



# Atrous Convolution-Based Residual Deep CNN for Image Dehazing with Spider Monkey-Particle Swarm Optimization

CH. Mohan Sai Kumar\* and R. S. Valarmathi

Department of Electronics and Communication Engineering, VelTech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, 600062, India

\*Corresponding Author: CH. Mohan Sai Kumar. Email: mohansaikumarchadalavada@gmail.com

Received: 28 November 2022; Accepted: 13 April 2023; Published: 23 June 2023

**Abstract:** Image dehazing is a rapidly progressing research concept to enhance image contrast and resolution in computer vision applications. Owing to severe air dispersion, fog, and haze over the environment, hazy images pose specific challenges during information retrieval. With the advances in the learning theory, most of the learning-based techniques, in particular, deep neural networks are used for single-image dehazing. The existing approaches are extremely computationally complex, and the dehazed images are suffered from color distortion caused by the over-saturation and pseudo-shadow phenomenon. However, the slow convergence rate during training and haze residual is the two demerits in the conventional image dehazing networks. This article proposes a new architecture “Atrous Convolution-based Residual Deep Convolutional Neural Network (CNN)” method with hybrid Spider Monkey-Particle Swarm Optimization for image dehazing. The large receptive field of atrous convolution extracts the global contextual information. The swarm based hybrid optimization is designed for tuning the neural network parameters during training. The experiments over the standard synthetic dataset images used in the proposed network recover clear output images free from distortion and halo effects. It is observed from the statistical analysis that Mean Square Error (MSE) decreases from 74.42 to 62.03 and Peak Signal to Noise Ratio (PSNR) increases from 22.53 to 28.82. The proposed method with hybrid optimization algorithm demonstrates a superior convergence rate and is a more robust than the current state-of-the-art techniques.

**Keywords:** Image dehazing; computer vision; convolutional neural network; color distortion; over-saturation; pseudo-shadow phenomenon; convergence rate

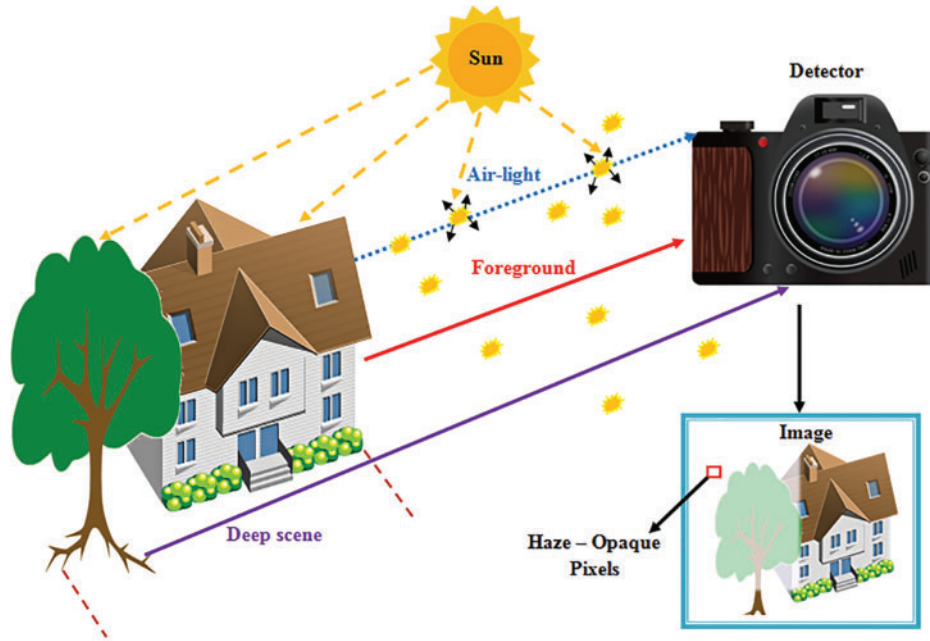
## 1 Introduction

Haze is regarded as a phenomenon due to the atmosphere, in which the dry particles, dust, or aerosols reduce the image scene quality. Haze generally makes specific degradation in the scene quality of the image, capping the total color contrast and making occlusion of objects present in the scene due



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

to ambient light [1]. The haze effect [2] causes low visual quality and diminished color contrast in real-world images taken in cloudy weather conditions. Various computer vision tasks like autonomous driving system and Land Use Land Cover (LULC) detection is performed by monitoring the real-world images and scene evaluation with satellite images that rely on the clean images to obtain accurate results. Therefore it is essential to reduce the haze effect over the captured images to enhance the images visual quality and the process of object recognition and detection [3]. The image haze formation model [4] is shown in Fig. 1.



**Figure 1:** Image haze formation

According to the theory of haze model [5,6], the influence of atmospheric light on natural images can be described as in Eqs. (1) and (2) by

$$I_{(m,n)} = J_{(m,n)} \cdot T_{(m,n)} + A(1 - T_{(m,n)}) \quad (1)$$

$$T_{(m,n)} = e^{-\beta d_{(m,n)}} \quad (2)$$

Here,  $I$  and  $J$  are hazy and haze-free images,  $T$  is the transmission map connected with  $\beta$  (atmospheric scattering coefficient), and  $d$  (depth between the camera and scene),  $A$  is the global ambient light (atmospheric light) at pixel location  $(m, n)$ . The haze-free image  $J$  is was obtained by transforming Eq. (1) into the following Eq. (3).

$$J_{(m,n)} = \frac{I_{(m,n)}}{T_{(m,n)}} - A \left( 1 - \frac{1}{T_{(m,n)}} \right) \quad (3)$$

$$I_{(m,n)} = J_{(m,n)}, \text{ when } d_{(m,n)} \rightarrow 0, T_{(m,n)} \rightarrow 1 \quad (4)$$

According to Eq. (4) the captured image will be the actual radiance from the scene. But, in the case of terrestrial imaging, the scene depth is very high, and the influence of atmospheric light will be

more as described by Eq. (5).

$$I_{(m,n)} = A, \text{ when } d_{(m,n)} \rightarrow \infty, T_{(m,n)} \rightarrow 0 \quad (5)$$

Both the cases of Eqs. (4) and (5) are not possible in practical situations. The estimation of  $T_{(m,n)}$  and  $A$  are essential in obtaining the dehazed image. Accurate estimation of  $A$  obtains the detailed dehazed image without losing the microscopic details from the high-intensity regions of the hazy input image.

Many existing model-based dehazing methods [7–9] propose the complex prior regularizations for the approximation of  $A$  and  $T_{(m,n)}$  for extracting the spatial features from the hazy input, but the approximate image will still be different from the haze-free image. These methods are time-consuming, involve huge parameters, and need help in the optimization.

To overcome the dependence on approximated prior, the design of learning-driven methods using deep learning techniques is promoted. The deep Convolutional Neural Network (CNN) model [10] maps the atmospheric scattering parameters learned from the network training to recover the image from haze effects. Furthermore, wavelet techniques are also used in image dehazing models, where Discrete Wavelet Transform (DWT) [11] divides the images into patches using various wavelet sub-bands or frequencies. To retain the high-frequency information and prevent the pseudo-shadow phenomenon in the dehazed image, the SSIM loss of each image patch is calculated, and the weights are updated.

The image dehazing aims to minimize the haze effect and enhance the contrast levels by maintaining the very fine details at the edges, in this work a new deep learning method for image dehazing using a hybrid swarm optimization algorithm is proposed. The structure of this article is as follows. In Section 2, the existing works related to the image dehazing are introduced. The architectural flow of the proposed image dehazing framework is described along with the appropriate datasets in Section 3. Section 4 defines the wavelet image decomposition and hybrid Spider Monkey Particle Swarm Optimization (SMPSO) algorithm. Section 5 discusses the image dehazing model using Atrous Convolution-based Residual Deep CNN. Section 6 provides the experimental settings, results, and ablation analysis. Section 7 outlines the conclusion and future scope of this work.

