Signal, Image and Video Processing (2024) 18:2167–2182
https://doi.org/10.1007/s11760-023-02877-5

ORIGINAL PAPER

# DC-GAN with feature attention for single image dehazing

Tewodros Tassew[1] · Nie Xuan[1]

Received: 13 April 2023 / Revised: 25 September 2023 / Accepted: 31 October 2023 / Published online: 22 December 2023

**Abstract**
In recent years, the frequent occurrence of smog weather has affected people's health and has also had a major impact on computer vision application systems. Images captured in hazy environments suffer from quality degradation and other issues such as color distortion, low contrast, and lack of detail. This study proposes an end-to-end, adversarial neural network-based dehazing technique called DC-GAN that combines Dense and Residual blocks efficiently for improved dehazing performance. In addition, it also consists of channel attention and pixel attention, which can offer more versatility when dealing with different forms of data. The Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) was used as an enhancement method to correct the short-comings in the original GAN's cost function and create an improvised loss. Based on the experiment results, the algorithm used in this research can generate sharp images with high image quality. The processed images were simultaneously analyzed using the objective evaluation metrics Peak Signal-to-Noise Ratio and Structural Similarity. The findings from our experiment demonstrate that the dehazing effect is favorable compared to other state-of-the-art dehazing algorithms.

**Keywords** Image dehazing · Deep learning · Generative adversarial networks

## 1 Introduction

Digital images are essential for many industries and research fields, as they serve as the main medium of information transmission. High-quality digital images provide a reliable input that effectively guarantees the performance of computer vision systems in image processing. As such, these images drive the use of various image processing technologies across multiple applications, including autonomous driving, intelligent robots, medical devices, and more. By providing high-quality data input to computer vision systems used in these areas, digital images play an invaluable role in advancing technological progress within them [1].

Haze is a natural phenomenon that has become increasingly common due to human-caused water pollution and increased industrial and agricultural waste. The presence of haze significantly impairs visibility, reducing light capture when taking photos with computer-based imaging devices. This results in reduced contrast, color distortion, loss of detailed content, as well as an overall grayish coloration to images. Additionally, it greatly impedes people's ability to travel safely due to decreased visibility caused by the hazy environment [2].

Atmospheric scattering models (ASM) are mathematical tools used to simulate the interaction between light and particles in a given atmosphere. The term "a given atmosphere" refers to the specific conditions and characteristics of an atmospheric layer or region, such as its temperature, pressure, humidity, and optical depth. These models can be applied to different types of atmospheres, such as Earth's or Mars', depending on the physical properties and composition of the particles. Different atmospheres can have different effects on light propagation and visibility.

These models can be used to accurately predict how sunlight is scattered, absorbed, and reflected by atmospheric components such as aerosols, clouds, and gases. The hazing effect occurs when sunlight is being scattered by these particles, resulting in a softening or blurring of distant objects due to reduced contrast between them and their background.

Tewodros Tassew and Nie Xuan contributed equally to this work.

✉ Nie Xuan
xnie@nwpu.edu.cn

Tewodros Tassew
ted.meg1234@mail.nwpu.edu.cn

[1] School of Software Engineering, Northwestern Polytechnical University, 127 West Youyi Road, Beilin District, Xi'an 710072, Shaanxi, China

The equation used to represent ASM is:

$$I(x) = J(x)t(x) + A(1 - t(x)) \tag{1}$$

where $I(x)$ represent the observed hazy image, $J(x)$ is the original image, $A$ is the global atmospheric lighting, and $t(x)$ is the transmission. A refers to the natural light of the atmosphere across the entire scene while $t(x)$ represents the amount of light that reaches the camera from the object and is calculated as follows:

$$t(x) = e^{-\beta d(X)} \tag{2}$$

The most common type of atmospheric scattering model is called Mie Theory which was developed by Gustav Mie in 1908. Koschmieder's atmospheric scattering model is a more suitable approach for describing the optical properties of haze than Mie theory. Mie theory assumes that the scattering particles are spherical and homogeneous, which is not the case for haze particles that have irregular shapes and compositions. Koschmieder's model, on the other hand, considers the effects of multiple scattering and absorption by the particles, as well as the variation of the particle size distribution and refractive index with altitude. Therefore, Koschmieder's model can better account for the attenuation and coloration of light by haze in the atmosphere.

