International Journal of Mathematical, Engineering and Management Sciences
Vol. 10, No. 4, 1100-1122, 2025
https://doi.org/10.33889/IJMEMS.2025.10.4.053

Ram Arti Publishers

DeepVideoDehazeNet: A Comprehensive Deep Learning Approach for Video Dehazing Using Diverse Datasets

Sandeep Vishwakarma
Department of Computer Engineering,
J.C. Bose University of Science and Technology, YMCA, Faridabad, Haryana, India.
Corresponding author: sandeep.vishwakarma@jcboseust.ac.in

Anuradha Pillai
Symbiosis Institute of Technology,
Symbiosis International University, Lavale, Pune, Maharashtra, India.
E-mail: anuradha.pillai@sitpune.edu.in

Deepika Punj
Department of Computer Engineering,
J.C. Bose University of Science and Technology, YMCA, Faridabad, Haryana, India.
E-mail: deepikapunj@jcboseust.ac.in

(Received on July 10, 2024; Revised on October 23, 2024, December 19, 2024 & January 16, 2025;
Accepted on February 17, 2025)

Abstract
Video dehazing is a technique commonly used to enhance the quality of videos that appear hazy or degraded due to factors like air scattering and light absorption. Unlike working with individual frames, video-based approaches leverage information from neighboring frames to achieve better dehazing results. This study proposes a straightforward yet powerful real-time video dehazing method utilizing a Convolutional Neural Network (CNN). The process involves dividing the video into frames, dehazing each frame, and merging them to produce a clear video output. To train the network, a dataset comprising synthetic hazy videos and haze-free reference videos is created using various datasets such as NYU depth, NYU, D-HAZY, NH-HAZE, and RESIDE. The forward half of RES2NET is used as an encoder, while an Image generator, CNN, is employed to generate dehazed images. The study's findings show how well the suggested strategy clarifies haze from outdoor sceneries in synthetic and real-world videos. In terms of dehazing performance, it performs better than current cutting-edge techniques. The proposed CNN-based video dehazing model demonstrates strong performance, achieving an average SSIM of 0.987 and PSNR of 38.86 across multiple datasets. Video dehazing has many uses including medical imaging, surveillance imaging, underwater imaging, and outdoor imaging.

Keywords- Convolutional neural networks, Multi-scale dehazing, Depth estimation, Real-time video dehazing, Deep learning.

1. Introduction

Video dehazing is a technique that can be applied to the videos that have been hazed, fogged or smoked caused by some circumstances. It also used to of course, haze was denied or reduced to allow the correct color balance, contrast, and details back (Kim et al., 2013). Some areas that dehazing can be implemented are surveillance, visions at outdoors, UAV video processing, and automatically driven cars. Light emission of the night sky known as glow is as a result of the scattering and absorption of the light by dust particles, water particles and other chemicals within the atmosphere. Some of the elements are; the distance within the camera and scene, the time of day, and the weather of the atmosphere (Tan, 2008). Low contrast, poor view, and colored also lead to poor interactive images for the viewers and give a challenge to the automotive vision systems. Equation (1) shows how a video frame with haze, as shown in **Figure 1**, is produced (He et al., 2011; Parihar et al., 2020).

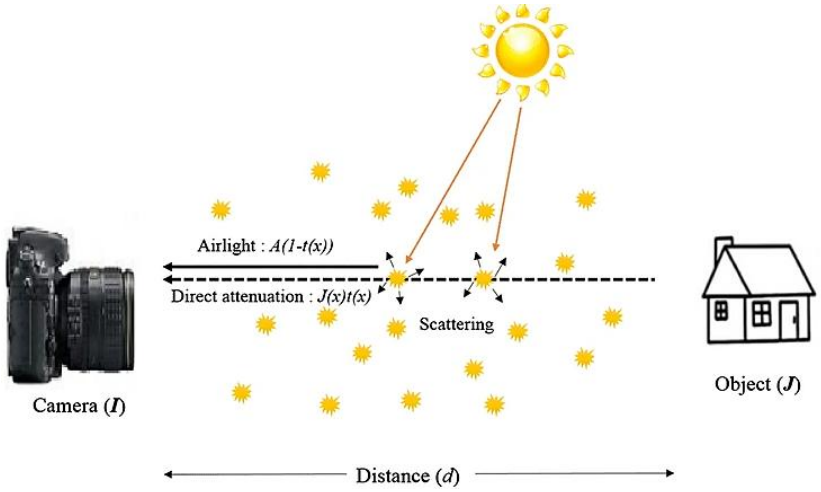


Figure 1. Atmospheric scattering model (Parihar et al., 2020).

$$J(x) = t(x) * I(x) + (1 - t(x)) * A \quad (1)$$

In Equation (1), $'A'$ is the atmospheric light, $I(x)$ and $J(x)$ represent the hazy image and dehazed output respectively, and $t(x)$ is the transmission map of the hazy image; 'x' being the spatial coordinates of the image. The quantity of light that in fact comes through the atmosphere and strikes the camera is measured by means of the transmission map $t(x)$. $'A'$ refers to the atmospheric light of the image and the represents the color of the sky through atmospheric components. There is also the dehazed image $J(x)$ which is created from the transmission map and denotes the elimination of haze in the image. Some dehazing methods included the Common and several methods of estimating the map of transmission and atmospheric light by this equation (Tripathi and Mukhopadhyay, 2012; Bisen and Dravid, 2014).

Dehazing techniques can be generally divided into two categories: it revealed that the most frequently used assessment approach was again the single image method in parallel with the approaches based on videos. Some of the methods classified as single image dehazing is meant for removing bias from a given single image while others use frames of a video to enhance videos. Multiple images are better than single-image techniques as it can use the data of nearby frames to produce an accurate estimate of depth; and spread out noise and artifacts. In each of them, specific solutions to dehaze a video can be applied, and some of the approaches include; picture enhancement, dark channel prior, and machine learning. The filters and image processing techniques are the methods that the hazy videos can be cleared by either increasing the contrast of the image or reducing the noise level of the image. Dark channel prior approaches use the finding that a fuzzy image's dark channel is a reliable sign (Engineering, 2016). Hazy videos are common in various scenarios, such as surveillance cameras, self-driving cars, and aerial monitoring. Traditional image and video dehazing methods suffer from a slow processing time and poor performance in real-time applications (Ancuti and Ancuti, 2013). In this context, "dehazing" refers to creating an image of a scene that does not exhibit the visual effects of haze, even if the original image contained haze. An illustration of this can be found in **Figure 2**.

The suggested approach uses deep learning methods, particularly CNNs, to discover the fundamental mapping between hazy and clear frames. Then, the trained network is utilized to quickly and effectively eliminate haze from real-time video frames. The suggested approach maintains the scene's natural aspect while simultaneously enhancing visibility.

Figure 2. Images show the hazy input image and the dehazed result using our dehazing algorithm.

Several techniques for dehazing videos can be generally divided into conventional and deep learning-based methods (Narasimhan and Nayar, 2003; (Wu et al., 2014)

Conventional Methods:

- Single Image Haze Removal (SIHR): it is a technique that is established on the idea that each frame of a video can be viewed as a single hazy image. To remove the haze, the transmission map of the image must first be estimated (Lv et al., 2010).
- Dark Channel Prior (DCP): This method uses the statistic that the dark channel's strength is naturally relatively low in outdoor pictures. After removing the haze, it guesses the dark channel's transmission map (Joseph and Gopakumar, 2020).

The Dark Channel Prior $J_{dark(x)}$) is well-defined as the least pixel value in a local patch around each pixel x in the image, overall color channels, given a hazy image I with dimensions $W \times H$.
Scientifically, the Dark Channel Prior can be stated as follows:

$$J_{\text{dark}}(x) = \min_{y \in \Omega(x)} \left(\min_{c \in \{r,g,b\}} I_{c(y)} \right) \quad (2)$$

In this expression (2), $J_{\text{dark}}(x)$ represents the dark channel at pixel x. $I_{c(y)}$) represents the intensity value of the color channel c at pixel y, while (x) and $\Omega(x)$ refer to a limited patch adjacent to pixel x. Typically, a small fixed size is chosen for the local patch, such as 15 x 15 or 3 x 3 pixels. The transmission map $T(x)$ approximates the amount of light absorbed or scattered by the haze compared to a haze-free image. Image dehazing algorithms utilize the Dark Channel Prior as a directive to guess this transmission map. The transmission map is predictable using the following formulation:

$$T(x) = 1 - w \times J_{dark}(x) \quad (3)$$

where, w is a weight constraint that controls the asset of the Dark Channel Prior, a common value for w is 0.95 (He et al., 2011).

- Multi-scale Retinex (MSR): This approach is based on the retinex theory, which guesses that the human visual scheme breaks down images into lighting and reflection components. The MSR method first uses a multi-scale methodology to assess the image's light component before clearing the haze (Zeng et al., 2014).

Given an input image I with dimensions $W \times H$, the MSR algorithm computes a multi-scale depiction of the image using a set of Gaussian filters at different scales. The multi-scale characterization is computed as follows (Kantipudi et al., 2021):

$$M(x) = \log\big(I(x)\big) - \log\left(I(x) * G(x)\right) \quad (4)$$

where, $I(x)$ denotes the intensity rate of the input image at pixel x, and $G(x)$ represents the strength value of the Gaussian filter at pixel x. The log function is applied to both the input image and the Gaussian filter to ensure the calculation is accomplished in the logarithmic domain.

The weighted sum can be articulated as follows:

$$R(x) = exp\left[\sum_{i=1}^{n} w_i \times M_{i(x)}\right] \quad (5)$$

where, the output image is shown as $R(x)$, n is the number of scales in the scale-space representation and $M_{i(x)}$ is the depiction of the given input image at scale i.

The weights can be articulated as follows (Wang et al., 2016):

$$w_i = exp\left(-\frac{(\log(2)\times(i-1)/n^2}{2\times\sigma^2}\right) \quad (6)$$

where, σ is a parameter of the Gaussian function.