## 2 Related Works

The image dehazing problem is considered as the “No-Reference (NR) Image Quality Assessment (IQA)” method. A superior way of performing image dehazing is to design specific quantitative evaluation metrics. However, this is challenging due to the complications in the image dehazing process as various dehazing algorithms (DHAs) have unique haze effects.

Most image restoration methods from haze rely on the air-light scattering model [12] due to the performance attained in processing the hazy daytime images. These methods are classified into prior-based computation models [5,13] and learning-based prediction models [14–16].

Based on the number of images used for dehazing process, the dehazing methods are classified into multiple images and single image-based haze removal methods. Polarization-based methods [17] use varying degrees of polarization of different images to restore the scene depth. Contrast enhancement methods [18,19], and Multiscale fusion [20] algorithms require a single image which depends on the distinctive traits of the ground truth image.

The application of deep CNN has been increased in the recent research works of image dehazing and achieved a remarkable throughput. These methods result in better recovery of explicit scenes in

the image from the haze effects by heuristically learning the complex relations from the haze data observations.

Li et al. [21] proposed calculating the ratio of the foggy image to transmission map estimate with residual-based deep CNN and avoiding estimating atmospheric light. Li et al. [22] present an accurate estimation of atmospheric light and strive to remove the bright region's effect on estimation using quad-tree decomposition. Liu et al. [23] developed an agnostic CNN model to fully eliminate the necessity for parameter estimation and require no insight into the atmospheric scattering model using local and global residual learning stages. Ren et al. [24] method learn the relation amid the hazy input and the corresponding scene depth using a multiscale deep neural network.

Gan et al. [25] propose to jointly estimate the air-light and transmission maps through a Conditional Generative Adversarial Network (CGAN). Zhang et al. [26] recover the clear scene through high-level features and the scene details from low-level feature maps using Deep Residual Convolutional Dehazing Network (DRCDN). Wang et al. [27] designed an enhanced network based on a CNN-based dehazing model with three different modules within the network. They jointly estimated the air-light and feature map to enhance the ability of CNN. Dharejo et al. [28] proposed using CNN with a wavelet hybrid network in the wavelet domain. The local features and global features are achieved from the multi-level resolution of hazy images.

Yang et al. [29] developed a new deep-learning method combining haze-related prior information and deep-learning architectures perspectives. Wang et al. [30] developed a module for feature extraction using the deep residual haze network and recovers the clear image by subtracting the trained residual transmission map and the corresponding hazy input. Min et al. [31] proposed a DHA quality evaluation method with the integration of haze-relevant features. The technique works better for aerial image databases with image structure recovering and enhancing the low-contrast areas. Yu et al. [32] use the concept of multi-resolution segmentation with an image fusion process for transmission map estimation. Bilal et al. [33,34] reduces the degradation effects in the images using evolutionary optimization algorithm and weighted filter with dilated convolution and watershed segmentation.

### 3 Proposed Methodology

#### 3.1 Architectural Flow of the Proposed Framework

The combined effect of haze and reflected light from the scene object reduces the contrast, makes the objects in the scene challenging to identify, and creates difficulty in further processing. The learning-based methods utilize machine learning frameworks for apparent image recovery. The deep neural network is designed to learn the scene depth and haze-based features for the ground truth image.

This article proposes a novel atrous convolution-based residual deep CNN using a hybrid SMPPO algorithm to recover the clear image from hazy input. The proposed framework consists of three major blocks viz., (1) forward wavelet transform, (2) residual deep CNN with atrous convolution using an optimization algorithm, and (3) inverse wavelet transform for reconstructing the dehazed image.

The dataset images are collected and given to the wavelet transform block. It decomposes the input images into four separate frequency-classified sub-bands. The low-frequency sub-image patch contains very fine details and most of the details of the input image, and this image patch will be the input to the deep CNN block.

The proposed network consists of fewer convolution layers, namely, convolution layers (vanilla convolution) and atrous convolution layers (dilated convolution), at the offset of not compromising

the performance with the other learning models. This type of filter used in network learning eliminates the possibility of blur in the obtained residual image. The hybrid optimization algorithm optimizes the batch size, epochs, and learning rate during the network training. The residual image was further combined with high-frequency decomposed patches using IDWT to obtain the haze-free output.

Fig. 2 shows the architectural flow of the proposed atrous convolution-based residual deep CNN framework using a hybrid SMPSO algorithm.

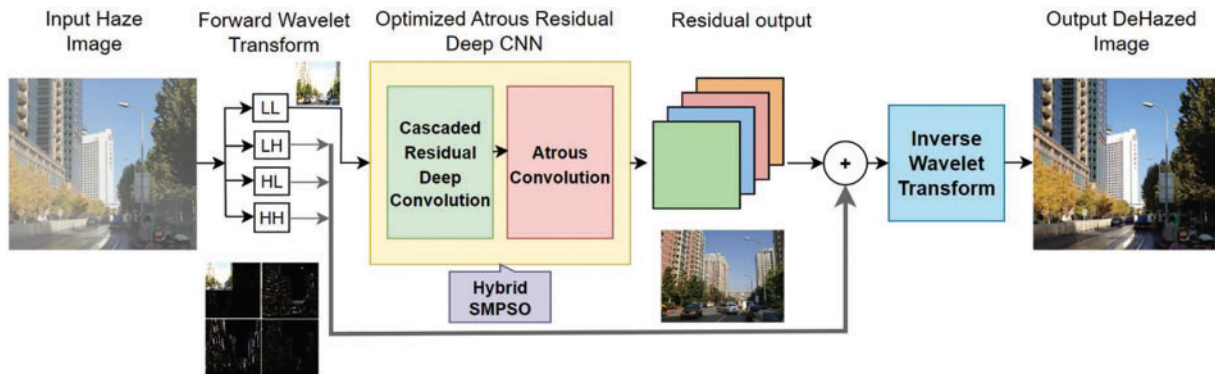


Figure 2: Architectural flow of the proposed image dehazing framework

### 3.2 Image Dehazing Dataset Collection

The above image dehazing framework is implemented on two different datasets: Synthetic Objective Testing Set (SOTS) [RESIDE], and I-Haze of NTIRE 2018. The details of the datasets are provided below.

**SOTS [RESIDE]:** The Synthetic Objective Testing Set (SOTS) RESIDE dataset consists of 500 hazy training images under indoor and outdoor subsets, and 50 indoor and 492 outdoor ground truth images. The dataset is accessed from the link “<https://www.kaggle.com/datasets/balraj98/synthetic-objective-testing-set-sots-reside>”, access date: 2022–05–24.

**I-Haze Dataset:** The I-Haze dataset is deployed in NTIRE 2018 CVPR dehazing challenge with 30 hazy and ground truth images respectively. The dataset is accessed from the link “<https://data.vision.ee.ethz.ch/cvl/ntire18/i-haze/>”, access date: 2022–06–19.

## 4 Wavelet-Based Image Decomposition and Hybrid Swarm Optimization

### 4.1 DWT-Based Image Decomposition

In the proposed method, before dehazing, DHWT is used to decompose the original hazy input image [35]. As a result, the four distinct sub-bands: approximation (LL), horizontal (LH), vertical (HL), and diagonal (HH) corresponding to the respective wavelet sub-band coefficient extract the information from the hazy input. The LL sub-band corresponds to the hazy input’s down-sampled version ( $\downarrow 2$ ). The LH and HL sub-bands tend to isolate the horizontal and vertical features, and the HH sub-band preserves the corresponding localized high-frequency point features in hazy input. The low-frequency sub-image is considered for dehazing using atrous convolution-based residual deep CNN, and the high-frequency sub-images retain the high-frequency hazy input details. In the end, the clear image was recovered by applying the IDWT for the combined residual image and the high-frequency component of the wavelet transform.

