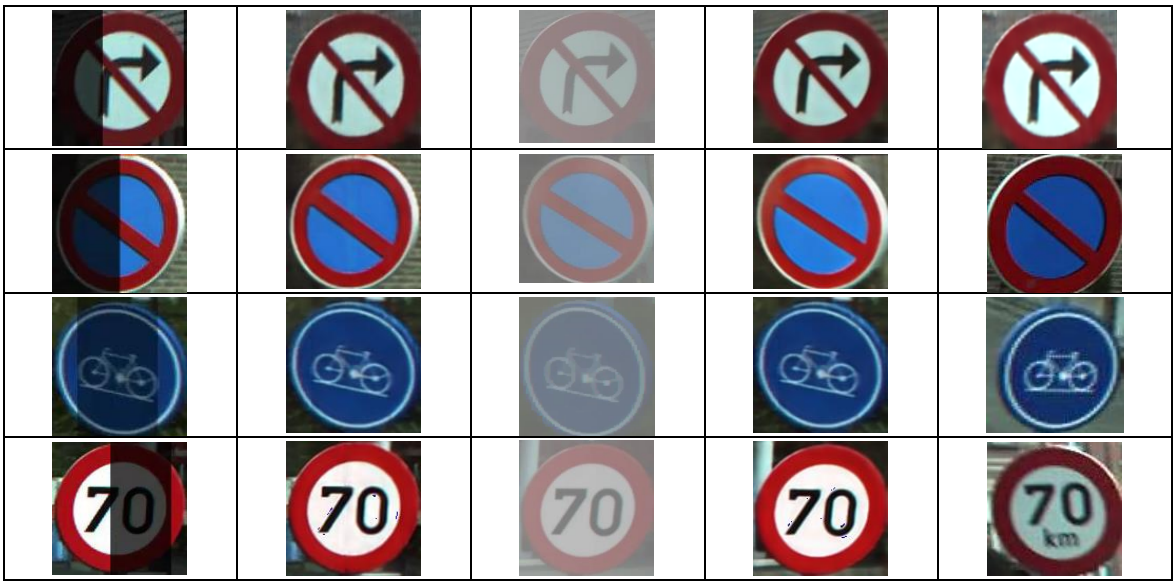
Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

5.5 Experimental Results

Figures 4-8 show the model results for a few test images in each challenging condition: Codec Error, Dirty Lens, Rain, Shadow and Haze. As shown in the figure, the input image in the first column(a) for every table is effectively restored by the model image shown in column (b), which is almost similar to the ground truth image shown in column (c).

Figure 4. Codec Error.		Figure 5. Dirty Lens.		Figure 6. Rainy Images.		Ground Truth Images (c)
I/P Image (a)	Restored Image (b)	I/P Image (a)	Restored Image (b)	I/P Image (a)	Restored Image (b)	
						

Figure 7. Images with Shadow.		Figure 8. Hazy Images.		Ground Truth Images (c)
I/P Image (a)	Restored Image (b)	I/P Image (a)	Restored Image (b)	
				



5.6 Comparative Analysis

The comparative analysis of CDRSHNet against existing methods in the literature reveals the model's significant prowess in diverse image-restoration challenges. As shown in Table 6, when addressing codec errors, CDRSHNet achieves a PSNR of 39.89 and an SSIM of 0.9800 for single images, outperforming other models not specifically tailored for this challenge. The model excels for dirty-lens correction, presenting a PSNR of 41.57 and an SSIM of 0.9820, surpassing previous techniques designed for single images and videos. CDRSHNet outshines prominent models in rain restoration, exhibiting a PSNR of 42.58 and an SSIM of 0.987, highlighting its superiority in handling rain-induced distortions. Moreover, in shadow removal, CDRSHNet remarkably outperforms existing techniques, demonstrating a PSNR of 44.34 and an SSIM of 0.9880, showcasing its effectiveness in eliminating shadows. However, in the domain of haze restoration, while CDRSHNet displays competitive results with a PSNR of 36.74 and an SSIM of 0.926, there are no metrics from previous methods in the literature for direct comparison. Overall, the CDRSHNet model consistently showcases superior performance across diverse image-restoration challenges, setting a new benchmark in the field.

Table 6. Comparison with existing methods.