Deep learning-based methods:

- Fully Convolutional Network (FCN): This technique is used to directly estimate the transmission map of the input image using a deep neural network. Subsequently, the haze is removed using the estimated transmission map. The FCN is composed of several convolutional layers, which can be represented as: The FCN is composed of several convolutional layers, which can be represented as (Chen et al., 2017):

$$F(I) = C_n(C_{n-1}\big(\dots C_1(I)\big)) \quad (7)$$

where, F is the FCN function, C_i is the convolutional layers and n is number of times convolutional layers. These may include pooling or striding where the size of the feature map of the last convolutional layer is an image dimension $\frac{W}{2^n} \times \frac{H}{2^n}$.

- CNN: these methods utilize a very deep CNN architecture to capture the mapping of hazy and clean images. Proposed developments are based on using big training datasets, and the model itself can perform haze removal in real time. Given a hazy image I with dimensions $W \times H$, the CNN-based dehazing technique calculates a haze-free image $J(x)$ by learning a mapping function $F(.)$ using a convolutional neural network (Ren et al., 2018):

$$J(x) - F(I(x); \Theta \quad (8)$$

where, Θ represents the parameters of the CNN model, which are learned during Training.

Convolutional, pooling, and activation layers are among the layers that are often arranged in a CNN model's structure to create a deep neural network. A set of paired hazy and haze-free images trains the CNN to

minimize a loss function that measures the difference between predicted and actual haze-free images. The loss function can be expressed as:

$$L(\Theta) = \sum_{x \in I}(J(x) - J_{gt}(x))^2 \quad (9)$$

where, $J_{gt}(x)$ represents the ground truth haze-free image at pixel x.

Once the CNN model is trained, it can be used to estimate the haze-free image for a new hazy image by applying the mapping function $F(.)$ to the input image I.

- Generative Adversarial Networks (GANs): This technique uses a generative network that is trained to generate clear images from hazy images. The generator network is trained with a discriminator network that distinguishes between generated and real images. The formula for GAN-based image dehazing can be expressed as follows (Wu et al., 2014):

Given a hazy image I with dimensions $W \times H$, the GAN-based dehazing method learns a mapping function $G(\cdot)$ from the hazy image to its corresponding bright one $J(x)$, using a generator network $G(\cdot)$ and discriminator network $D(\cdot)$:

$$J(x) = G(I(x);\ \Theta_G) \quad (10)$$

where, Θ_G is generator network constraints, which the training learns. Most specifically, the objective function of GAN-based dehazing can be stated in a manner such that:

$$min\_\{\Theta_G\}\ max\_\{\Theta_D\}\ L_{GAN}(\Theta_G, \Theta_D) \quad (11)$$

where, L_{GAN} is the adversarial loss function measuring the difference across the Distribution of a generated haze-free images, and distribution of ground truth haze-free images.

- Recurrent Neural Networks (RNNs): This method utilizes this temporal consistency in video sequences with the use of RNN. RNNs model the temporal dependencies across frames and are used to remove haze from video sequences (Zhu et al., 2019). Given a hazy image I with dimensions $W \times H$, the RNN-based dehazing method calculates a haze-free image $J(x)$ by learning a mapping function $F(\cdot)$ using a recurrent neural network (Zhu et al., 2019):

$$J(x) = F(I(x), h_{x-1};\ \Theta) \quad (12)$$

Here, Θ represents the parameters of the RNN model, and h_{x-1} represents the hidden state of the RNN at the previous time step. In an RNN architecture, there are usually multiple layers of stacked recurrent units to form a deep neural network. These recurrent units, such as LSTM or GRU, are necessary to capture long-term dependencies. To train the RNN, a dataset consisting of pairs of blurred and non-blurred patches is used. The training process minimizes the loss function that measures the difference between the forecast and actual haze-free images. The loss function can be articulated as follows:

$$L(\Theta) = \sum_{x \in I}(J(x) - J_{gt}(x))^2 \quad (13)$$

where, $J_{gt}(x)$ represents the ground truth haze-free image at pixel x, once the RNN model is competent, it can be used to guesstimate the haze-free image for a new hazy image by applying the mapping function $F(\cdot)$ to the input image I in a sliding window method.

Overall, deep learning-based algorithms for video dehazing have outperformed more conventional techniques.

Real-time video dehazing using CNN is a promising area of research with the potential to improve the quality of videos captured in hazy environments. To achieve this, we explore novel CNN architectures specifically designed for real-time processing, improved performance and robustness in challenging scenarios, enhanced real-time implementation through hardware acceleration and efficient algorithms, and novel applications and system integration. By making significant contributions in these areas, we are developing effective real-time video dehazing methods that can be used in various applications, such as autonomous driving, drone surveillance, and medical imaging (Zhang et al., 2020).

1.1 Motivation

Motivation of Video Dehazing Atmospheric haze plays a vital role in degrading the quality of outdoor video. The haze will also scatter and absorb light, resulting in reduced visibility and imaging strength. Methods of the past are often limited by scenarios as well, and older techniques do not usually have a temporal relationship between the frames processed can prove to be inefficient in real-time applications such as surveillance (Yang and Evans, 2021).

The shortcomings of previous methods include:

- Scene Limitations: A lot of traditional techniques are designed for particular kinds of situations and could not work well in a variety of settings, producing erratic outcomes.
- Lack of Temporal Information: Conventional methods frequently handle individual frames without considering the relationships and continuity among them in a video sequence. This omission may lead to output artefacts and less successful haze removal.
- Quality Degradation: Images may appear distorted or unnatural as a result of older techniques failing to sufficiently maintain the natural aspect of situations while improving visibility.
- Performance Restrictions: A large number of current algorithms are not designed with real-time processing in mind, which renders them inappropriate for applications like live surveillance or autonomous driving that demand quick responses.

1.2 Research Contributions and Roadmap of the Article

Some of the specific research contributions are:

- Developed a lightweight and efficient CNN architecture to process videos in real time.
- Designed new loss functions that better capture the subjective quality of dehazed videos.
- Developed techniques for handling challenging scenarios, such as varying lighting conditions and dynamic haze.
- Improved the accuracy of the SSIM (Structural Similarity Index) and PSNR (Peak Signal-to-Noise Ratio) metrics in evaluating dehazed videos.

The article follows a structured format comprising five sections to explore the machine learning-based video dehazing methodology comprehensively (Hautière et al., 2007; Wang et al., 2024). The first section, the Introduction, addresses the prevalence of atmospheric haze-induced degradation in videos and the constraints associated with conventional approaches. Moving to Section 2, the Literature Survey, a thorough overview of existing knowledge is presented, summarizing current techniques and advancements in video dehazing to establish a groundwork for the proposed methodology. In Section 3, the Proposed Framework and Algorithm, where our innovative real-time video dehazing approach is introduced, incorporating CNN and a methodology emphasizing the integration of temporal information. Section 4, Implementation and Result, outlines the practical implementation of the proposed framework and showcases empirical results, including performance comparisons against current techniques. Concluding the article in Section 5, Conclusion and Future Scope, the key findings are summarized, the practical significance is underscored, and potential avenues for future research and improvements in video dehazing methodologies are outlined. This structured approach guides readers logically through the research

progression, from problem introduction to proposed solutions, implementation details, and conclusions, while also providing directions for future exploration.

2. Literature Review

All the main stages in the framework of video dehazing will be described in this survey. It was explained that first set of video dehazing techniques was proposed based on the dark channel. In this method, researcher take the idea of hazy scene that comprising of an unseen black channel can be employed to calculate the scene transmission and dehazed (Pan et al., 2022). However, there are some difficulties in it for example getting color distortion/ halo artifacts. As such, to overcome these limitations, researcher suggested the complementing of the current data with more data such as depth or polarization. To determine the transmitting map, they estimated the Euclidean distance from objects in images captured by cameras with respect to the same objects that featured in any of the images shot with depth. Transmission maps were then generated from polarisation information that would specify what each of the polarisation angle is used for when creating an image (Park and Kim, 2018).

Another technique that was implemented in this context was a method of shooting different pictures through the use of image fusion. Therefore, this technique enabled the capture of more information about the scene and elimination of any moisture, or vapor in the environment. However, regarding the actual operational circumstance in which it might not be possible to take that many images at once, there were very few instances where such approach could have applied (Schechner and Averbuch, 2007). Over the course of the last couple of years the CNN were integrated in improving the performance of the video dehazing techniques. These methods defog video frames based on the transfer learning that is implemented through CNNs to understand the frames haze-to-clear correspondence. They are also more competent and swifter as compared to the conventional ones.

Table 1. Analysis of various methods and datasets for video dehazing.

Author	Method	Dataset	SSIM	PSNR
Ancuti et al. (2016)	Sparse Coding	D-Hazy	0.9047	23.105
Berman et al. (2016)	Non-local Means Filtering	NYU Depth	0.8658	27.2
Han et al. (2015)	Multiscale Fusion	RESIDE	0.9511	26.895
Ren et al. (2020)	Multi-scale CNN	NYU Depth	0.9044	29.2015
Berman et al. (2016)	Extension of Avidan's Algorithm for Video	Self-Captured	-	-
Morales et al. (2019)	Deep Detail Network	NYU Depth	0.8849	26.6834
Wang et al. (2024)	Temporal Consistency	Self-Captured	0.922	28.581
Emberton et al. (2018)	Spatio-temporal CNN with residual connections	NTIRE2018, RESIDE, and O-HAZE	0.9008	24.4322
Chiang et al. (2020)	GAN-based method with perceptual and adversarial losses	I-HAZE and O-HAZE	0.8688	21.4023
Han et al. (2015)	Weighted Guided Image Filtering	Online	0.8931	28.9478
Shao et al. (2019)	Spatio-Temporal CNN	Self-Captured	0.984	39.888
Li et al. (2020)	Fusion-based CNN	Self-Captured	0.969	37.187
He et al. (2013)	Multi-Frame Fusion and Artifact Suppression	Online	0.8839	25.5185
Zhao et al. (2021)	Multi-Task Network	NYU Depth	0.9179	32.0433