The hazing effect caused by aerosol particles has many practical implications ranging from aviation safety (reducing visibility) to climate change (affecting solar radiation balance). It also affects our perception of the environment since it reduces contrast making it harder for us humans to perceive details at long distances like mountains and buildings. This information is invaluable for applications ranging from climate change research to predicting air quality levels or visibility conditions in an area. Atmospheric scattering models provide a powerful tool for understanding the effects of our changing environment on both local weather patterns as well as global climate systems (Fig. 1).

The future of image processing and analysis will be heavily reliant on low-quality images, which have the potential to affect other related projects and progress in the field. For instance, when using advanced technologies such as unmanned aerial photography and autonomous driving systems there are stringent requirements for image quality that must be met in order to ensure high accuracy. As a result, it is essential that researchers continue to develop methods for effectively utilizing low-resolution images while still achieving accurate results [3].

Image defogging is crucial for computer vision systems, as it enables the successful application of image control, object identification, and tracking. Haze and fog can severely impede the accuracy of video recordings taken in hazy conditions. Image defogging can use methods such as noise reduction algorithms or advanced color restoration techniques to enhance the clarity and details of the image, which are important for computer vision applications. Therefore, it is essential to study how to obtain better results from photos taken in hazy environments, as this can improve the performance of these technologies.

Currently, image defogging algorithms can be classified into three main categories based on their theoretical foundations: image enhancement-based, physical model-based, and deep learning-based methods. Image enhancement-based methods use image processing techniques to increase the contrast and sharpen the details of the image. Physical model-based methods use a physical model of the fog or haze formation and try to estimate and remove its effect from the image. Deep learning-based methods use neural networks to learn the mapping between foggy and defogged images from a large dataset of images. The image enhancement method has an advantage over the other two methods due to its long development time and technological advancement. When compared to the other two methods, the main disadvantages of this method are the loss of details and color distortion in the generated defogging image.

The second method employs techniques based on physical models [4]. The basic idea is to develop a physical model of atmospheric scattering, identify the key variables, and then derive an image free of fog. The physical model-based defogging algorithm has an advantage over the other two methods in that it retains more image structure information and has more image details after de-fogging, resulting in a more natural image without fog. However, due to the difficulty in establishing physical models and estimating intermediate variables, the complexity and cost of the physical model-based defogging algorithm technology are high, so the physical model-based defogging method is not widely recognized in the computer vision field at the moment [5].

Fog removal algorithms based on deep learning can be classified into two types. The first type uses the atmospheric scattering physical model to estimate some intermediate variables and then, reconstructs the fog-free image. The second type uses an end-to-end learning method to directly learn the mapping function between the foggy and fog-free images. Image enhancement-based fog removal algorithms can be divided into two categories: global enhancement and local enhancement. These algorithms use image processing techniques to improve the image quality by enhancing the contrast and the details [6].

## 2 Related work

A common challenge in image processing is to reduce the impact of haze on the quality and clarity of images. Several approaches have been proposed in the past to enhance or