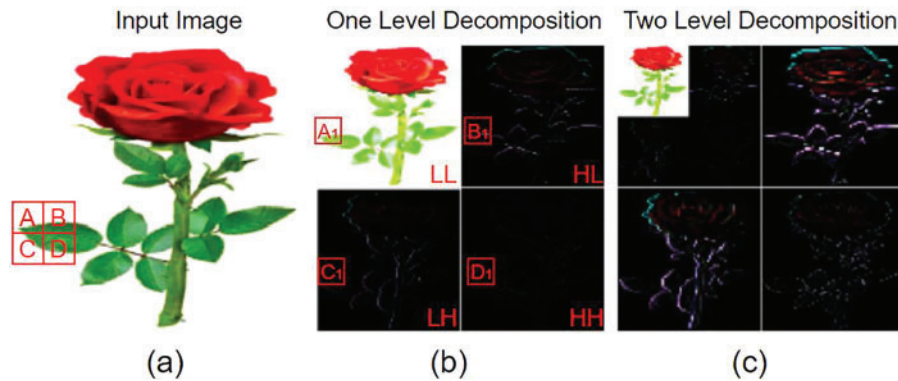
Corresponding to four wavelet sub-bands seen above, one scaling function  $\phi_{LL}(cc, dd)$  and three wavelet functions ( $\psi_{LH}(cc, dd)$ ,  $\psi_{HL}(cc, dd)$  and  $\psi_{HH}(cc, dd)$ ) are shown in Eq. (6).

$$\left. \begin{aligned} \phi_{LL}(cc, dd) &= \phi_L(cc) \phi_L(dd) \\ \psi_{LH}(cc, dd) &= \psi_L(cc) \psi_H(dd) \\ \psi_{HL}(cc, dd) &= \psi_H(cc) \psi_L(dd) \\ \psi_{HH}(cc, dd) &= \psi_H(cc) \psi_H(dd) \end{aligned} \right\} \quad (6)$$

The IDWT coefficients can be obtained through the Haar wavelet by using the Eq. (7).

$$\left. \begin{aligned} A &= A_1 + B_1 + C_1 + D_1 \\ B &= A_1 - B_1 + C_1 - D_1 \\ C &= A_1 + B_1 - C_1 - D_1 \\ D &= A_1 - B_1 - C_1 - D_1 \end{aligned} \right\} \quad (7)$$

here, A, B, C, and D are the input image pixel values and  $A_1$ ,  $B_1$ ,  $C_1$ , and  $D_1$  are the corresponding one-level wavelet decomposed image pixel values as shown in Fig. 3.



**Figure 3:** (a) Input image, (b) one-level decomposition of the input image, (c) two-level decomposition of the same

#### 4.2 Meta-Heuristic Swarm-Based Hybrid SMPSO Algorithm

A single optimization algorithm cannot address the full aspects of all search and optimization issues. Because each optimization will have distinct merits and demerits, combining the features of these optimizations enable to development of a hybrid algorithm for better robustness and more flexibility to solve complicated issues [36]. The hybrid SMPSO algorithm combines the PSO and SMO algorithms, is presented in this section.

##### 4.2.1 Particle Swarm Optimization (PSO) Algorithm

The swarm intelligence-based algorithms have gained wide popularity and are addressed in many engineering optimization issues. PSO is an adaptable and particle-based heuristic search optimization for solving engineering optimization problems proposed by Eberhart, R. Kennedy, J and Shi [37,38].

According to PSO, any problem is computationally expensive and can be solved by finding the best solution that fits into it using an exploration and exploitation strategy. The algorithm is broadly accepted due to its superior and fast convergence to optimum, high accuracy, and low memory requirements. The optimization randomly initializes a set of particles (candidate solutions) called a

swarm flying in the search space of z-dimension. The vector form of velocity  $V_i$  is  $V_i = (v_{i1}, \dots, v_{iz})$  and position  $X_i$  is  $X_i = (x_{i1}, \dots, x_{iz})$  of each particle  $i$  and will update based on the particle's individual experience and that of its neighbors as in Eqs. (8.a) and (8.b).

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_{b_i}^k - X_i^k) + c_2 r_2 (G_{b_i}^k - X_i^k) \quad (8.a)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (8.b)$$

where  $i = 0, 1, \dots, N-1$ ,  $V_i^k$  and  $X_i^k$  is the current velocity and position of the particle  $i$  at  $k^{th}$  iteration,  $V_i^{k+1}$  is updated velocity of particle  $i$  at  $(k+1)^{th}$  iteration with inertia weight  $\omega$ , acceleration coefficients ( $c_1$  and  $c_2$ ), and  $r_1, r_2$  are evenly distributed arbitrary numbers between (0, 1) generated for each iteration. The best possible position for particle  $i$  is  $P_{b_i}^k$  and the best possible position among the population until  $k^{th}$  iteration is  $G_{b_i}^k$ .

Every particle  $i$  will update using  $y_i$ , the best solution for particle  $i$  in the local search space, and  $y^*(k)$ , the position that yields the best solution among all the  $y_i$ 's in the global search space. The  $P_{b_i}$  and  $G_{b_i}$  values will be updated at iteration  $k$  with a swarm of  $N$  particles using Eqs. (9) and (10).

$$y_i(k+1) = \begin{cases} x_i(k+1), & \text{if } f(y_i(k)) > f(P_{b_i}) \\ y_i(k), & \text{if } f(y_i(k)) \leq f(P_{b_i}) \end{cases} \quad (9)$$

$$y^*(k) \in \{y_1(k), y_2(k), \dots, y_N(k)\},$$

$$f(y^*(k)) = \min\{f(y_1(k)), f(y_2(k)), \dots, f(y_N(k))\} \quad (10)$$

The acceleration coefficients control the particle speed at each iteration. In Eq. (11) the time-varying inertia weight  $\omega$  controls the PSO algorithm's convergence and linearly decreases over time. The high value of inertia weight during the early stages of search facilitates the chance of global exploration (visiting more positions in the search range) and low value in the later stages facilitates local exploration (for local search ability).

$$\text{inertia weight } \omega = (\omega_1 - \omega_2) \left( \frac{\text{max.allowable iter. iter.current}}{\text{iter.current}} \right) + \omega_2 \quad (11)$$

where  $\omega_1$  and  $\omega_2$  represent the inertia weight's initial and final values respectively.

#### 4.2.2 Spider Monkey Optimization (SMO) Algorithm

It is a swarm intelligence-based optimization influenced by the spider monkeys' Fission-Fusion Social (FFS) behavior during the foraging process. The characteristics of the SMO algorithm [39] depend on the grouping of monkeys into groups from small to large and vice-versa. The SMO contains four phases: Local Leader Phase (LLP), Global Leader Phase (GLP), Learning Phase, and Decision phase.

Let  $X_i$  represents the  $i^{th}$  individual in a  $k$ -dimension vector of the swarm population, each  $i^{th}$  individual is initialized as in Eq. (12):

$$X_i^k = X_{min}^k + U(0, 1) \times (X_{max}^k - X_{min}^k) \quad (12)$$

where  $X_{min}^{kj}$  and  $X_{max}^k$  are the boundary values of the search space.  $U(0, 1)$  is an evenly distributed arbitrary number between (0, 1).