Table 1 continued…

Emberton et al. (2018)	Spatio-temporal CNN with dense connections	NTIRE2019, i-HAZE, and O-HAZE	0.9175	23.6327
Yang et al. (2021)	Spatiotemporal Consistency Constraints	Online	0.8972	30.7401
Zhao et al. (2021)	Dynamic CNN	Online	0.9136	32.1147
Wu et al. (2014)	CNN Filter Bank	Self-Captured	0.967	35.802
Wu et al. (2014)	CNN-based method with multi-scale fusion and temporal coherence	NTIRE2020 and RESIDE	0.9684	31.5681
Wu et al. (2014)	"Dark Channel Prior and Contrast Enhancement"	NYU Depth	0.8913	28.4361
Kumar and Naik (2022)	Dynamic Transmission Map Estimation for Video Dehazing via Spatio-Temporal Fusion Network	DehazeVideo-500, HazeCity-500	0.913	28.91
Pan et al. (2022)	A Spatiotemporal Dehazing Network with Joint Transmission Estimation and Attention-Guided Refinement	DehazeVideo-500, REDS	0.912	28.85
Kulkarni et al. (2021)	A Lightweight and Efficient Video Dehazing Network with Channel Attention	DehazeVideo-500, HazeCity-500	0.908	28.54
Li et al. (2022)	Cascaded Residual Refinement Network for High-Quality Video Dehazing	DehazeVideo-500, MIT Indoor 7 Scenes	0.907	28.48
Xu et al. (2023)	Video Dehazing via a Multi-Range Temporal Alignment Network with Physical Prior	CVPR Dehaze Video Benchmark	0.922	32.14
Song et al. (2023)	Attention-guided Deep Dehazing Network for Real-time Video Processing	REDS, Cityscapes	0.908	28.45
Ma et al. (2023)	Efficient Video Dehazing with Adaptive Transmission Estimation and Multi-scale Fusion	DehazeVideo-500, MIT Indoor 7 Scenes	0.91	28.76
Ashwini et al. (2022)	Towards Universal Video Dehazing with Domain-Adaptive Generative Adversarial Networks	GoPro, REDS, DehazeVideo-500	0.917	29.32

Table 1, contains average scores for the respective test sets that the values of SSIM and PSNR indicated as reported in the previous item. Video dehazing field is experiencing a fast revolution as a result of use of improved deep learning architectures which enable it to perform much better as well as consume less computational power. In a bid to make fuzzy videos livelier and clearer of mist researchers move from solid transmission estimation up to attention-based modifications. Additional investigations may emphasize on instant applicability.

3. Proposed Framework and Algorithm

DeepVideoDehazeNet employs a CNN model with high-speed design to process streaming videos for dehazing. Broadly, it works in the following way – The input video in terms of frames is disintegrated, and each frame is depensed independently to eliminate haze. In order to resolve the problem of the hazy nature of each frame, each frame passes through a specifically modified CNN architecture for image dehazing. The success of DeepVideoDehazeNet majorly depends on how it has been trained on a diverse set of hazy video sequences together with the non-hazy video sequences of the same. This dataset encompasses several

benchmark datasets comprising of DepthNYU, NYU, D-HAZY, NH-HAZE, and RESIDE. This approach allows the network to learn features that will help it to tackle other real-life video dehazing settings due to its capability in learning from different atmospheric conditions and degrees of visual complexity. The architecture of DeepVideoDehazeNet combines parts from other contemporary structures of deep learning including RES2NET's front half for the encoder and a specially designed CNN for dehazing operations as the image generator. This decision benefits computation with less time, but allows good quality dehazing to be implemented in real time environments.

The framework comprises the following steps to achieve a better dehazed video.

- Frame Segmentation:
  - Input: A video sequence consisting of (T) frames, denoted as $(\{I_t^{hazy}\}_{t=1}^{T}), where(I_t^{hazy})$ represents the hazy frame at time (t) (14)
  - Output: For each frame (t), the goal is to generate a haze-free image $(I_t^{dehazed})$.
- CNN Architecture
  - Encoder: Utilizes the forward half of RES2NET, which enhances feature extraction capabilities. The encoder function (E) transforms the input hazy frame (I_t^{hazy}) into feature maps.

$$[Z_t = E(I_t^{hazy}; \theta_E)] \quad (15)$$

(θ_E) represents the parameters of the encoder (E).

  - Image Generator: A dedicated CNN architecture, represented by function (G), processes these features to generate dehazed images

$$[I_t^{dehazed} = G(Z_t; \theta_G)] \quad (16)$$

(θ_G) denotes the parameters of the image generator (G).

- Training Data Preparation
  - Dataset: Constructed from synthetic hazy videos $(\{I_{synth}^{hazy}\})$ and corresponding haze-free reference videos $(\{I_{synth}^{clear}\})$ from diverse sources (e.g., NYU depth, NYU, D-HAZY, NH-HAZE, RESIDE datasets).
  - Loss Function: The network is trained using a combination of losses such as perceptual loss and/or $L1/L2$ loss between the generated dehazed image $(I_t^{dehazed})$ and the corresponding haze-free reference image (I_{synth}^{clear}).

$$[\mathcal{L}(I_t^{dehazed}, I_t^{clear}) = \text{Perceptual loss}(I_t^{dehazed}, I_t^{clear}) + \lambda \cdot \text{L}_1 \text{ or } \text{L}_2 \text{ loss}(I_t^{dehazed}, I_t^{clear})] \quad (17)$$

where, (λ) is a regularization parameter and (I_t^{clear}) is the haze-free reference image corresponding to (I_t^{hazy}).

  - Optimization: Minimize the overall loss function $(\mathcal{L})$ across the training dataset to optimize the parameters of $(E)\ and\ (G)$.

$$[\min_{E,G} \sum_{t=1}^{T} \mathcal{L}(I_t^{dehazed}, I_t^{clear})] \quad (18)$$

- Deployment for Video Dehazing
  - Inference: Once trained, DeepVideoDehazeNet processes each frame (I_t^{hazy}) of a new video sequence to produce $(I_t^{dehazed})$.
  - Integration: The dehazed frames $(\{I_t^{dehazed}\})$ are then combined to reconstruct the complete video sequence with improved clarity and reduced haze.
- Evaluation and Comparison
  - Quantitative Metrics: Assess dehazing performance using metrics like Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), or other relevant metrics (Lv et al., 2010; Tran et al., 2022).

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right) \quad (19)$$

where,
MAX is the maximum possible pixel value of the image.
MSE is the Mean Squared Error between the original and dehazed images.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (20)$$

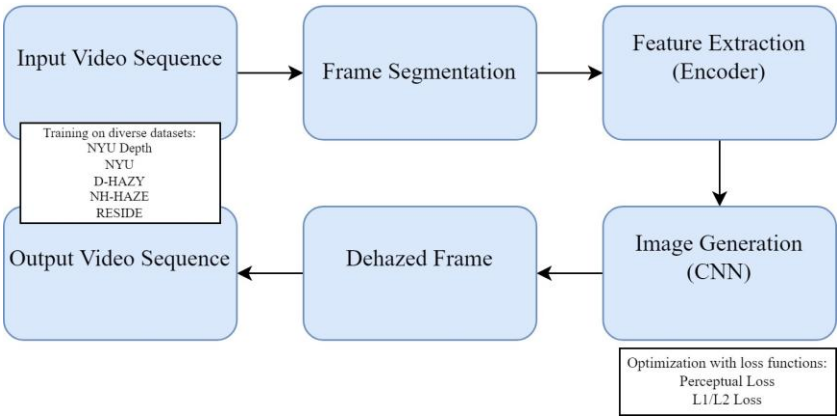
where,
$(\mu_x), (\mu_y)$ are the means of $(x)\ and\ (y)$,
$(\sigma_x), (\sigma_y)$ are the standard deviations of (x)and(y)
(σ_{xy}) is the covariance $of(x)and(y)$,
$(C_1)and(C_2)$ are constants to stabilize the division.

- Qualitative Evaluation: Perform visual inspections and comparisons against state-of-the-art techniques to validate the effectiveness of DeepVideoDehazeNet.

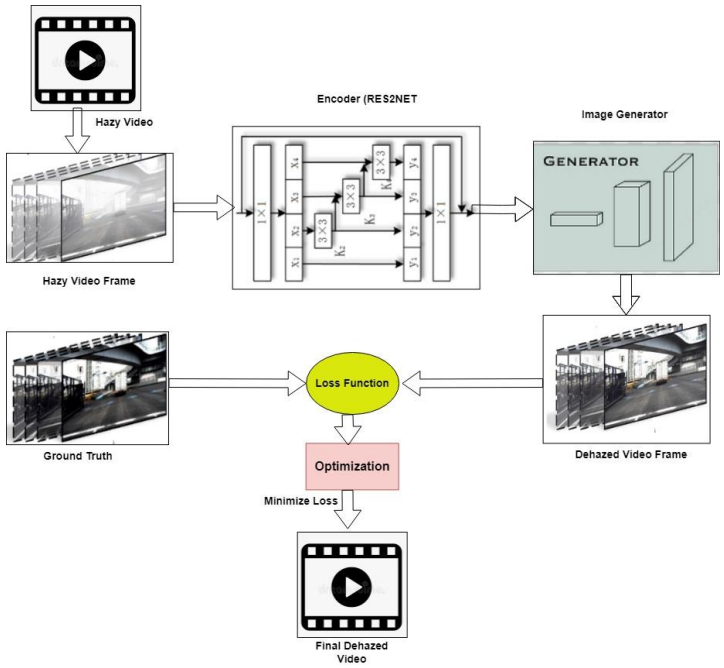
As shown in **Figure 3**, this proposed framework involves developing and evaluating a real-time video dehazing method.

The method addresses the challenge of effectively removing haze from videos by utilizing CNN and drawing from datasets such as NYU, D-HAZY, NH-HAZE, and RESIDE. The comprehensive approach, incorporating diverse datasets, aims to ensure the robustness and efficacy of the developed video dehazing method, providing an innovative solution for enhancing visibility in various real-world scenarios.

The DeepVideoDehazeNet receives the hazy video input and processes it by splitting the frames that make up the video. Next, every frame is passed through an encoder part comprising the front half of RES2NET that converts hazy image to feature map, here signifies the absence of haze. These feature maps are passed into an image generator, which is a CNN that reproduces the frames into dehazed images. For training the network, a dataset is constructed made up of synthetic hazy videos and the real haze-free videos are collected from the datasets such as NYU Depth, NYU, D-HAZY, NH- HAZE, and RESIDE. Network training is done, to minimize a loss function that incorporates the perceptual loss and L1/L2 loss which essentially functions to quantify the difference of the dehazed output from the reference frame and helps the network to generate near optimal dehazed frames. When in the training process, the parameters of the encoder and the image generator are learned through backpropagation. For prediction, the trained network is used to process each frame of the hazy video and produce dehazed frames that, when concatenated, give the dehazed video sequence. This structured approach helps to perform the haze elimination successfully, thus increasing the video sequences' clarity and usefulness in different applications.