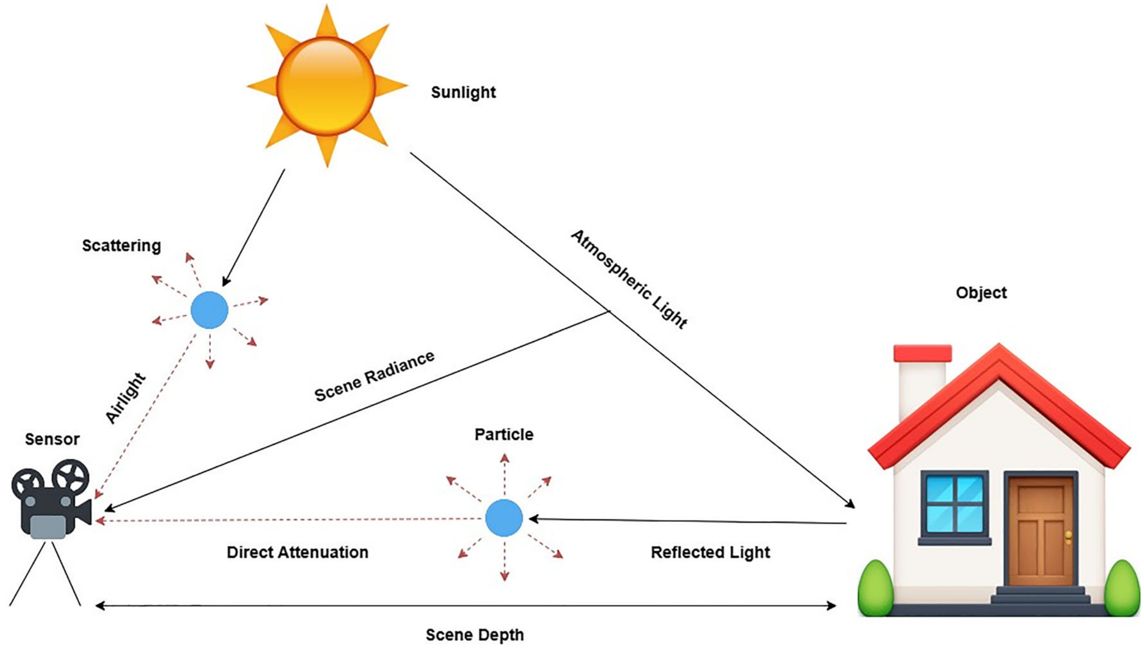

**Fig. 1** Atmospheric scattering model

remove haze from images, such as image enhancement algorithms or model-based dehazing algorithms. However, these methods may have limitations in terms of computational efficiency, robustness, or generalization. Recently, deep learning methods based on convolutional neural networks (CNNs) and generative adversarial networks (GANs) have shown promising results for various high-level vision tasks, such as image classification or deblurring. Therefore, these methods have also been applied to the problem of image dehazing, achieving state-of-the-art performance, and overcoming some of the drawbacks of previous methods.

Deep learning-based approaches are superior to classical methods because they use deeper features rather than superficial ones when processing an input image or video frame sequence containing hazy scenes. Deep learning-based methods for dehazing can be divided into two categories: single image dehazing and video dehazing. Single image dehazing aims to restore a clear image from a single hazy input, while video dehazing aims to restore a clear video from a hazy video sequence. Both problems are ill-posed, as there is no unique solution for dehazing, and the haze parameters are unknown. However, deep learning-based methods can overcome these challenges by learning from large-scale datasets of hazy and clear images or videos, or by generating synthetic data using physical models of haze formation.

For example, some GAN architectures have been developed which can generate clear images from hazy inputs using end-to-end training frameworks. GANs can produce realistic and natural-looking results, as they can capture the high-level semantic information and the low-level details of the scenes. CNNs can be trained with paired datasets, where each hazy image or video is matched with a corresponding clear one, or with unpaired datasets, where there is no direct correspondence between the hazy and clear samples. CNNs can learn to estimate the haze parameters, such as the transmission map and the atmospheric light, or to directly output the clear images or videos without explicitly modeling the haze [7].

## 2.1 CNN-based dehazing

Convolutional Neural Networks (CNNs) are increasingly being used in image dehazing, typically trained on synthetic data. This is done by finding the dehazed image directly from the hazy image or by extracting a transmission map. By using CNNs for this task, they can learn to identify and remove haze from images with high accuracy as compared to traditional methods like dark channel prior or multi-scale fusion-based approaches [8].

The Generic Model-Agnostic Convolutional Neural Network (GMAN) [9] has revolutionized the image restoration process by providing a solution to the problem of lossy reconstruction of original images from hazy ones. Unlike previous methods, GMAN does not rely on parameters *A* and *t(x)* or their variants for this purpose, as it is unlikely that these can be transformed equivalently into an estimation problem when subject to the same evaluation metric. Parameters A and t(x) are commonly used in haze removal models based on the atmospheric scattering model. Other image restoration models may use different parameters depending on the degradation they aim to address. For example, some models for image deblurring use a blur kernel K to model the motion blur effect. Common parameters in image restoration models include noise level, regularization parameter, and learning rate, which are not specific to the degradation model but rather to the optimization or learning algorithm used to solve the inverse problem. Relations between models with parameters are not trivial and require assumptions or approximations.

In addition, GMAN overcomes issues with ASM which fails to capture complex relationships between the original and hazy images and produces unsatisfactory results on naturally hazed images despite its effectiveness in restoring synthetically hazed ones. As such, GMAN provides a much-needed breakthrough in terms of restoring clear images from hazy sources without relying heavily on traditional parameters or models.