During LLP, the individuals adapt their positions using the group member's and local leader knowledge in their group. The updation in new position is based on the individual's fitness and the perturbation rate probability. The new position update Eq. (13) is:

$$X_i^{k+1} = X_i^k + U(0, 1) \times (LL_m^k - X_i^k) + U(-1, 1) \times (X_r^k - X_i^k) \quad (13)$$

where  $X_i^{k+1}$  is the  $i^{th}$  individual in  $k^{th}$  dimension,  $LL_m^k$  represents the local leader, and  $X_r^k$  is the arbitrary selected individual from the  $m^{th}$  group so that  $r \neq i$  in  $k^{th}$  dimension of the search space,  $U(-1, 1)$  is an evenly distributed arbitrary number in the range  $(-1, 1)$  and perturbation rate varies in between  $[0.1, 0.8]$ .

During the Global Leader phase, the individuals update their position using the global leader, members' knowledge in the group, and their own persistence. The new position update equation is given in Eq. (14):

$$X_i^{k+1} = X_i^k + U(0, 1) \times (GL_m^k - X_i^k) + U(-1, 1) \times (X_r^k - X_i^k) \quad (14)$$

where the  $GL_m^k$  represents the global leader of the  $m^{th}$  group in  $k^{th}$  dimension.

During the Learning Phase, the best individual is recognized as the swarms' global leader, and the individual with the best fitness is recognized as the local leader of the particular group. If the global leader and local leader fail to update their position, then the global limit counter and the local limit counter is incremented by one.

During the Decision phase, all the group members in the local group initialize randomly or update their position employing global and local leader information as in Eq. (15):

$$X_i^{k+1} = X_i^k + U(0, 1) \times (GL_m^k - X_i^k) + U(0, 1) \times (X_r^k - LL_m^k) \quad (15)$$

The global leader decides on creating the smaller subgroups or fusing the small groups into a large group based on a predefined maximum number of groups.

#### 4.2.3 Hybrid Spider Monkey Particle Swarm Optimization (SMPSO) Algorithm

The primary objective of the hybrid Spider Monkey Particle Swarm Optimization (SMPSO) is to increase the Spider Monkey Optimizer's exploration with Particle Swarm Optimization's exploitation capabilities. It is a co-evolutionary technique since both techniques operate in parallel, not one after another. The first step is initializing each of PSO's original parameters. Further the position and velocity of the population are randomly initialized within the predetermined ranges. The next step is calculating each particle's fitness function, global best ( $G_{b_i}$ ) and local best ( $P_{b_i}$ ). After that, the fitness of the particle is evaluated with the current iteration to its previous iteration as per Eq. (16). If the fitness value is improved, the SMO algorithm will take over the other. Otherwise, the PSO algorithm continues following Eqs. (8.a) and (8.b) with the updated inertia weight  $\omega$  in Eq. (17).

$$y(i, k + 1) = \begin{cases} true, & \text{if } f(y_i(k + 1)) > f(y_i(k)) \\ false, & \text{if } f(y_i(k + 1)) \leq f(y_i(k)) \end{cases} \quad (16)$$

$$\omega = \left(1.1 - \frac{G_{b_i}}{P_{b_i}}\right) \quad (17)$$

If the SMO algorithm succeeds, the position and velocity of particle  $i$  is calculated with the local leader, global leader, and its own conscience according to Eqs. (18) and (19). All of the particles' position and velocity ranges are updated and the algorithm will terminate with the optimum fitness



value, and the final result will be the output.

$$X_i^{k+1} = X_i^k + U(0, 1) \times (X_i^k - GL_m^k) + U(0, 1) \times (X_r^k - LL_m^k) \quad (18)$$

$$V_i^{k+1} = X_i^{k+1} - X_i^k \quad (19)$$

The proposed hybrid SMPSP algorithm pseudo-code is given below.

---

**Algorithm Hybrid SMPSP Algorithm**

---

*Initialize the parameters of PSO*

*Initialize the population of particles  $X_i$ , velocities  $V_i$  of population vector, minimum velocity ( $V_{min}$ ) and maximum velocity ( $V_{max}$ ) ranges*

*Calculate the fitness function of each particle  $i$ , local best ( $P_{b_i}$ ) and global best ( $G_{b_i}$ )*

$t=0$ ;

**while** *Stopping Criteria* **do**

$t=t+1$ ;

**if**  $f(y_i) > P_{b_i}$

$X_i^k = y(i,k)$

*update the local and global leader with best fitness*

*update  $X_i^{k+1}$  and  $V_i^{k+1}$  for the current iteration using Eqn. (18) and (19).*

**else**

*update inertia weight  $\omega$  using  $G_{b_i}$  and  $P_{b_i}$  using Eqn. (17)*

*update  $X_i^{k+1}$  and  $V_i^{k+1}$  for the current iteration using Eqn. (8.a) and (8.b).*

**end**

*update  $G_{b_i}$*

**end while**

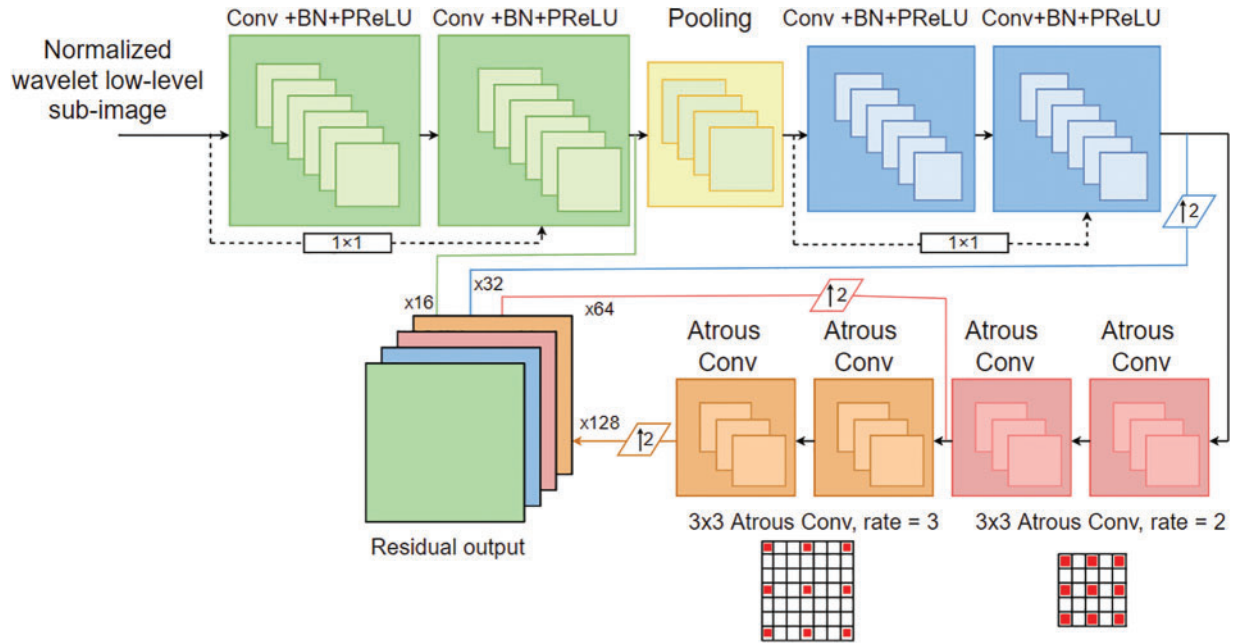
---

## 5 Image Dehazing Model Using Atrous Convolution-Based Residual Deep CNN

### 5.1 Atrous Convolution-Residual Deep CNN

In general, activation functions employed in convolution operations are typically utilized to extract the input data information by injecting a non-linearity into the network. The deep CNN uses 2D convolutional kernels which spans along the input image's rows and columns and extract the features layer by layer. More feature maps will be extracted with the increase in layers, but at the expense of network complexity and training time. The deep learning model of atrous convolution-based residual deep CNN is shown in Fig. 4.