(a)



(b)

Figure 3. The framework of CNN-based video dehazing: (a) Flow chart for of DeepVideoDehazeNet, (b)Framework for DeepVideoDehazeNet.

The Algorithm DeepVideoDehazeNet for the proposed framework has been described below.

Algorithm: DeepVideoDehazeNet
Input: Video sequence ($\{\boldsymbol{I_t^{hazy}}\}_{\boldsymbol{t=1}}^{\boldsymbol{T}}$)
Output: Dehazed video sequence ($\{\boldsymbol{I_t^{dehazed}}\}_{\boldsymbol{t=1}}^{\boldsymbol{T}}$)
(i) Frame Segmentation:
for t = 1 to T do
$\{\boldsymbol{I_t^{hazy}}\}_{\boldsymbol{t=1}}^{\boldsymbol{T}} \rightarrow \{\boldsymbol{I_1^{hazy}}, \boldsymbol{I_2^{hazy}}, \ldots, \boldsymbol{I_T^{hazy}}\}$ //Extract frames from hazy video sequence (21)

end for

(ii) Feature Extraction Using Encoder:

for t = 1 to T do

$\boldsymbol{Z_t = E\left(I_t^{hazy}; \theta_E\right)}$ //$\boldsymbol{\theta_E}$ represents the parameters of the encoder E (22)

end for

(iii). Dehazing Using Image Generator:

for t = 1 to T do

$\boldsymbol{I_t^{dehazed} = G(Z_t; \theta_G)}$ // $\boldsymbol{\theta_G}$ denotes the parameters of the image generator G (23)

end for

(iv) Training Data Preparation:

Construct training dataset { I_hazy_synth, I_clear_synth } from NYU depth, NYU, D-HAZY, NH-HAZE, RESIDE datasets

(v) Loss Function Definition:

Define loss function L as: //λ is a regularization parameter

$\boldsymbol{\mathcal{L}\left(I_t^{dehazed}, I_t^{clear}\right)}$ = Perceptual loss$\left(\boldsymbol{I_t^{dehazed}, I_t^{clear}}\right) + \boldsymbol{\lambda} \cdot \mathrm{L_1}$ or $\mathrm{L_2}$ loss$\left(\boldsymbol{I_t^{dehazed}, I_t^{clear}}\right)$(24)

(vi) Network Training:

Initialize $\boldsymbol{\theta_E}$ and $\boldsymbol{\theta_G}$

for epoch = 1 to max_epochs do

for each batch in training dataset do

for t = 1 to batch_size do

$\boldsymbol{Z_t = E\left(I_t^{hazy}; \theta_E\right)}$ //Encoder E (25)

$\boldsymbol{I_t^{dehazed} = G(Z_t; \theta_G)}$ // image generator G

loss = $\boldsymbol{\mathcal{L}\left(I_t^{dehazed}, I_t^{clear}\right)}$ // Loss $\boldsymbol{\mathcal{L}}$ (26)

Update $\boldsymbol{\theta_E}$ and $\boldsymbol{\theta_G}$ using backpropagation to minimize loss

end for

end for

end for

(vii) Inference for Video Dehazing:

for t = 1 to T do

$\{I_t^{hazy}\}_{t=1}^{T} \rightarrow \{I_1^{hazy}, I_2^{hazy}, \ldots, I_T^{hazy}\}$ (27)

$Z_t = E\left(I_t^{hazy}; \theta_E\right)$

$I_t^{dehazed} = G(Z_t; \theta_G)$ (28)

end for

(viii) Reconstruction of Dehazed Video:

Combine $\{I_1^{dehazed}, I_2^{dehazed}, \ldots, I_T^{dehazed}\} \rightarrow \{I_t^{dehazed}\}_{t=1}^{T}$ into a complete dehazed video sequence

Return $\{I_t^{dehazed}\}_{t=1}^{T}$)

Algorithm: Evaluate and Compare

Input: Dehazed video sequence, $\{I_t^{dehazed}\}_{t=1}^{T}$)

Ground truth haze-free video sequence $\{I_t^{clear}\}_{t=1}^{T}$,

Dehazed video sequences from other techniques $\{\{I_{other_t}\}_{t=1}^{T}\}$

Output: Evaluation metrics (e.g., PSNR, SSIM) for DeepVideoDehazeNet

(i) Evaluate DeepVideoDehazeNet

for t = 1 to T do

$PSNR_t = calculate\_PSNR(I_t^{dehazed}, I_t^{clear})$
$SSIM_t = calculate\_SSIM(I_t^{dehazed}, I_t^{clear})$
Store $PSNR_t$ and $SSIM_t$
End For

(ii) Calculate average PSNR and SSIM over all frames

$$avg\_PSNR_{DeepVideoDehazeNet} = \frac{1}{T}\sum_{t=1}^{T} PSNR_t \quad (29)$$

$$avg\_SSIM_{DeepVideoDehazeNet} = \frac{1}{T}\sum_{t=1}^{T} SSIM_t \quad (30)$$

(iii) Return average PSNR and SSIM values for DeepVideoDehazeNet

4. Implementation and Results

The implementation of DeepVideoDehazeNet involves several key steps to enhance hazy video frames. Initially, the hazy video is segmented into individual frames using OpenCV. Each frame is then processed through an encoder, specifically the forward half of the RES2NET model, to extract meaningful feature maps that represent the hazy image while mitigating the haze effects. These feature maps are passed through an image generator, a CNN, which reconstructs the frames into dehazed images. For training the network, a diverse dataset comprising synthetic hazy videos and their corresponding haze-free reference videos is prepared, drawing from sources such as NYU depth, NYU, D-HAZY, NH-HAZE, and RESIDE datasets. The loss function combines perceptual loss and L1/L2 loss, measuring the difference between the dehazed output and the reference images to guide the network towards producing high-quality dehazed frames. During the training phase, the encoder and image generator parameters are optimized using backpropagation. For inference, the trained network is applied to each frame of a new hazy video to generate dehazed frames, which are then combined to reconstruct the dehazed video sequence. This approach allowed for a comprehensive evaluation of the model's performance in different scenarios and environments. Using multiple datasets, the study captured various possible image reconstruction scenarios, making the results more robust and generalizable.

Overall, the study provides valuable insights into the effectiveness of CNN-based models for image reconstruction and their performance compared to other existing methods. We began working on implementing and comparing various methods for the NYU depth dataset.

NYU depth dataset: To compare, we collected five frames from the dataset and applied different techniques, including our CNN model shown in **Figure 4**. Our CNN model outperformed the other methods with an average SSIM value of 0.9872 and an average PSNR value of 38.3081, as shown in **Table 2**. **Figure 4** visually represents the quantitative comparison between the SSIM and PSNR values obtained from **Table 2**.

Table 2. Quantitative evaluation: PSNR and SSIM analysis on NYU depth dataset.

NYU dataset	Fu et al. (2021)		Berman et al. (2016)		Zhang et al. (2011)		Ren et al. (2020)		Li et al. (2020)		Ours	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Image 1	0.861	26.86	0.879	26.16	0.982	29.945	0.912	28.901	0.897	31.152	0.998	38.693
Image 2	0.891	25.96	0.899	27.35	0.881	27.987	0.897	29.134	0.901	32.161	0.979	38.567
Image 3	0.891	26.695	0.868	27.79	0.869	28.431	0.896	28.989	0.911	31.782	0.991	37.993
Image 4	0.893	26.989	0.811	27.83	0.799	28.065	0.913	29.768	0.895	31.998	0.988	38.193
Image 5	0.891	26.888	0.898	27.36	0.898	27.431	0.894	29.451	0.957	32.943	0.978	38.093
Average:	0.885	26.680	0.871	27.30	0.886	28.371	0.902	29.2480	0.912	32.007	0.987	38.308

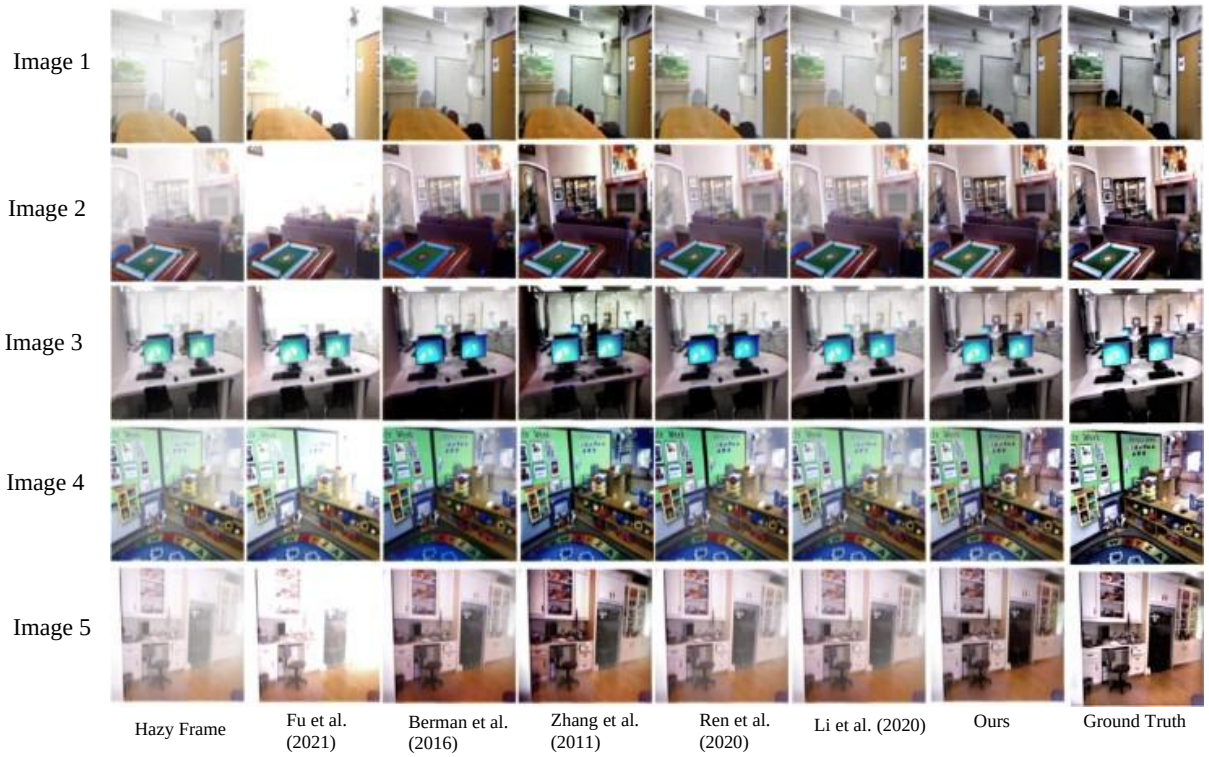


Figure 4. Comparative results (NYU depth dataset): the leftmost column displays a hazy image, while the rightmost column exhibits the ground truth of the picture. Moving from left to right, the remaining columns present the outputs produced by Zhang et al. (2011), Berman et al. (2016), Li et al. (2020), Ren et al. (2020), Fu et al. (2021) and our approach.