In [10] demonstrated a highly successful CNN-based dehazing method that utilizes semantic features extracted from a single image to derive color priors. This model was tested on both synthetic and real-world hazy images, resulting in superior performance for recovering clear images even under difficult conditions with high ambiguity. Despite its success, the model needs to be further trained on an expanded set of natural outdoor scenes to improve accuracy and robustness going forward.

Two popular neural networks for calculating the transmission map from a hazy input image are DehazeNet [11] and AOD-Net [12]. AOD-Net has the advantage of estimating both the transmission map and atmospheric light simultaneously; however, when applied to videos, this method may cause flickering artifacts, making it unsuitable for video dehazing. To address these issues, an alternative approach was required that employs a deep neural network capable of learning an atmospheric scattering model to directly optimize haze removal via end-to-end mapping without the need for explicit estimation of medium transmission maps.

The AOD-Net model has been further extended to EVD-Net [13], and an additional network used for video dehazing. This new network seeks to exploit the temporal coherence property of adjacent video frames, making it more effective at removing haze from videos than its predecessor. However, this method also suffers from flicker artefacts when applied on certain sequences due to the inherent limitations of using a single frame as input.

GCANet [14] is an effective method for removing grid artifacts caused by dilated convolution. To achieve this goal, it takes advantage of the most recent smoothed dilation approach. FFANet (Feature Fusion Attention Network) developed by [15] provides additional flexibility when dealing with diverse types of information. It uses L1 loss to compare dehazed images with ground truths, which works well in most cases; however, in real-world scenarios, it can produce some flickering artifacts.

CNN-based dehazing methods have achieved remarkable results in recent years, but they still suffer from some limitations. GAN-based dehazing frameworks have emerged as a promising solution for restoring clear images from hazy ones. Unlike CNN-based methods, which have some drawbacks such as needing paired data, being prone to noise and artifacts, and lacking perceptual quality, GAN-based methods can leverage unpaired data, preserve the original details and colors, and enhance the visual appeal of the results. They can produce more realistic and natural dehazed images that are closer to human perception.

### 2.2 GAN-based dehazing

Generative Adversarial Networks (GANs) are a deep learning and unsupervised machine learning technique first introduced by [16]. It is made up of a generator and a discriminator that compete against each other in a zero-sum game. Both blocks, which are based on deep neural network architecture, are designed to simplify the generative and adversarial processes and can be trained using forward and backward propagation. GANs have made significant strides since they were first introduced. Numerous noteworthy advancements have been made to enhance system performance and the learning process, which include Deep Convolutional GAN (DCGAN) [17], Wasserstein GAN (WGAN) [18], Conditional GAN (cGAN) [19], and Cycle-GAN [20]. Many GANs have recently been demonstrated to be quite successful at image dehazing. One such network is the All-in-One Dehazing Network, or AOD-Net (All-in-One Dehazing Network) [12], a lightweight CNN architecture based on a revised atmospheric scattering model. De-Haze and Smoke GAN (DHSGAN) [21] is another dehazing network that does not require any post-processing or atmospheric model inversion. Yang et al. [22] used a cycle-consistency loss to demonstrate a disentangled dehazing network (DDN). It uses only one generator to remove haze; however, this model replaces the other generator with the atmospheric scattering model to generate hazy images. The cycle-consistency loss is then computed using both the input and reconstructed hazy images.

Engin et al. [23] proposed Cycle-Dehaze, a method that enhances the visual quality of dehazed images by combining perceptual loss and cycle-consistency loss. They designed a network that does not rely on the Koschmieder model, unlike DDN. Wei et al. [24] presented a Cycle-GAN-based end-to-end learning system for dehazing. On the other hand, Li et al. [25] developed an encoder–decoder architecture based on a cGAN (conditional generative adversarial network) to learn dehazed scenes. They used unpaired training images that are both haze-free and hazy and removed the haze using an adversarial discriminator that enforces cycle consistency.

FD-GAN [26] employs GANs with a fusion discriminator for single image dehazing. HIDeGAN [27] introduced an image dehazing architecture that takes a blurry RGB image as input and converts it to a hyperspectral image (HSI). Because of its enormous ability to generate realistic images, GAN has been employed in a variety of vision-based applications such as image denoising [28], de-raining [29, 30], and super-resolution [31–33]. Because these approaches do not use physical scattering models, they do not provide dehazed images with adequate contrast and color. In this paper, we propose a unique GAN-based image dehazing network called DC-GAN with feature attention that can dehaze both real and synthetic images.