The typical network architecture consists of four convolution layers designed with vanilla convolution, one max-pooling layer, four atrous convolution layers, concatenation of feature maps, residual connections, and skip connection. The Parametric REctified Linear Unit (PRELU) [40] activation function is used to combat the over-fitting and allow the network to learn more complex functions. The gradient descent algorithm used for this research is Stochastic Gradient Descent (SGD). The performance of the dehazing process is improved by maintaining the optimum batch size, epochs and the learning rate for more iterations using the hybrid SMPSP algorithm.



**Figure 4:** Deep learning model of atrous convolution-based residual deep CNN

The feature maps will succeed from one layer to the next layer and convolve with the filter banks in the corresponding convolution layer. The response of each convolution layer is calculated using Eq. (20).

$$V_j^{l+1} = f \left( \sum_i V_i^l * k_{ij}^{l+1} + b_{ij}^{l+1} \right) \quad (20)$$

here,  $V_j^{l+1}$  and  $V_i^l$  are the respective feature maps of the  $(l+1)^{\text{th}}$  layer and  $l^{\text{th}}$  layer,  $k$  is the 2D convolution kernel,  $*$  denotes the convolution operator,  $i, j$  represents the current layer  $i^{\text{th}}$  feature map to the next layer  $j^{\text{th}}$  feature,  $f$  is the activation function and  $b_n^{l+1}$  is the offset. The first shallow convolution layer ( $l = 1$ ) learns the feature maps from the hazy input image.

The issue of overfitting occurs when there are fewer training samples than there are feature maps. To overcome this, PReLU activation function is used because of its superior convergence functionality than the Sigmoid and Tanh functions. The form of the PReLU activation function with learnable parameter  $z_i$  as in Eq. (21). Max-pooling operation [41] further reduces the features resolution and network parameters and maintains texture information of the image patch.

$$f(z_i) = \max(0, z_i) + a_i \min(0, z_i) \quad (21)$$

The process of dehazing requires many feature maps to maintain the resolution. The latest U-Net [42] and 3D-Unet [43] utilize the convolution layers between features of different resolutions. This leads to a loss of feature representation which could not serve the purpose. A linear mapping using the shortcut connection with a convolution layer of size  $1 \times 1$  increases the network training speed and combines features from different depths to communicate between the other resolutions. The features from different depths are concatenated to the final residual output.

The shallow convolutional layers of the network use a small receptive field to learn the short-range contextual information (local information). The deep atrous convolution layers use the larger receptive

field to understand the long-range contextual information (global information). This enhances the extraction of contextual information. The global features of deep layers from different depths are propagated to the final output, resulting in a residual image. The atrous convolution kernel position can be expressed as in Eq. (22).

$$y(i) = \sum_j x(i + r \cdot j) w[j] \quad (22)$$

here,  $x$  and  $y$  is are the respective layer input and output with a kernel  $w$ ,  $i$  is the kernel position, and  $r$  is the dilation rate.

Every CNN model typically anticipates the normalized image before processing to reduce the dimensions and computational complexity. The normalized input images are split into patches, pooled in batches of size  $N$ , and sent to the CNN model. In the experiment, we chose the of  $256 \times 256$  size input image patch to the network, and  $N = 50$  as the batch size for training.

The shallow two vanilla convolution layers of the model are made up of 16 filters with stride equal to one with a kernel of  $9 \times 9$ . The max-pooling layer has a receptive field of  $2 \times 2$ . The second set of vanilla convolution layers of the model consists of 32 filters with a stride equal to one with kernel size  $7 \times 7$ . The deep atrous convolution layers consist of 64 filters and 128 filters with kernel sizes of  $3 \times 3$  with a rate of 2 and 3 respectively. The network learns about 375,785 parameters in total from training.

## 5.2 Optimized Atrous Convolution-Based Residual Deep CNN

To enhance the dehazing performance of the atrous convolution-based residual deep CNN method, the hybrid SMP SO algorithm optimizes the batch size (B), epochs (E), and learning rate (L) with the objective function of optimum MSE and PSNR as in Eq. (23).

$$Obj = \underset{B, E, L}{\operatorname{argmin}} \left( \frac{1}{PSNR} \right) \quad (23)$$

## 5.3 Loss Function

In recent image dehazing research, the different combinations of loss functions evolve endlessly to increase the image details. This increases the risk of tuning the hyper-parameters. After extensive testing, using a mean absolute error (MAE) function as in Eq. (24), as the loss function of our proposed method can have an apparent dehazing effect, as this function is sensitive to sparse features of the input image.

$$Loss_{MAE} = \frac{1}{CWH} \sum_m \sum_n \sum_p \|\hat{Z}_{m,n,p} - Z_{m,n,p}\| \quad (24)$$

here,  $\hat{Z}_{m,n,p}$  represents the recovered haze-free image, and  $Z_{m,n,p}$  represents the corresponding ground truth training label,  $C, W, H$  denotes the channel count, width and height of the training image respectively.

## 6 Results and Discussions

### 6.1 Experimental Setup

The proposed image dehazing framework requires TensorFlow deep learning framework. The optimized batch size is 48 and 44; optimum number of epochs is 75 and 62; the optimum learning rate is 0.84 and 0.76 for corresponding datasets. The network training and evaluation are performed in a server with a personal computer, 16 GB RAM, Intel® Core i7-1280P Processor, and NVIDIA

Graphical Processing Unit to speed up the network operation. The network is subjected to critical training and reconstruction constraints.

## 6.2 Performance Metrics

The performance assessment metrics used for analysis in this research work are Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Mean Absolute Error (MAE), Average Difference (AD), and Maximum Difference (MD) and Structural Similarity Index (SSIM).

- (a) Mean Square Error (MSE): It is a dispersion metric for measuring the efficiency of image enhancement algorithms in the applications of real-world image processing. It is measured as the difference among the recovered clear image and the corresponding ground truth image as in Eq. (25).
- (b) Peak Signal to Noise Ratio (PSNR): It is a higher order and correlative metric depending on MSE used to measure the quality of the recovered clear image as in Eq. (26).
- (c) Mean Absolute Error (MAE): It is a conventional metric to measure the amount of haze effect present in an input image as in Eq. (27).
- (d) Average Difference (AD): It is widely used in image processing applications including object detection and recognition. It is the average concern pixel change in the recovered image and its respective ground truth image as in Eq. (28).
- (e) Maximum Difference (MD): It is the maximum of the absolute difference among the recovered image and its respective ground truth image as in Eq. (29).
- (f) Structural Similarity Index Metric (SSIM): It is a higher order metric used to assess the quality of the dehazed image concerning ground truth image as in Eq. (30), typically its value lies from 0 to 1.