Figure 5 shows chart of the SSIM and PSNR tested for hazy frames shown in **Table 2**.

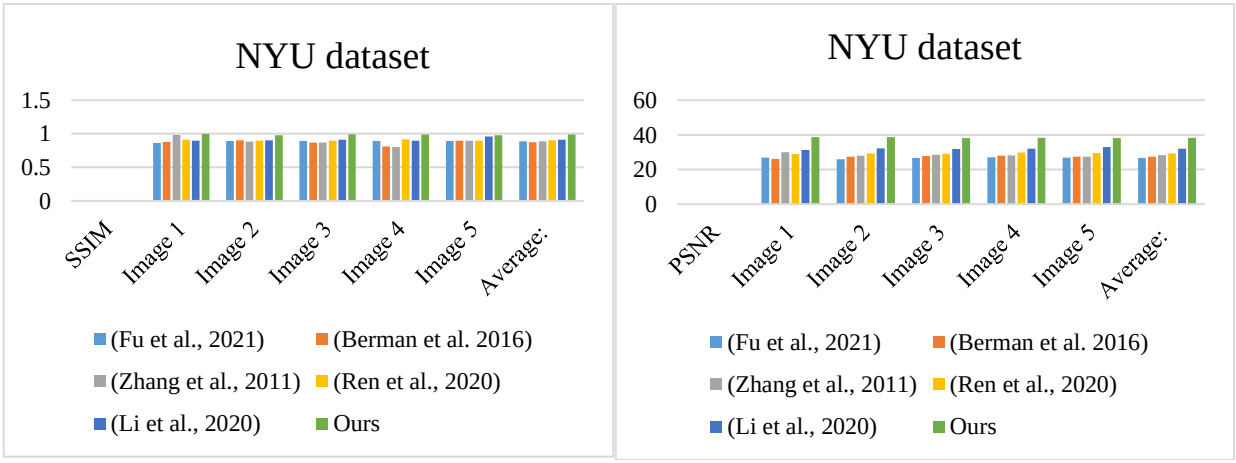


Figure 5. The SSIM and PSNR tested for the hazy frame shown in **Figure 4**.

D-HAZY and Self-Collected Dataset: After collecting five frames from the D-HAZY and Self-Collected Dataset, we tested various techniques, including our CNN model in **Figure 6**.

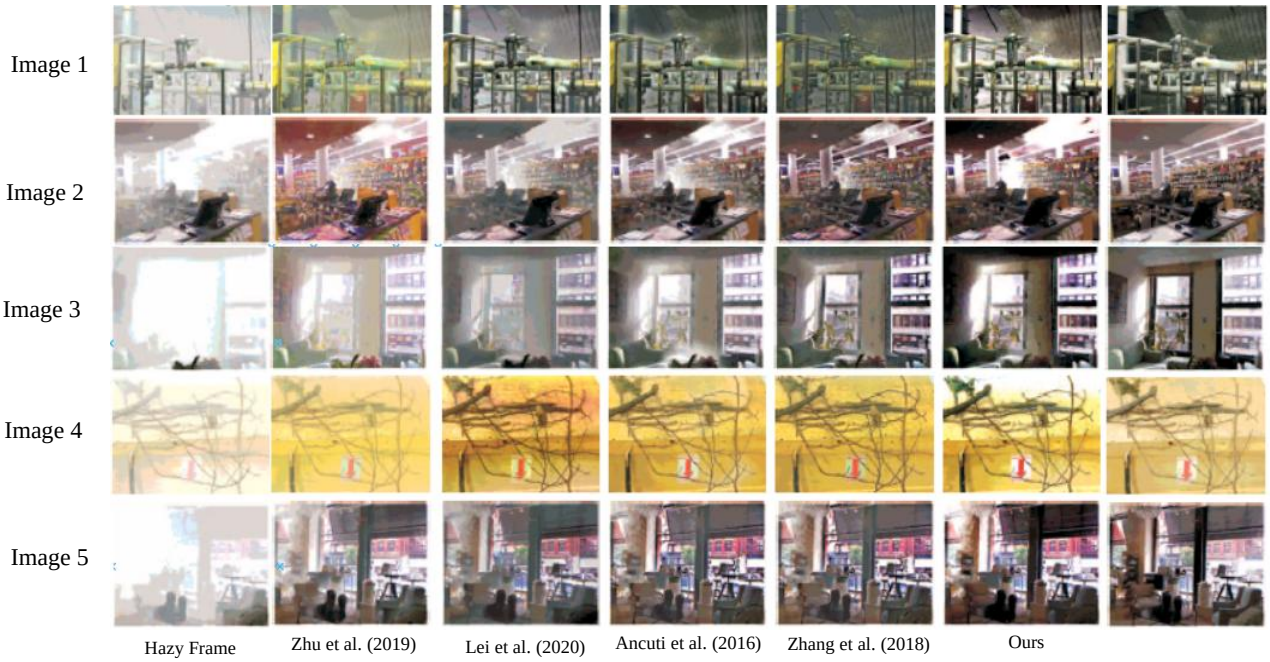


Figure 6. Comparative results (D-HAZY and Self-Collected Dataset): The leftmost column displays a hazy image, while the rightmost column exhibits the ground truth of the image. Moving from left to right, the remaining columns present the outputs produced by Ancuti et al. (2016), Zhang et al. (2018), Zhu et al. (2019), Lei et al. (2020) and Ours.

Our observations revealed that our CNN model surpassed the other methods, producing an average SSIM value of 0.987 and an average PSNR value of 39.359, as presented in **Table 3**. To provide a graphical representation of the quantitative comparison between the SSIM and PSNR values obtained from **Table 3**, we included **Figure 7**.

Table 3. Quantitative evaluation: PSNR and SSIM analysis on D-HAZY and self-collected dataset.

D-HAZY and self-collected dataset	Zhu et al. (2019)		Lei et al. (2020)		Ancuti et al. (2016)		Zhang et al. (2018)		Ours	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Image 1	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Image 2	0.978	36.324	0.989	35.372	0.933	23.087	0.945	38.923	0.986	39.467
Image 3	0.991	38.087	0.986	35.729	0.887	23.102	0.927	38.196	0.986	38.992
Image 4	0.891	37.235	0.899	35.183	0.885	23.087	0.980	38.761	0.980	39.792
Image 5	0.968	37.233	0.968	35.897	0.880	23.117	0.979	38.789	0.992	38.799
Average:	0.957	37.214	0.966	35.596	0.902	23.111	0.964	38.695	0.987	39.359

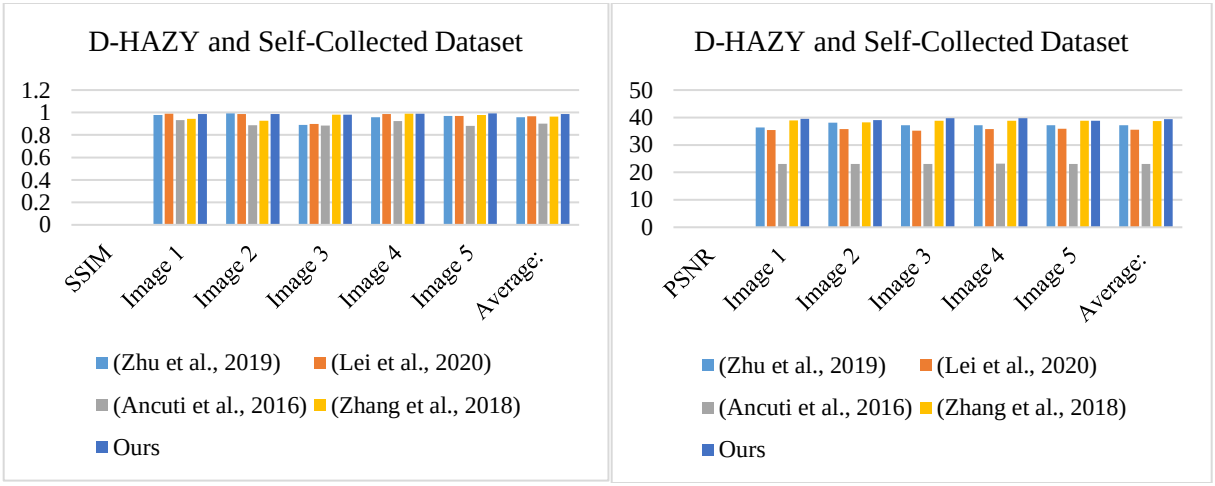


Figure 7. The SSIM and PSNR tested for the hazy frame shown in **Figure 6**.

NH-HAZE Dataset: Once we had obtained five frames from the NH-HAZE Dataset, we carried out tests using different techniques, including the CNN model illustrated in **Figure 8**. Our findings indicated that our CNN model outperformed other methods, resulting in an average SSIM value of 0.986 and an average PSNR value of 39.760, as showcased in **Table 4**. To visually depict the quantitative comparison between the SSIM and PSNR values obtained in **Table 4**, we have included **Figure 9**.