GAN-based dehazing methods have shown promising results in removing haze from images without relying on prior knowledge or assumptions about the atmospheric conditions. However, most of these methods suffer from some limitations, such as low resolution, color distortion, or artifacts. To address these issues, we propose a novel densely connected GAN with feature attention for image dehazing. Our method leverages the advantages of dense connections and feature attention to enhance the quality and diversity of the generated images. We also introduce a new loss function that combines adversarial, perceptual, and WGAN-GP losses to guide the learning process. We also evaluate our method on several benchmark datasets.

## 3 Methodology

The paper presents a novel GAN-based dehazing method that uses densely connected networks and feature attention to generate realistic and clear images. This method addresses limitations of conventional GANs, such as mode collapse, lack of diversity, and artifacts. It enhances information flow and feature reuse among layers, improving performance. The method adaptively emphasizes important features and suppresses irrelevant ones, preserving details and contrast. However, it may not remove thin or dense haze regions or introduce color distortion or noise. Future work will incorporate advanced techniques like multi-scale feature fusion. In this section, the dataset, preprocessing procedures, the architecture of the proposed DC-GAN network for dehazing along with detail on the loss functions that were used during the training stage are discussed.

### 3.1 Dataset

The New Trends in Image Restoration and Enhancement (NTIRE) 2018-Dehazing challenge and REalistic Single Image DEhazing (RESIDE) [34] datasets are utilized to train the proposed network. Unlike previous methods that use video sequences or multiple images as input, our method can handle single images.

NTIRE is one of the recent datasets to be used as a benchmark for the most advanced image dehazing methods and to encourage additional study in the area. The Indoor dataset [35] of the NTIRE-Dehazing contains 25 training images, 5 validation images, and 5 test images, whereas the Outdoor dataset [36] contains 40 training images, 5 validation images, and 5 test images. Both the indoor and outdoor datasets contain two classes, namely the ground truth (GT) and hazy. Because the ground truth of the test data set has not been released, we present the findings and evaluate the model using the validation set. The images had a high resolution of about 3000 × 3000 pixels each. A specialized haze/fog generator was used to produce the haze effect, which simulates the realistic conditions of foggy scenarios.

The RESIDE dataset is a large-scale dehazing benchmark dataset composed of single images and an empirical and experimental expansion known as RESIDE-$\beta$. The RESIDE dataset was created by researchers from the University of Science and Technology of China, and it consists of more than 30,000 images with different levels of haze. The dataset covers various scenarios, such as indoor, outdoor, natural, urban, and synthetic. The dataset also provides ground truth images without haze for evaluation and comparison. It is divided into five subsets: synthetic large-scale Indoor Training Set (ITS), Synthetic Objective Testing Set (SOTS), Hybrid Subjective Testing Set (HSTS), an Outdoor Training Set (OTS), and a Real-world Task-driven Testing Set (RTTS).

For our experiments, we focused on the SOTS subset, which is one of the most commonly used benchmarks for dehazing methods. This subset contains 1560 synthetic images with haze and their corresponding ground truth images. The images are divided into two categories: indoor (500 images) and outdoor (1060 images). The indoor images are from the same source as ITS, while the outdoor images are from the Middlebury Stereo Dataset and the Make3D Dataset. The images in this subset are used for objective evaluation of dehazing methods.

This dataset was split into training and validation sets in the same manner as the NTIRE 2018 dataset. Since the 10 indoor images of the haze class were synthetically generated from each ground truth image, only the image with the highest noise was selected to address the class imbalance issue. As a result, the training set for the indoor dataset contains 45 images per class, whereas the validation set has 5 images for each class (Fig. 2).

For the outdoor dataset, 470 images were assigned to each class for the training set and 20 images per class were assigned to the validation set. The train-test split is summarized in Table 1.