Method	**Challenge Type**	**Image / Video**	**PSNR**	**SSIM**
Proposed	**Codec Error**	**Single Image**	**39.89**	**0.9800**
X. Li et al., [14]	Dirty lens	Video	35.37	0.9800
Y. Wang et al., [33]	Dirty lens	Single Image	23.43	0.8640
Proposed	**Dirty lens**	**Single Image**	**41.57**	**0.9820**
H. Wang et al., [16]	Rain	Single Image	35.31	0.9448
S. Li et al., [17]	Rain	Single Image	33.75	0.9412
D. Ren et al., [32]	Rain	Single Image	33.78	0.977
Proposed	**Rain**	**Single Image**	**42.58**	**0.987**
Z. Liu et al., [19]	Shadow	Single Image	25.92	0.9090
H. Fan et al., [21]	Shadow	Single Image	25.70	0.9826
X. Hu et al., [22]	Shadow	Single Image	25.07	0.8930
Proposed	**Shadow**	**Single Image**	**44.34**	**0.9880**
W. Yi et al., [34]	Haze	Single Image	19.1736	0.8864
D. Zhao et al., [26]	Haze	Single Image	16.032	0.626
Proposed	**Haze**	**Single Image**	**36.74**	**0.926**

To demonstrate generalization of the model, it is tested on the German Traffic Sign Recognition Benchmark (GTSRB) dataset and the Belgium Traffic Sign Dataset, as per the results in Table 7. The proposed model exhibits commendable performance across these datasets. Starting with the CURE-TSR dataset, the model achieves a substantial PSNR of 41.02, indicating high fidelity in image reconstruction. At the same time, the SSIM of 97.26 denotes a robust structural similarity to the original images. On the

GTSRB dataset, PSNR and SSIM show improvement, with values of 41.96 and 98.34, respectively. This signifies enhanced image-restoration capabilities, excelling in preserving image quality and structural details. The model's performance peaks on the BelgiumTS dataset, recording a PSNR of 42.62 and an impressive SSIM of 98.96. These values reflect the model's exceptional ability to restore images, surpassing its performance on previous datasets. In summary, the model consistently delivers superior results across datasets, with higher PSNR and SSIM values indicative of its proficiency in reconstructing images while maintaining structural fidelity, making it exceptionally reliable for scenarios where preserving fine details is paramount.

Table 7. PSNR and SSIM across diverse datasets.

CURE-TSR Dataset	**PSNR**	**41.02**
	SSIM	**97.26**
GTSRB Dataset	**PSNR**	**41.96**
	SSIM	**98.34**
BelgiumTS Dataset	**PSNR**	**42.62**
	SSIM	**98.96**

6. CONCLUSION

In this paper, we presented a novel CDRSHNet architecture for restoring traffic-sign images affected by different types of image degradation, including rain, shadow and haze, images taken with dirty lenses and corrupted due to codec error. This work introduces the fusion of self-attention and variance-guided multiscale attention modules with a custom-made loss function to effectively restore the images captured in adverse conditions. The proposed method uses varying learning-rate techniques, group normalization and dilation for better model performance. The experimental results show that the proposed model effectively restores images with high-quality metrics. The overall SSIM average of 0.978 for the Test RID dataset and an overall SSIM average of 0.963 for the Test SID dataset indicate the high restoration quality of our model. Similarly, the overall average PSNR values of 39.58 and 39.46 for Test RID and Test SID datasets, respectively, with an overall accuracy of 99.3%, further confirm the superior performance of the proposed novel architecture. The proposed model exhibits commendable performance across different datasets.

REFERENCES

[1] D. Temel, G. Kwon, M. Prabhushankar and G. AlRegib, "CURE-TSR: Challenging Unreal and Real Environments for Traffic Sign Recognition," arXiv (Cornell University), DOI: 10.48550/arxiv.1712.02463, Dec. 2017.

[2] J. Su, B. Xu and H. Yin, "A Survey of Deep Learning Approaches to Image Restoration," Neurocomputing, vol. 487, pp. 46–65, DOI: 10.1016/j.neucom.2022.02.046, May 2022.

[3] Z. Shen and D. Dang, "Mixed Hierarchy Network for Image Restoration," arXiv (Cornell University), , DOI: 10.48550/arxiv.2302.09554, Feb. 2023.

[4] M. Maru and M. C. Parikh, "Image Restoration Techniques: A Survey," Int. Journal of Computer Applications, vol. 160, no. 6, pp. 15–19, DOI: 10.5120/ijca2017913060, Feb. 2017.

[5] L.-Y. Chang and A. I. Kirkland, "Comparisons of Linear and Nonlinear Image Restoration," Microscopy and Microanalysis, vol. 12, no. 6, pp. 469–475, DOI: 10.1017/s1431927606060582, Oct. 2006.

[6] Z. Liu, "Literature Review on Image Restoration," Journal of Physics, Conference Series, vol. 2386, no. 1, p. 012041, IOP Publishing, DOI: 10.1088/1742-6596/2386/1/012041, Dec 2022.