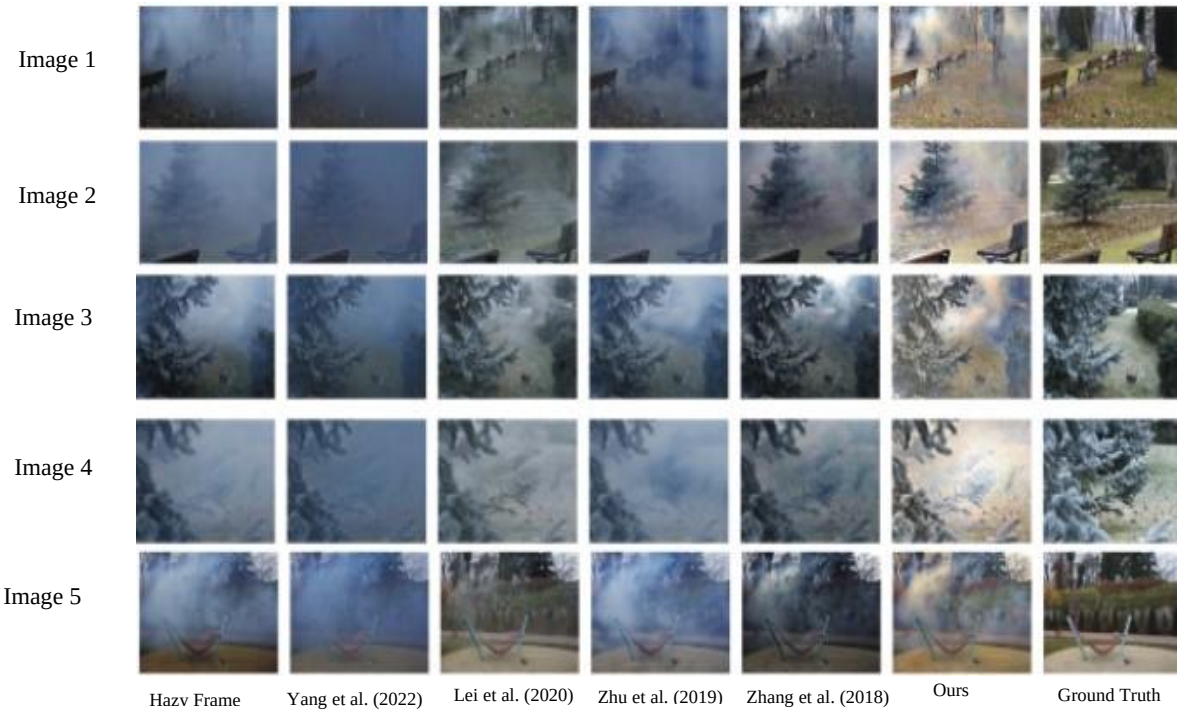


Figure 8. Comparative results (NH-HAZE Dataset): The leftmost column displays a hazy image, while the rightmost column exhibits the ground truth of the image. The remaining columns present the outputs produced by Zhang et al. (2018), Zhu et al. (2019), Lei et al. (2020), Yang et al. (2022) and Ours.

Table 4. Quantitative evaluation: PSNR and SSIM analysis on NH-HAZE dataset.

NH-HAZE dataset	Yang et al. (2022)		Lei et al. (2020)		Zhu et al. (2019)		Zhang et al. (2018)		Ours	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Image 1	0.956	28.729	0.935	35.543	0.955	Image 1	0.956	28.729	0.935	35.543
Image 2	0.893	28.591	0.979	35.812	0.932	Image 2	0.893	28.591	0.979	35.812
Image 3	0.889	28.846	0.841	35.098	0.979	Image 3	0.889	28.846	0.841	35.098
Image 4	0.921	27.747	0.965	35.818	0.951	Image 4	0.921	27.747	0.965	35.818
Image 5	0.948	28.917	0.991	35.782	0.980	Image 5	0.948	28.917	0.991	35.782
Average:	0.921	28.566	0.942	35.611	0.959	37.229	0.968	38.563	0.986	39.760

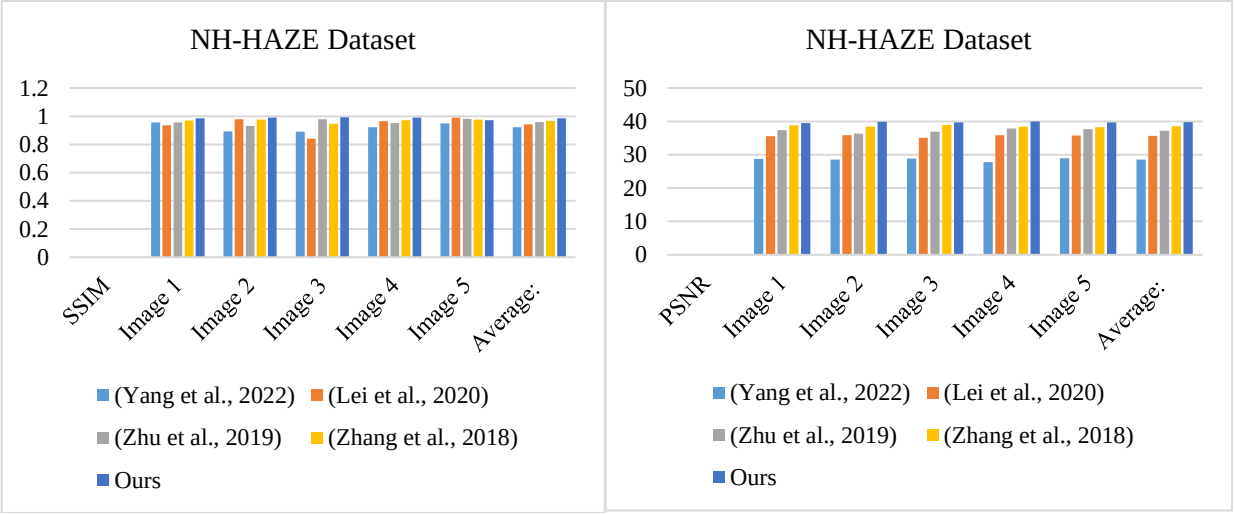


Figure 9. The SSIM and PSNR tested for hazy frame shown in **Figure 8**.

RESIDE Datasets: After obtaining five frames from the RESIDE datasets, we conducted several tests using different techniques, one of which was our CNN model illustrated in **Figure 10**. Our findings indicate that our CNN model outperformed other methods, achieving an average SSIM value of 0.989 and an average PSNR value of 38.024, as demonstrated in **Table 5**. To illustrate the visual comparison between the SSIM and PSNR values presented in **Table 5**, we have included **Figure 11**.

Table 5. Quantitative evaluation: PSNR and SSIM analysis on RESIDE dataset.

RESIDE dataset	Zhang et al. (2018)		Sahu et al. (2021)		Ren et al. (2020)		Ours	
	SSIM	PSNR	SSIM	PSNR	SSIM	SSIM	PSNR	SSIM
Image 1	0.867	21.40	0.871	25.24	Image 1	0.867	21.40	0.871
Image 2	0.853	21.35	0.885	25.37	Image 2	0.853	21.35	0.885
Image 3	0.868	21.66	0.872	25.55	Image 3	0.868	21.66	0.872
Image 4	0.861	21.16	0.888	25.78	Image 4	0.861	21.16	0.888
Image 5	0.852	21.24	0.889	25.45	Image 5	0.852	21.24	0.889
Average:	0.860	21.36	0.881	25.48	Average:	0.860	21.36	0.881

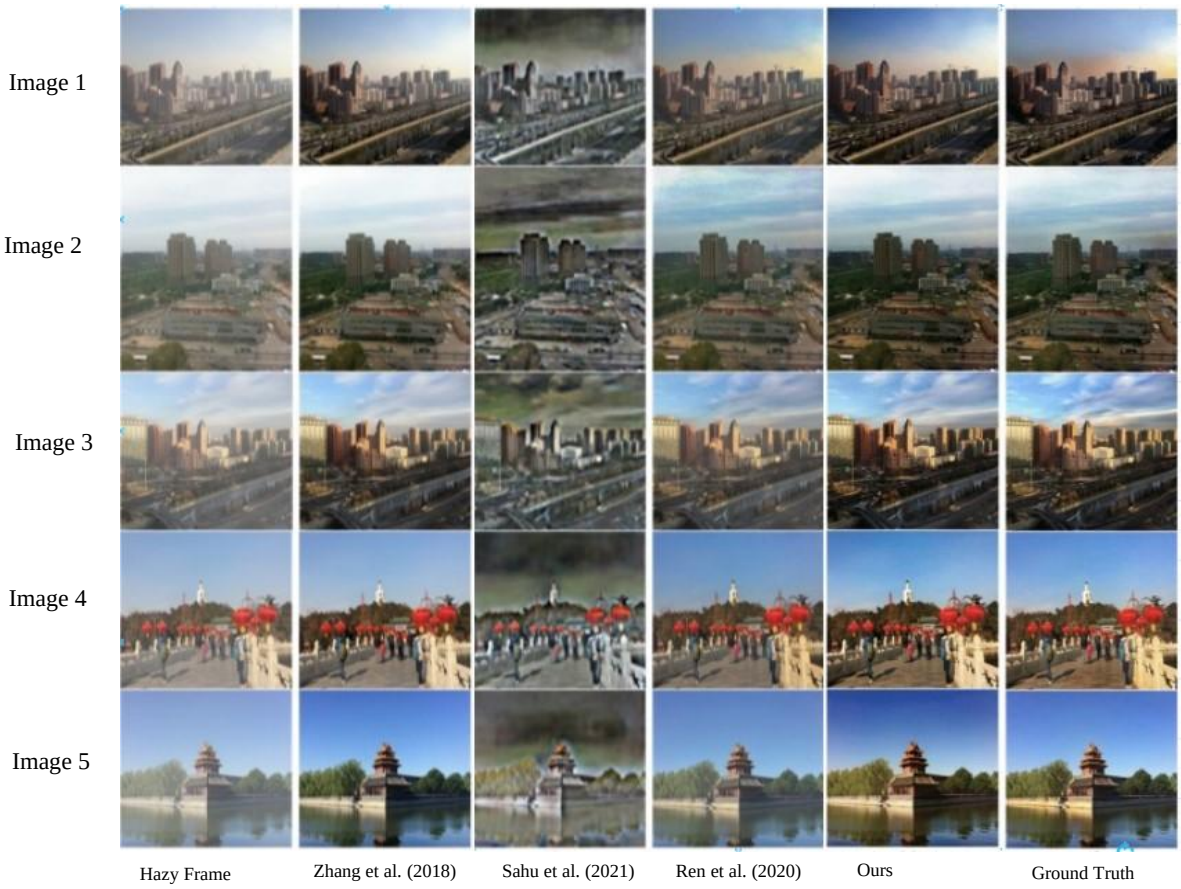


Figure 10. Comparative outcomes (RESIDE datasets): The leftmost column shows a hazy image, while the rightmost column exhibits the ground truth of the image. The remaining columns show the results of Zhang et al. (2018), Sahu et al. (2021), and our own.