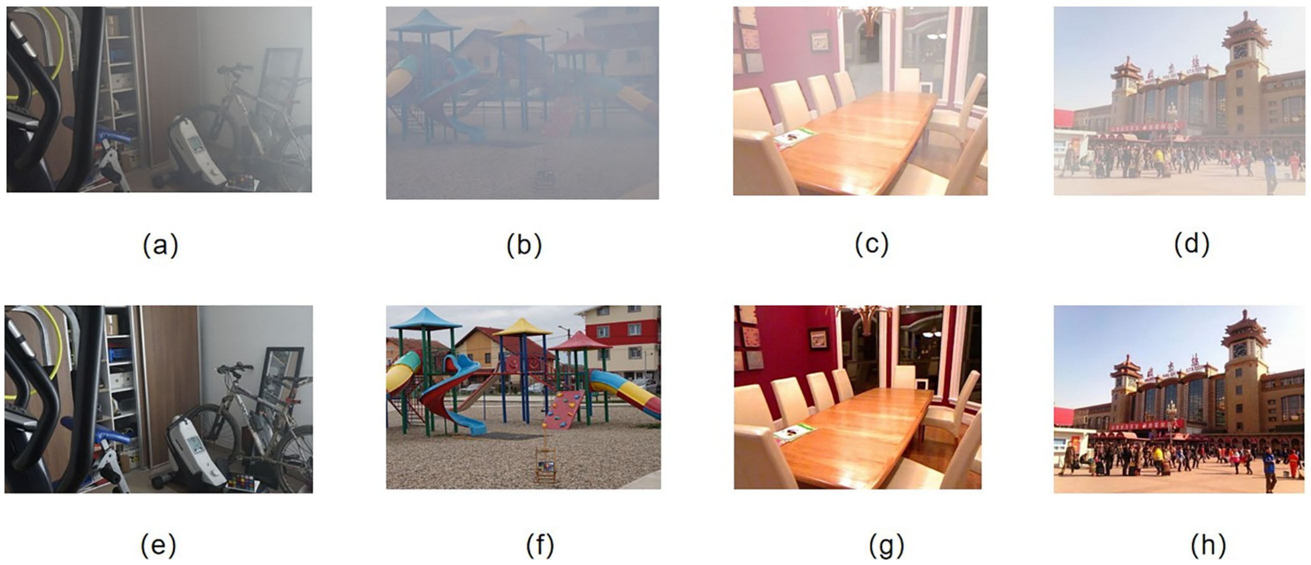

**Fig. 2** **a** and **e** are sample indoor hazy and ground truth images, while **b** and **f** are sample outdoor hazy and ground truth images of the NTIRE 2018 dataset, respectively, and **c** and **g** are sample indoor hazy and ground truth images while **d** and **h** are sample outdoor hazy and ground truth images of the RESIDE SOTS dataset, respectively

**Table 1** The train-test split for NTIRE 2018 and SOTS datasets

| Name of the subset | Types of images | Classes | Training images | Testing images |
|---|---|---|---|---|
| NTIRE 2018 | Indoor | Ground truth | 25 | 5 |
| | | Hazy | 25 | 5 |
| | Outdoor | Ground truth | 40 | 5 |
| | | Hazy | 40 | 5 |
| SOTS | Indoor | Ground truth | 45 | 5 |
| | | Hazy | 45 | 5 |
| | Outdoor | Ground truth | 470 | 20 |
| | | Hazy | 470 | 20 |

### 3.2 Pre-processing

Because of the high resolution of the images in the NTIRE 2018, it is difficult to use the entire image for training. A possible approach is to resize the images to a lower resolution; however, this may result in the loss of important high-level features. A patch-based training technique is utilized to overcome this issue, in which the entire image is split into smaller sized patches. Even if the memory issue is handled using the cropping technique, the network's receptive field is reduced, resulting in the loss of global context information.

To train our model, we applied a multi-scale cropping technique, where we extracted patches of varying sizes that were 1024 × 1024, 1024 × 2048, and 2048 × 2048 pixels. These patches were then resized to 512 × 512 pixels before feeding them to the network. However, for the SOTS dataset, which contained smaller images, we did not use the patch-based approach. Instead, we simply resized all the images to 512 × 512 pixels without cropping. This way, we ensured a consistent input size for our model across different datasets. Resizing images can have an impact on the hazing and picture quality of the dehazed results. On one hand, resizing can reduce the noise and artifacts in the original images, which can help the network to learn better features and produce clearer outputs. On the other hand, resizing can also introduce some distortion and blurring, which can degrade the quality of the dehazed images. Therefore, it is important to balance the trade-off between resizing and preserving the original details of the images. In our experiments, we found that resizing to 512 × 512 pixels did not significantly affect the performance of our model on different datasets, and it also reduced the computational cost and memory usage. However, for some applications that require high-resolution images, resizing may not be desirable. In that case, our patch-based approach can be applied to handle large-scale images without compromising the quality.