The expressions for the above metrics are provided below

$$MSE = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (D_{(m,n)} - G_{(m,n)})^2 \quad (25)$$

$$PSNR = 10 \log_{10} \frac{I_{max}^2}{MSE} \quad (26)$$

$$MD = |D_{(m,n)} - G_{(m,n)}| \quad (27)$$

$$MAE = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N |D_{(m,n)} - G_{(m,n)}| \quad (28)$$

$$AD = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (D_{(m,n)} - G_{(m,n)}) \quad (29)$$

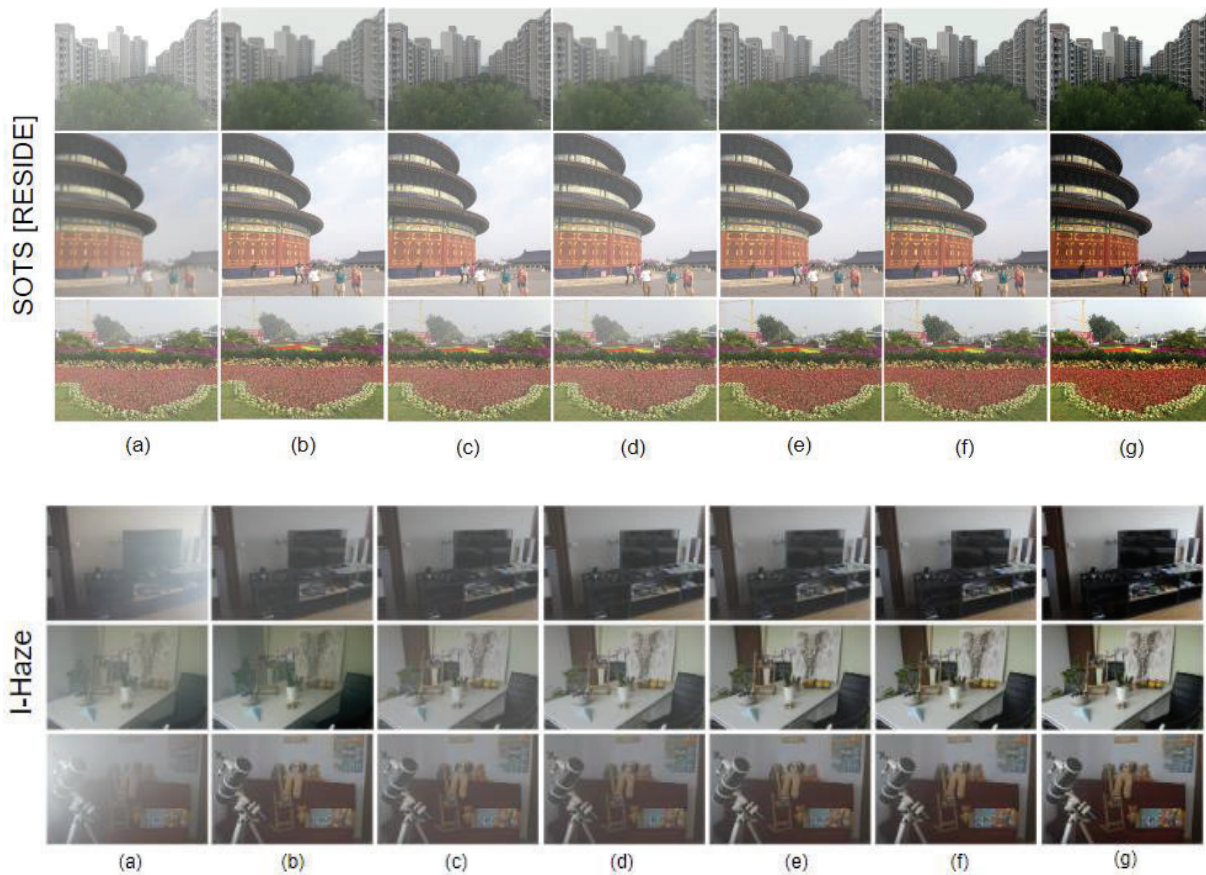
here,  $D$  and  $G$  are the dehazed and corresponding hazy input image at location  $m, n$  coordinates with a pixel count of  $M \times N$  along the horizontal and vertical directions respectively,  $I_{max}$  is the maximum possible intensity, typically it is set to 255

$$SSIM = \frac{(2\mu_m\mu_n + C_1)(\sigma_{mn} + C_2)}{(\mu_m^2 + \mu_n^2 + C_1)(\sigma_m^2 + \sigma_n^2 + C_2)} \quad (30)$$

here,  $\mu_m, \mu_n$  is the average of  $m, n$ ,  $\sigma_m, \sigma_n$  is the standard deviation of  $m, n$ ,  $\sigma_{mn}$  is the covariance of  $m, n$  and  $C_1, C_2$  are set constants to avoid any possibility of the denominator being zero.

### 6.3 Statistical Analysis of Atrous-Deep CNN with Optimization Algorithms

The recovered clear images obtained using CNN [21], MSCNN [24], Ranking CNN [44], Deep CA learning [45], Autoencoder [46], and the proposed method with an optimization algorithm, are depicted in Fig. 5 for respective datasets.



**Figure 5:** Dehazed images using (a) Haze input (b) CNN [21] (c) MSCNN [24] (d) Ranking CNN [44] (e) DeepCA learning [45] (f) Autoencoder [46] (g) Proposed deep learning network

The statistical analysis of MSE, PSNR, SSIM, AD, MD, and MAE using proposed atrous convolution-based deep CNN with Deer Hunting Optimization (DHO) [47], Whale Optimization (WO) [48], SMO [39], PSO [38] and newly designed hybrid SMPSO algorithm on the respective datasets is shown in Table 1.

The analysis of MSE and PSNR for existing methods and the proposed method with hybrid SMPSO algorithm on the respective datasets are shown in Table 2 and the analysis of SSIM, MAE, AD and MD is shown in Fig. 6. The optimized solutions of meta-heuristic algorithms on the proposed method are shown in Table 3.

The statistical analysis in Tables 1 and 2 shows that the proposed atrous convolution-based hybrid SMPSO algorithm performs better than other optimization algorithms and existing methods respectively. Table 3 shows that the optimum values of no. of epochs, batch size and the learning rate are obtained with a proposed method with a hybrid optimization algorithm.

The proposed atrous convolution extracts the full range of contextual information from the low-frequency wavelet decomposed images. These reduces the number of convolution layers during network training and enhances the efficiency of the network in dehazing process. The results obtained shows the qualitative and quantitative performance of the proposed method with the optimum batch size, epochs and learning rate using hybrid optimization.

**Table 1:** Statistical analysis of MSE, PSNR, SSIM, AD, MD and MAE using the proposed method with different optimization algorithms

Dataset	Methods/Terms	Avg. MSE	Avg. PSNR	Avg. SSIM	Avg. AD	Avg. MD	Avg. MAE
SOTS [RESIDE]	Atrous-Deep CNN-DHO	77.066	19.94	0.76	0.814	0.57	0.809
	Atrous-Deep CNN-WO	75.093	20.014	0.764	0.786	0.562	0.774
	Atrous-Deep CNN-SMO	69.032	19.987	0.802	0.731	0.437	0.723
	Atrous-Deep CNN-PSO	62.081	19.602	0.878	0.692	0.429	0.701
	<b>Atrous-Deep CNN-SMPSO</b>	<b>56.265</b>	<b>24.968</b>	<b>0.891</b>	<b>0.62</b>	<b>0.311</b>	<b>0.63</b>
	I-Haze	Atrous-Deep CNN-DHO	82.817	24.136	0.769	0.69	0.583
Atrous-Deep CNN-WO		79.932	23.359	0.776	0.65	0.549	0.63
Atrous-Deep CNN-SMO		75.834	26.947	0.791	0.53	0.516	0.54
Atrous-Deep CNN-PSO		73.516	27.562	0.826	0.48	0.43	0.49
<b>Atrous-Deep CNN-SMPSO</b>		<b>67.79</b>	<b>31.745</b>	<b>0.912</b>	<b>0.41</b>	<b>0.326</b>	<b>0.43</b>