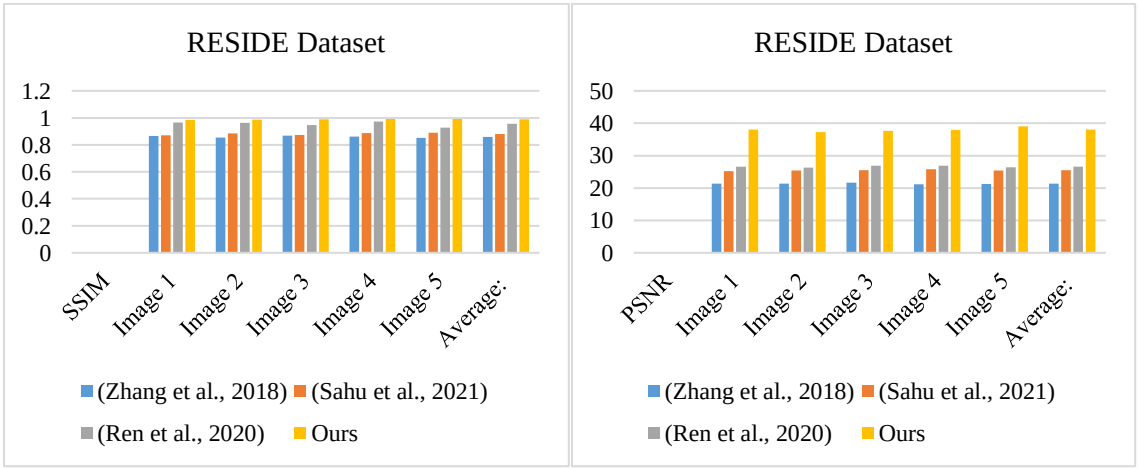


Figure 11. The SSIM and PSNR tested for hazy frame shown in **Figure 10**.

Following a thorough evaluation of our CNN methodology on multiple datasets such as NYU depth, D-HAZY, self-collected, NH-HAZE, and RESIDE, we are delighted to announce the exceptional performance achieved by our approach. Our analysis revealed remarkable results, with an average SSIM score of 0.987 across all datasets. This high score indicates a significant structural similarity between the original and reconstructed images. Moreover, the average PSNR score of 38.86 further validates the superior quality of the reconstructed images, showcasing minimal distortion. These findings highlight the effectiveness and robustness of our approach across diverse scenarios and datasets. For detailed analysis, please refer to **Table 6**. Utilizing a CNN-based model and four datasets, including NYU, D-HAZY, NH-HAZE, and RESIDE, proved crucial for training and validating the CNN model. The achieved SSIM and PSNR values of 0.987 and 38.86, respectively, indicate the model's effectiveness in image reconstruction. These outcomes demonstrate the potential of CNN-based models and the use of multiple datasets to enhance image reconstruction performance. Future research could explore the possibility of employing different architectures or additional datasets to further improve the performance of image reconstruction models.

Table 6. Quantitative analysis of our CNN approach on various datasets.

Our CNN approach	NYU dataset	D-HAZY and collected dataset	NH-HAZE dataset	RESIDE dataset	Average
SSIM	0.987	0.987	0.986	0.989	0.987
PSNR	38.31	39.36	39.76	38.02	38.86

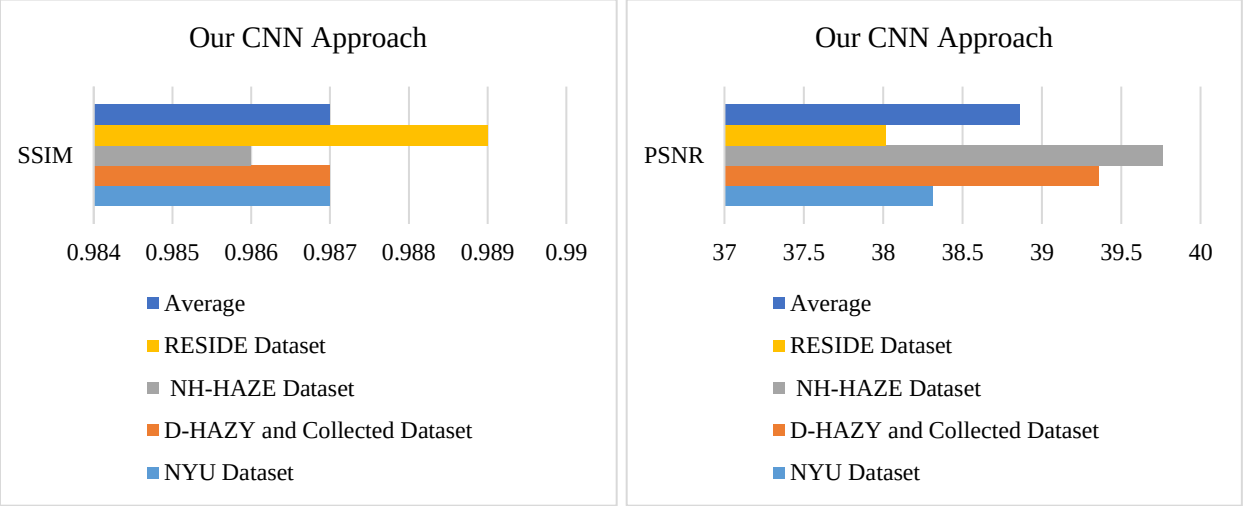


Figure 12. The SSIM and PSNR were tested for various datasets shown in **Table 6**.

5. Conclusion and Future Scope

The proposed work, describes and focuses on the development of an enhanced real-time video dehazing using CNNs. The trained model is then used on each frame of a hazy video sequence to get dehazed frames. From the results of the next sections, it is evident that several datasets were used to test the performance of the dehazing model as represented by the graph plots of the SSIM and PSNR. On average, the experiments yielded the PSNR and SSIM scores for all databases as 38. 86 and 0.987 respectively, proving the stability of the dehazing model regardless of the dataset used. Thus, the outcomes of this work have once again demonstrated the efficiency of the discussed dehazing approach in maintaining image quality and the relevant structural information in the course of dehazing.

The DeepVideoDehazeNet model effectively utilizes for real-time video dehazing, demonstrating high performance in preserving image quality and structural integrity across multiple datasets. Its adaptability to various real-world applications, such as autonomous driving and surveillance, emphasizes the model's worth. However, the work faces limitations in terms of hardware implementation for real-time processing and could benefit from integrating more advanced architectures and expanding the training dataset to handle more complex hazy videos, improving its efficiency in resource-constrained environments.

Future work for this approach includes improving the model's accuracy by exploring advanced architectures and Training methodologies. The training dataset can be expanded by incorporating additional datasets or creating new ones specifically for video dehazing. Optimizing preprocessing steps aims to enhance the dehazing performance through alternative or improved techniques. Exploring real-time hardware implementation involves implementing the model on different hardware platforms for better real-time performance. Additionally, the approach can be generalized to other applications, such as image denoising or video super-resolution, by adapting the underlying CNN architecture and Training methodology. These future endeavors have the potential to enhance the proposed approach, making it more robust, versatile, and applicable in various image and video processing tasks.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

AI Disclosure

During the preparation of this work the author(s) used generative AI in order to improve the language of the article. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that help improve the quality of this work.

References

Ancuti, C., Ancuti, C.O., & De Vleeschouwer, C. (2016). D-hazy: a dataset to evaluate quantitatively dehazing algorithms. In *2016 IEEE International Conference on Image Processing* (pp. 2226-2230). IEEE. Phoenix, AZ, USA.

Ancuti, C.O., & Ancuti, C. (2013). Single image dehazing by multi-scale fusion. *IEEE Transactions on Image Processing*, *22*(8), 3271-3282.

Ashwini, K., Nenavath, H., & Jatoth, R.K. (2022). Image and video dehazing based on transmission estimation and refinement using Jaya algorithm. *Optik*, *265*, 169565. https://doi.org/10.1016/j.ijleo.2022.169565.

Berman, D., Treibitz, T., & Avidan, S. (2016). Non-local image dehazing. In *2016 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1674-1682). IEEE, Las Vegas, NV, USA. https://doi.org/10.1109/CVPR.2016.185.

Bisen, L., & Dravid, A. (2014). Survey on haze removal techniques. *International Journal of Modern Trends in Engineering and Research*, *1*(5), 334-345.

Chen, Q., Xu, J., & Koltun, V. (2017). Fast image processing with fully-convolutional networks. In *2017 IEEE International Conference on Computer Vision* (pp. 2516-2525). IEEE, Venice, Italy. https://doi.org/10.1109/ICCV.2017.273.

Chiang, J.Y., Chen, Y.C., Singh, S.S., Sachdeva, R., Singh, A., Alenezi, F., Armghan, A., Mohanty, S.N., Jhaveri, R. H., Tiwari, P., Emberton, S., Chittka, L., Cavallaro, A., Fabbri, C., Islam, M.J., Sattar, J., Xia, Y., Sattar, J., Carlevaris-Bianco, N., & Mahmood, M.T. (2020). Initial results in underwater single image dehazing. *MTS/IEEE Seattle, OCEANS 2010*, *5*(2), 1-10. https://doi.org/10.1109/OCEANS.2010.5664428.