[7] L. Yu, J. Guo and Y. Chen, "Research Status and Development Trend of Image Restoration Technology," Journal of Physics, vol. 2303, no. 1, DOI: 0.1088/1742-6596/2303/1/012081, 2022.

[8] C. Zhang, F. Du and Y. Zhang, "A Brief Review of Image Restoration Techniques Based on Generative Adversarial Models," Lecture Notes in Electrical Engineering, pp. 169–175, DOI: 10.1007/978-981-32-9244-4_24, 2019.

[9] S. Ahmed, U. Kamal and Md. K. Hasan, "DFR-TSD: A Deep Learning Based Framework for Robust Traffic Sign Detection under Challenging Weather Conditions," IEEE Transactions on Intelligent Transportation Systems, pp. 1–13, DOI: 10.1109/tits.2020.3048878, 2021.

[10] R. Huang, Y. Zhang and Z. Luo, "Inpainting of Compressed Images with Autoencoder-based Prior Learning," Proc. of the 26th ACM Int. Conf. on Multimedia, 236-244.

[11] S. Jeon, H. Kim and H. Kwon, "Compressed Image Restoration Using Autoencoder Regularization," Journal of Imaging Science and Technology, vol. 63, no. 6, pp.060403-1 - 060403-11, DOI: 10.2352/J.ImagingSci.Technol.2019.63.6.060403, 2019.

[12] K. Zhang, Y. Li and Y. Wang, "A Two-stage Method for Video Codec Error Concealment Using Deep Learning," IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 7, pp. 2102-2115, DOI: 10.1109/TCSVT.2020.2977168, 2020.

[13] M. Uricar et al., "Let's Get Dirty: GAN Based Data Augmentation for Camera Lens Soiling Detection in Autonomous Driving," Proc. of the IEEE/CVF Winter Conf. on Applications of Computer Vision, pp. 766-775, [Online], Available: http://arxiv.org/pdf/1912.02249.pdf, Dec. 2021.

[14] X. Li, B. Zhang, J. Liao and P. V. Sander, "Let's See Clearly: Contaminant Artifact Removal for Moving Cameras," Proc. of the Int. Conf. on Computer Vision, pp. 2011–2020, Montreal, Canada, Oct. 2021.

[15] J. Mohd, Sandra Mamani Reyes and J. Xiao, "Camera Lens Dust Detection and Dust Removal for Mobile Robots in Dusty Fields," Proc. of the 2021 IEEE Int. Conf. on Robotics and Biomimetics (ROBIO), DOI: 10.1109/robio54168.2021.9739233, Dec. 2021.

[16] H. Wang, Y. Wu, Q. Xie, Q. Zhao, Y. Liang et al., "Structural Residual Learning for Single Image Rain Removal," Knowledge-based Systems, vol. 213, p. 106595, Feb. 2021.

[17] S. Li, W. Ren, J. Zhang, J. Yu and X. Guo, "Single Image Rain Removal *via* a Deep Decomposition–Composition Network," Computer Vision and Image Understanding, vol. 186, pp. 48–57, Sep. 2019.

[18] M. Umair Arif, M. U. Farooq, R. H. Raza, Z. U. A. Lodhi and M. A. R. Hashmi, "A Comprehensive Review of Vehicle Detection Techniques under Varying Moving Cast Shadow Conditions Using Computer Vision and Deep Learning," IEEE Access, vol. 10, pp. 104863–104886, 2022.

[19] Z. Liu, H. Yin, Y. Mi, M. Pu and S. Wang, "Shadow Removal by a Lightness-guided Network with Training on Unpaired Data," IEEE Transactions on Image Processing, vol. 30, pp. 1853–1865, Jan. 2021.

[20] H. van Le and D. Samaras, "From Shadow Segmentation to Shadow Removal," arXiv (Cornell University), DOI: 10.48550/arxiv.2008.00267, Aug. 2020.

[21] H. Fan, M. Han and J. Li, "Image Shadow Removal Using End-to-End Deep Convolutional Neural Networks," Applied Sciences, vol. 9, no. 5, p. 1009, DOI: 10.3390/app9051009, Mar. 2019.

[22] X. Hu, Y. Jiang, C.-W. Fu and P.-A. Heng, "Mask-ShadowGAN: Learning to Remove Shadows from Unpaired Data," Proc. of the 2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV), vol. 2019, pp. 2472–2481, Seoul, S. Korea, Jan. 2019.

[23] L. Ren, Z. Pan, J. Cao, J. Liao and Y. Wang, "Infrared and Visible Image Fusion Based on Weighted Variance Guided Filter and Image Contrast Enhancement," Infrared Physics & Technology, vol. 114, p. 103662, DOI: 10.1016/j.infrared.2021.103662, May 2021.