**Table 2:** Statistical analysis of MSE and PSNR for respective datasets

Methods/Terms	Dataset	CNN [21]	MSCNN [24]	Ranking CNN [44]	DeepCA learning [45]	Autoencoder [46]	Atrous-Deep CNN-SMPSO
Avg. MSE	SOTS [RESIDE]	65.754	81.637	71.256	77.513	73.825	<b>56.265</b>
	I-Haze	75.263	89.726	79.168	85.637	82.492	<b>67.79</b>
Avg. PSNR	SOTS [RESIDE]	22.672	19.512	21.796	19.168	20.714	<b>24.968</b>
	I-Haze	26.814	21.538	25.162	22.723	24.352	<b>31.745</b>

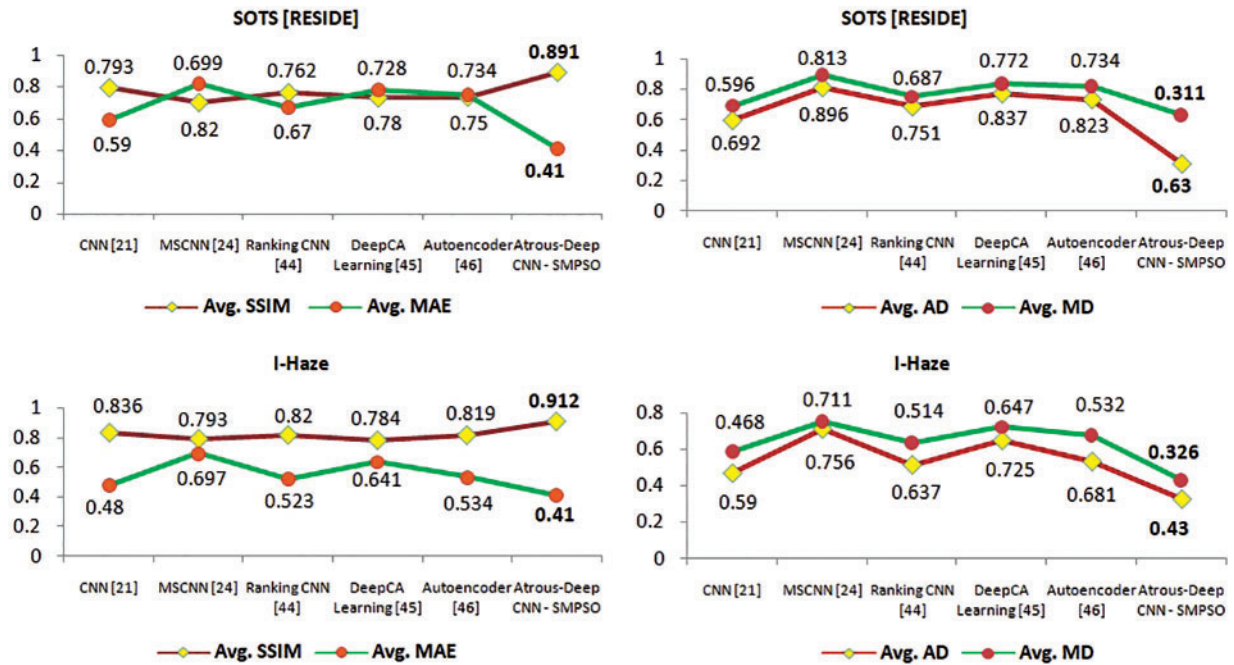


Figure 6: Statistical analysis of SSIM, MAE, AD and MD on existing methods and proposed method

Table 3: Optimized solutions of meta-heuristic algorithms for the proposed method

Optimized solutions	Epochs	Batch size	Learning rate
SOTS [RESIDE]			
Atrous-Deep CNN-SMO	96	45	0.3468
Atrous-Deep CNN-PSO	93	40	0.673204
Atrous-Deep CNN-SMPSO	75	48	0.84074
I-Haze			
Atrous-Deep CNN-SMO	81	52	0.710147
Atrous-Deep CNN-PSO	74	50	0.923873
Atrous-Deep CNN-SMPSO	62	44	0.764504

## 7 Conclusion

In this article, we propose an optimized atrous convolution-based Deep CNN using a hybrid SMPSO algorithm for image dehazing without relying on atmospheric scattering models. This method generates a clean image free from halo artifacts and poor contrast. The wavelets provide extensive features useful for deep learning models and have high-performance achieving capabilities. The proposed Deep CNN model uses the wavelet multi-resolution features for training, to estimate the extent of the haze effect and predict the clear image from the haze patches. The hybrid optimization algorithm is designed to fine-tune the batch size, epochs, and learning rate required during the data model training. The designed atrous-deep CNN network inherits the local and higher-order

features using atrous convolution and residual connection. The experimental analysis has proven the outstanding performance of the proposed method and quantitative metrics like MSE, PSNR, SSIM, MAE, AD, and MD have been considerably improved with existing optimization algorithms and deep learning methods. The method further can be extended to real-world aerial hazy datasets in change detection, image segmentation tasks, hazy video surveillance tasks and the network model can be strengthened by optimizing more parameters for accurate estimation of the image scenes.

**Acknowledgement:** We would like to thank the Management for providing the NVIDIA GPU support to train the proposed neural network for analysis.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or *et al.*, “Deep photo: Model-based photograph enhancement and viewing,” *ACM Transactions on Graphics*, vol. 27, no. 5, pp. 1–10, 2008.
- [2] T. Treibitz and Y. Y. Schechner, “Polarization: Beneficial for visibility enhancement?” in *2009 IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 525–532, 2009.
- [3] L. P. Yao and Z. I. Pan, “The retinex-based image dehazing using a particle swarm optimization method,” *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3425–3442, 2021.
- [4] C. M. S. Kumar, R. S. Valarmathi and S. Aswath, “An empirical review on image dehazing techniques for change detection of land cover,” in *2021 Asian Conf. on Innovation in Technology (ASIANCON)*, Pune, India, pp. 1–9, 2021.
- [5] K. He, J. Sun and X. Tang, “Single image haze removal using dark channel prior,” in *2009 IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 1956–1963, 2009.
- [6] D. Berman, T. Treibitz and S. Avidan, “Air-light estimation using haze-lines,” in *2017 IEEE Int. Conf. on Computational Photography (ICCP)*, Stanford, CA, USA, pp. 1–9, 2017.
- [7] Q. Zhu, J. Mai and L. Shao, “A fast single image haze removal algorithm using color attenuation prior,” *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3522–3533, 2015.
- [8] L. Peng and B. Li, “Single image dehazing based on improved dark channel prior and unsharp masking algorithm,” in *Intelligent Computing Theories and Application, ICIC 2018*, Wuhan, China, vol. 10954, pp. 347–358, 2018.
- [9] D. Singh, V. Kumar and M. Kaur, “Single image dehazing using gradient channel prior,” *Applied Intelligence*, vol. 49, no. 12, pp. 4276–4293, 2019.
- [10] B. Cai, X. Xu, K. Jia, C. Qing and D. Tao, “DehazeNet: An end-to-end system for single image haze removal,” *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.
- [11] W. -Y. Hsu and Y. -S. Chen, “Single image dehazing using wavelet-based haze-lines and denoising,” *IEEE Access*, vol. 9, pp. 104547–104559, 2021. <https://doi.org/10.1109/ACCESS.2021.3099224>
- [12] S. G. Narasimhan and S. K. Nayar, “Contrast restoration of weather degraded images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 713–724, 2003.
- [13] D. Singh and V. Kumar, “A novel dehazing model for remote sensing images,” *Computers & Electrical Engineering*, vol. 69, pp. 14–27, 2018. <https://doi.org/10.1016/j.compeleceng.2018.05.015>
- [14] C. Li, J. Guo, F. Porikli, H. Fu and Y. Pang, “A cascaded convolutional neural network for single image dehazing,” *IEEE Access*, vol. 6, pp. 24877–24887, 2018. <https://doi.org/10.1109/ACCESS.2018.2818882>
- [15] Y. Liu, J. Shang, L. Pan, A. Wang and M. Wang, “A unified variational model for single image dehazing,” *IEEE Access*, vol. 7, pp. 15722–15736, 2019. <https://doi.org/10.1109/ACCESS.2019.2894525>