Emberton, S., Chittka, L., Cavallaro, A., Mao, J., Kumari, A., Sahdev, S., Sahoo, S.K., Li, B., Peng, X., Wang, Z.Z., Xu, J., Feng, D., Lu, H., Li, Y., Nakashima, S., Serikawa, S., Luan, Z., Zeng, H., Shang, Y., & Guo, J.I. (2018). Study on image dehazing with the self-adjustment of the Haze degree. *Mathematical Problems in Engineering*, *2018*(1), 1-8.

Fu, M., Liu, H., Yu, Y., Chen, J., & Wang, K. (2021). Dw-gan: A discrete wavelet transform gan for nonhomogeneous dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 203-212).

Han, J., Zhang, D., Cheng, G., Guo, L., & Ren, J. (2015). Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Transactions on Geoscience and Remote Sensing*, *53*(6), 3325-3337. https://doi.org/10.1109/TGRS.2014.2374218.

Hautière, N., Tarel, J.P., & Aubert, D. (2007). Towards fog-free in-vehicle vision systems through contrast restoration. In *2007 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-8). IEEE. Minneapolis, MN, USA. https://doi.org/10.1109/CVPR.2007.383259.

He, K., Sun, J., & Tang, X. (2011). Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(12), 2341-2353. https://doi.org/10.1109/TPAMI.2010.168.

He, K., Sun, J., & Tang, X. (2013). Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(6), 1397-1409. https://doi.org/10.1109/TPAMI.2012.213.

Joseph, J.E., & Gopakumar, G. (2020). A comprehensive review on image dehazing [J]. *International Journal of Engineering Research & Technology*, *9*(6), 1074-1077.

Kantipudi, M.P., Kumar, S., & Jha, A.K. (2021). Scene text recognition based on bidirectional LSTM and deep neural network. *Computational Intelligence and Neuroscience*, *2021*(1), 2676780. https://doi.org/10.1155/2021/2676780.

Kim, J.H., Jang, W.D., Sim, J.Y., & Kim, C.S. (2013). Optimized contrast enhancement for real-time image and video dehazing. *Journal of Visual Communication and Image Representation*, *24*(3), 410-425. https://doi.org/10.1016/j.jvcir.2013.02.004.

Kulkarni, A., Patil, P.W., & Murala, S. (2021). Progressive subtractive recurrent lightweight network for video deraining. *IEEE Signal Processing Letters*, *29*, 229-233. https://doi.org/10.1109/LSP.2021.3134171.

Kumar, R.P., & Naik, N.M. (2022). A review on efficient state-of-the-art image dehazing techniques. In *2022 International Conference on Innovations in Science and Technology for Sustainable Development* (pp. 115-121). IEEE. Kollam, India.

Lei, C., Xing, Y., & Chen, Q. (2020). Blind video temporal consistency via deep video prior. *Advances in Neural Information Processing Systems*, *2020-Decem*(NeurIPS), 1-11.

Li, B., Gou, Y., Liu, J.Z., Zhu, H., Zhou, J.T., & Peng, X. (2020). Zero-shot image dehazing. *IEEE Transactions on Image Processing*, *29*, 8457-8466. https://doi.org/10.1109/TIP.2020.3016134.

Li, F., Di, X., Zhao, C., Zheng, Y., & Wu, S. (2022). FA-GAN: a feature attention GAN with fusion discriminator for non-homogeneous dehazing. *Signal, Image and Video Processing*, *16*(5), 1243-1251. https://doi.org/10.1007/s11760-021-02075-1.

Lv, X., Chen, W., & Shen, I.F. (2010). Real-time dehazing for image and video. In *2010 18th Pacific Conference on Computer Graphics and Applications* (pp. 62-69). IEEE. Hangzhou, China.

Ma, S., Pan, W., Liu, H., Dai, S., Xu, B., Xu, C., & Guan, H. (2023). Image dehazing based on improved color channel transfer and multiexposure fusion. *Advances in Multimedia*, *2023*(1), 8891239. https://doi.org/10.1155/2023/8891239.

Morales, P., Klinghoffer, T., & Jae Lee, S. (2019). Feature forwarding for efficient single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (pp. 2078-2085). https://doi.org/10.1109/CVPRW.2019.00260.

Narasimhan, S.G., & Nayar, S.K. (2003). Contrast restoration of weather degraded images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(6), 713-724. https://doi.org/10.1109/TPAMI.2003.1201821.

Pan, Y., Bano, S., Vasconcelos, F., Park, H., Jeong, T.T., & Stoyanov, D. (2022). DeSmoke-LAP: improved unpaired image-to-image translation for desmoking in laparoscopic surgery. *International Journal of Computer Assisted Radiology and Surgery*, *17*(5), 885-893. https://doi.org/10.1007/s11548-022-02595-2.

Parihar, A.S., Gupta, Y.K., Singodia, Y., Singh, V., & Singh, K. (2020). A comparative study of image dehazing algorithms. In *2020 5th International Conference on Communication and Electronics Systems* (pp. 766-771). IEEE. Coimbatore, India.

Park, Y., & Kim, T.H. (2018). Fast execution schemes for dark-channel-prior-based outdoor video dehazing. *IEEE Access*, *6*, 10003-10014. https://doi.org/10.1109/ACCESS.2018.2806378.

Ren, W., Pan, J., Zhang, H., Cao, X., & Yang, M.H. (2020). Single image dehazing via multi-scale convolutional neural networks with holistic edges. *International Journal of Computer Vision*, *128*(1), 240-259. https://doi.org/10.1007/s11263-019-01235-8.

Ren, W., Zhang, J., Xu, X., Ma, L., Cao, X., Meng, G., & Liu, W. (2018). Deep video dehazing with semantic segmentation. *IEEE Transactions on Image Processing*, *28*(4), 1895-1908.

Sahu, G., Seal, A., Krejcar, O., & Yazidi, A. (2021). Single image dehazing using a new color channel. *Journal of Visual Communication and Image Representation*, *74*, 103008. https://doi.org/10.1016/j.jvcir.2020.103008.

Schechner, Y.Y., & Averbuch, Y. (2007). Regularized image recovery in scattering media. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(9), 1655-1660. https://doi.org/10.1109/TPAMI.2007.1141.

Shao, S., Guo, Y., Zhang, Z., & Yuan, H. (2019). Single remote sensing multispectral image dehazing based on a learning framework. *Mathematical Problems in Engineering*, *2019*(1), 4131378. https://doi.org/10.1155/2019/4131378.

Song, Y., He, Z., Qian, H., & Du, X. (2023). Vision transformers for single image dehazing. *IEEE Transactions on Image Processing*, *32*, 1927-1941. https://doi.org/10.1109/TIP.2023.3256763.

Tan, R.T. (2008). Visibility in bad weather from a single image. In *2008 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-8). IEEE. Anchorage, AK, USA.

Tran, L.A., Moon, S., & Park, D.C. (2022). A novel encoder-decoder network with guided transmission map for single image dehazing. *Procedia Computer Science*, *204*, 682-689. https://doi.org/10.1016/j.procs.2022.08.082.

Tripathi, A.K., & Mukhopadhyay, S. (2012). Single image fog removal using anisotropic diffusion. *IET Image Processing*, *6*(7), 966-975. https://doi.org/10.1049/iet-ipr.2011.0472.

Wang, R., Li, R., & Sun, H. (2016). Haze removal based on multiple scattering model with superpixel algorithm. *Signal Processing*, *127*, 24-36. https://doi.org/10.1016/j.sigpro.2016.02.003.

Wang, Y., Yan, X., Wang, F.L., Xie, H., Yang, W., Zhang, X.P., Qin, J., & Wei, M. (2024). UCL-dehaze: toward real-world image dehazing via unsupervised contrastive learning. *IEEE Transactions on Image Processing*, *33*, 1361-1374.

Wu, D., Zhu, Q., Wang, J., Xie, Y., & Wang, L. (2014). Image haze removal: status, challenges and prospects. In *2014 4th IEEE International Conference on Information Science and Technology* (pp. 492-497). IEEE. Shenzhen, China.

Xu, J., Hu, X., Zhu, L., Dou, Q., Dai, J., Qiao, Y., & Heng, P.A. (2023). Video dehazing via a multi-range temporal alignment network with physical prior. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp.18053-18062). https://doi.org/10.1109/CVPR52729.2023.01731.

Yang, G., & Evans, A.N. (2021). Improved single image dehazing methods for resource-constrained platforms. *Journal of Real-Time Image Processing*, *18*(6), 2511-2525. https://doi.org/10.1007/s11554-021-01143-6.

Yang, G., Evans, A.N., Agrawal, S.C., & Jalal, A.S. (2022). A Comprehensive review on analysis and implementation of recent image dehazing methods. *Archives of Computational Methods in Engineering*, *29*(7), 4799-4850. https://doi.org/10.1007/s11831-022-09755-2.

Zeng, F., Wu, Q., & Du, J. (2014). Foggy image enhancement based on filter variable multi-scale retinex. *Applied Mechanics and Materials*, *505*,1041-1045. https://doi.org/10.4028/www.scientific.net/AMM.505-506.1041.

Zhang, J., Li, L., Zhang, Y., Yang, G., Cao, X., & Sun, J. (2011). Video dehazing with spatial and temporal coherence. *Visual Computer*, *27*(6-8), 749-757. https://doi.org/10.1007/s00371-011-0569-8.

Zhang, R., Isola, P., Efros, A.A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 586-595). https://doi.org/10.1109/CVPR.2018.00068.

Zhang, X., Wang, T., Luo, W., & Huang, P. (2020). Multi-level fusion and attention-guided CNN for image dehazing. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(11), 4162-4173. https://doi.org/10.1109/TCSVT.2020.3046625.

Zhao, D., Li, J., Li, H., & Xu, L. (2021). Complementary feature enhanced network with vision transformer for image dehazing. *arXiv preprint arXiv:2109.07100*.

Zhu, X., Li, Z., Zhang, X.Y., Li, C., Liu, Y., & Xue, Z. (2019). Residual invertible spatio-temporal network for video super-resolution. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 5981-5988. https://doi.org/10.1609/aaai.v33i01.33015981.