[24] Q. Yang, C. Zhang, H. Wang, Q. He and L. Huo, "SV-FPN: Small Object Feature Enhancement and Variance-guided RoI Fusion for Feature Pyramid Networks," Electronics, vol. 11, no. 13, pp. 2028–2028, DOI: 10.3390/electronics11132028, Jun. 2022.

[25] X. Yang, "An Overview of the Attention Mechanisms in Computer Vision," Journal of Physics: Conference Series, vol. 1693, p. 012173, DOI: 10.1088/1742-6596/1693/1/012173, Dec. 2020.

[26] D. Zhao, L. Xu, Y. Yan, J. Chen and L.-Y. Duan, "Multi-scale Optimal Fusion Model for Single Image dehazing," Signal Processing-Image Communication, vol. 74, pp. 253–265, DOI: 10.1016/j.image.2019.02.004, May 2019.

[27] W. Yi et al., "Towards Compact Single Image Dehazing *via* Task-related Contrastive Network," Expert Systems with Applications, vol. 235, p. 121130, 2024.

[28] X. Zhu et al., "GAN-based Image Super-resolution with a Novel Quality Loss," Mathematical Problems in Engineering, vol. 2020, p. e5217429, DOI: 10.1155/2020/5217429, Feb. 2020.

[29] B. Wu, H. Duan, Z. Liu and G. Sun, "SRPGAN: Perceptual Generative Adversarial Network for Single Image Super Resolution," arXiv (Cornell University), DOI: 10.48550/arXiv.1712.05927, Dec. 2017.

[30] B. Vasant Gajera, S. Raj Kapil, D. Ziaei, J. Mangalagiri, E. L. Siegel and D. Chapman, "CT-Scan Denoising Using a Charbonnier Loss Generative Adversarial Network," IEEE Access, vol. 9, pp. 84093–84109, DOI: 10.1109/access.2021.3087424, Jun. 2021.

[31] W. Xue, L. Zhang, X. Mou and A. C. Bovik, "Gradient Magnitude Similarity Deviation: A Highly Efficient Perceptual Image Quality Index," IEEE Transactions on Image Processing, vol. 23, no. 2, pp. 684–695, DOI: 10.1109/tip.2013.2293423, Feb. 2014.

[32] D. Ren, W. Zuo, Q. Hu, P. Zhu and D. Meng, "Progressive Image Deraining Networks: A Better and Simpler Baseline," Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 3937-3946, DOI: 10.1109/cvpr.2019.00406, Long Beach, USA, Jun. 2019.

[33] Y. Wang, R. Wan, W. Yang, B. Wen, L.-P. Chau and A. C. Kot, "Removing Image Artifacts from Scratched Lens Protectors," arXiv (Cornell University), DOI: 10.48550/arxiv.2302.05746, Feb. 2023.

[34] W. Yi, M. Liu, L. Dong, Y. Zhao, X. Liu and M. Hui, "Restoration of Haze-free Images Using Generative Adversarial Network," Proceedings of the SPIE, vol. 11432, DOI: 10.1117/12.2541893, Feb. 2020.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

ملخص البحث:

في أحوال الطّقس الصّعبة، تؤثّر أمور متنوّعة على نوعيّة الصّور، ومنها قطرات المطر، والظّلال، والعيوب النّاتجة عن عدسات كاميرات متّسخة، ... وغيرها. وتكافح طرق معالجة هذه العيوب القائمة من أجل التغلّب عليها على نحوٍ شامل؛ إذ إنّ ذلك يتطلّب تقنيات إبداعية وحلولاً مبتكرة لاستعادة الصّور المشوبة بأيٍّ من العيوب سالفة الذّكر بحيث تكون أقرب ما يمكن للصّور الأصلية الخالية من العيوب.

تقدّم هذه الورقة نظاماً هجيناً يستفيد من عددٍ من التقنيات المتنوّعة الّتي تركّز على معالجة مواقع العيوب في الصّور الملتقطة تحت ظروف الطّقس السّيّئة. وقد تمّ تدريب النّظام المقترح وتقييمه باستخدام مجموعات بيانات ملائمة؛ إذ بلغت دقّة النّظام في استعادةِ الصّور (99.3%).

وقد جرت مقارنة النّظام المقترح في هذه الدراسة مع عددٍ من الأنظمة الواردة في أدبيات الموضوع، حيث تبين أنّ النّظام المقترح يتفوّق على ما سواه من حيث وضوح الصّور المعالَجة وقُربها من الصّور الأصلية.