- [16] H. Zhang and V. M. Patel, "Densely connected pyramid dehazing network," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 3194–3203, 2018.
- [17] Y. Y. Schechner, S. G. Narasimhan and S. K. Nayar, "Polarization-based vision through haze," *Applied Optics*, vol. 42, no. 3, pp. 511–525, 2003.
- [18] Z. Mi, H. Zhou, Y. Zheng and M. Wang, "Single image dehazing via multiscale gradient domain contrast enhancement," *IET Image Processing*, vol. 10, no. 3, pp. 206–214, 2016.
- [19] Y. Wan and Q. Chen, "Joint image dehazing and contrast enhancement using the HSV color space," in *2015 Visual Communications and Image Processing (VCIP)*, Singapore, pp. 1–4, 2015.
- [20] C. O. Ancuti and C. Ancuti, "Single image dehazing by multi-scale fusion," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3271–3282, 2013.
- [21] J. Li, G. Li and H. Fan, "Image dehazing using residual-based deep CNN," *IEEE Access*, vol. 6, pp. 26831–26842, 2018. <https://doi.org/10.1109/ACCESS.2018.2833888>
- [22] Z. Li, B. Gui, T. Zhen and Y. Zhu, "Grain depot image dehazing via quadtree decomposition and convolutional neural networks," *Alexandria Engineering Journal*, vol. 59, no. 3, pp. 1463–1472, 2020.
- [23] Z. Liu, B. Xiao, M. Alrabeiah, K. Wang and J. Chen, "Single image dehazing with a generic model-agnostic convolutional neural network," *IEEE Signal Processing Letters*, vol. 26, no. 6, pp. 833–837, 2019.
- [24] W. Ren, J. Pan, H. Zhang, X. Cao and M. yang, "Single image dehazing via multi-scale convolutional neural networks with holistic edges," *International Journal of Computer Vision*, vol. 128, pp. 240–259, 2020. <https://doi.org/10.1007/s11263-019-01235-8>
- [25] K. Gan, J. Zhao and H. Chen, "Multilevel image dehazing algorithm using conditional generative adversarial networks," *IEEE Access*, vol. 8, pp. 55221–55229, 2020. <https://doi.org/10.1109/ACCESS.2020.2981944>
- [26] S. Zhang and F. He, "DRCDN: Learning deep residual convolutional dehazing networks," *The Visual Computer*, vol. 36, no. 9, pp. 1797–1808, 2020.
- [27] S. Wang, L. Zhang and X. Wang, "Single image haze removal via attention-based transmission estimation and classification fusion network," *Neurocomputing*, vol. 447, pp. 48–63, 2021.
- [28] F. A. Dharejo, Y. Zhou, F. Deeba, M. A. Jatoi, M. A. Khan *et al.*, "A deep hybrid neural network for single image dehazing via wavelet transform," *Optik*, vol. 231, pp. 166462, 2021. <https://doi.org/10.1016/j.ijleo.2021.166462>
- [29] D. Yang and J. Sun, "A Model-driven deep dehazing approach by learning deep priors," *IEEE Access*, vol. 9, pp. 108542–108556, 2021. <https://doi.org/10.1109/ACCESS.2021.3101319>
- [30] C. Wang, Z. Li, J. Wu, H. Fan, G. Xiao *et al.*, "Deep residual haze network for image dehazing and deraining," *IEEE Access*, vol. 8, pp. 9488–9500, 2020. <https://doi.org/10.1109/ACCESS.2020.2964271>
- [31] X. Min, G. Zhai, K. Gu, Y. Zhu, J. Zhou *et al.*, "Quality evaluation of image dehazing methods using synthetic hazy images," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2319–2333, 2019. <https://doi.org/10.1109/TMM.2019.2902097>
- [32] T. Yu, M. Zhu and H. Chen, "Single image dehazing based on multi-scale segmentation and deep learning," *Machine Vision and Applications*, vol. 33, no. 2, pp. 1–11, 2022.
- [33] A. Bilal, G. Sun and S. Mazhar, "Diabetic retinopathy detection using weighted filters and classification using CNN," in *Proc. of 2021 Int. Conf. on Intelligent Technologies (CONIT)*, Hubli, India, pp. 25–27, 2021.
- [34] A. Bilal, G. Sun, Y. Li, S. Mazhar and J. Latif, "Lung nodules detection using grey wolf optimization by weighted filters and classification using CNN," *Journal of the Chinese Institute of Engineers*, vol. 45, no. 2, pp. 175–186, 2022.
- [35] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.
- [36] C. Blum, M. J. B. Aguilera, A. Roli and M. Sampels, "Hybrid metaheuristics," in *An Emerging Approach to Optimization*, Heidelberg: Springer Berlin, 2008. <https://doi.org/10.1007/978-3-540-78295-7>
- [37] R. Poli, J. Kennedy and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, pp. 33–57, 2007. <https://doi.org/10.1007/s11721-007-0002-0>

- [38] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE Int. Conf. on Evolutionary Computation Proc. IEEE World Congress on Computational Intelligence*, Anchorage, AK, USA, pp. 69–73, 1998.
- [39] J. C. Bansal, H. Sharma, S. S. Jadon and M. Clerc, "Spider monkey optimization algorithm for numerical optimization," *Memetic Computing*, vol. 6, pp. 31–47, 2014. <https://doi.org/10.1007/s12293-013-0128-0>
- [40] R. S. Thakur, R. N. Yadav and L. Gupta, "PReLU and edge-aware filter-based image denoiser using convolutional neural network," *IET Image Process*, vol. 14, pp. 3869–3879, 2020. <https://doi.org/10.1049/iet-ipr.2020.0717>
- [41] Z. Zuo, B. Shuai, G. Wang, X. Liu, X. Wang *et al.*, "Learning contextual dependence with convolutional hierarchical recurrent neural networks," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 2983–2996, 2016.
- [42] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015, Lecture Notes in Computer Science*, vol. 9351, 2015. [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
- [43] K. Kamnitsas, C. Ledig, V. F. J. Newcombe, J. P. Simpson, A. D. Kane *et al.*, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Medical Image Analysis*, vol. 36, pp. 61–78, 2017. <https://doi.org/10.1016/j.media.2016.10.004>
- [44] Y. Song, J. Li, X. Wang and X. Chen, "Single image dehazing using ranking convolutional neural network," *IEEE Transactions on Multimedia*, vol. 20, no. 6, pp. 1548–1560, 2018.
- [45] S. Tangsakul and S. Wongthanavas, "Single image haze removal using deep cellular automata learning," *IEEE Access*, vol. 8, pp. 103181–103199, 2020. <https://doi.org/10.1109/ACCESS.2020.2999076>
- [46] G. Kim, S. W. Park and J. Kwon, "Pixel-wise wasserstein autoencoder for highly generative dehazing," *IEEE Transactions on Image Processing*, vol. 30, pp. 5452–5462, 2021. <https://doi.org/10.1109/TIP.2021.3084743>
- [47] P. Durgaprasadarao and N. Siddaiah, "Design of deer hunting optimization algorithm for accurate 3D indoor node localization," *Evolutionary Intelligence*, 2021. <https://doi.org/10.1007/s12065-021-00673-z>
- [48] J. Rahebi, "Vector quantization using whale optimization algorithm for digital image compression," *Multimedia Tools and Applications*, vol. 81, pp. 20077–20103, 2022. <https://doi.org/10.1007/s11042-022-11952-x>