The data augmentation technique that was used during the experiment is random rotation. The rotation augmentation randomly rotates the images in the training set clockwise by a given number of degrees from 0 to 270 before being fed to the model. This helps in artificially increasing the dataset since there are not a lot of images, especially in the NTIRE 2018 dataset. Random rotation can improve the model without us having to collect and label more data. A possible justification for using random rotation as a data augmentation technique is that it can make the model more robust to orientation changes in the input images, which may occur in real-world scenarios.

## 3.3 Network architecture

In this section, we present the details of our method, which includes the generator, discriminator, and experiments on different GAN Loss functions. The dehazing model has two main parts—a generator that produces dehazed images from hazy inputs, and a discriminator that evaluates the quality and realism of the generated images. During training, the generator learns to construct a mapping from an input hazy image $I$ to its corresponding ground truth $J$. The generated output $G(I)$ is then given as input to the discriminator along with its ground truth for comparison. Adversarial learning between these two components is used in order to extract information from $I$, by minimizing their respective loss functions $L_d$ and $L_g$ during training, while ignoring one when updating parameters for another component, respectively. This means that during training, if either component misclassifies or fails at creating realistic outputs, respectively (i.e., generating fake images), it will be penalized accordingly through their respective losses which can help improve performance overall. The overall architecture is shown in Fig. 3.

### 3.3.1 Generator

The aim of the generator is to produce sharp images from hazy inputs without predicting intermediate factors. The generator must keep the image content and recover the details while removing the haze in order to accomplish this goal. Many studies have shown that dense connections have the potential to improve feature extraction and utilization, particularly for low-level vision tasks. We propose a novel encoder–decoder architecture for the generator that leverages dense connections and feature attention to enhance the performance of image dehazing. The main components and contributions of the proposed network architecture for image dehazing are presented below.

1. The use of dense blocks instead of convolutional layers to enhance the feature extraction and convergence of the encoder–decoder network by allowing more information flow across feature maps.
2. Our encoder consists of the first convolutional layer and the first three dense blocks with their corresponding transition blocks from a pre-trained DenseNet-121 model.
3. The use of a multi-level pyramid pooling module to capture global context information at different scales and fuse it with the encoder features.
4. The up sampling blocks are connected to the encoder features through skip connections.
5. Our decoder is composed of a series of up sampling blocks, each containing a dense bottleneck layer and a transition layer.
6. The use of a group layer structure based on FFANet to combine residual learning and dense connections in the decoder, which improves the reconstruction quality and preserves fine details.
7. The use of feature attention to adaptively weight the channel and pixel features according to their relevance for dehazing, which enhances the contrast and visibility of the output image. By combining channel and pixel attention, our network can effectively handle various types of haze distributions and enhance the visibility and contrast of the dehazed images.

*Encoder* Dense blocks are used to improve convergence by maximizing information flow along features. A multi-level pyramid pooling module is utilized to refine the learned features. The first Conv layer and the first three Dense Blocks with their related down sampling Transition-Blocks from a pre-trained dense-net 121 are utilized as an encoder structure to exploit the pre-defined weights.

*Skip-Connection* The proposed densely connected encoder–decoder structure integrates different features within the network; however, it lacks the "global” structural information of objects at various scales. A multi-level pyramid pooling block is used to capture additional global context information between different objects to overcome this issue. This is motivated by using global context information in classification and segmentation tasks. Hence, a four-level pooling operation with pooling sizes 1/16, 1/8, 1/4 and 1/2 is adopted to down sample the input image. Then, all four-level features are concatenated into the corresponding feature maps in the up sampling and down sampling layers. Following the last up sampling operation, the original image is additionally concatenated before the final estimation.

*Decoder* The decoder structure is carefully designed with a set of residual and dense blocks. It is designed to improve the image quality by increasing the spatial resolution and enhancing the feature diversity. As shown in Fig. 3, each up sampling block in the decoder is composed of a dense bottleneck layer (DenseBlock D) and a transition block (TransBlock D). The dense bottleneck layer consists of two ReLu and Conv layers applied one after another, which is then concatenated with the input. This allows the network

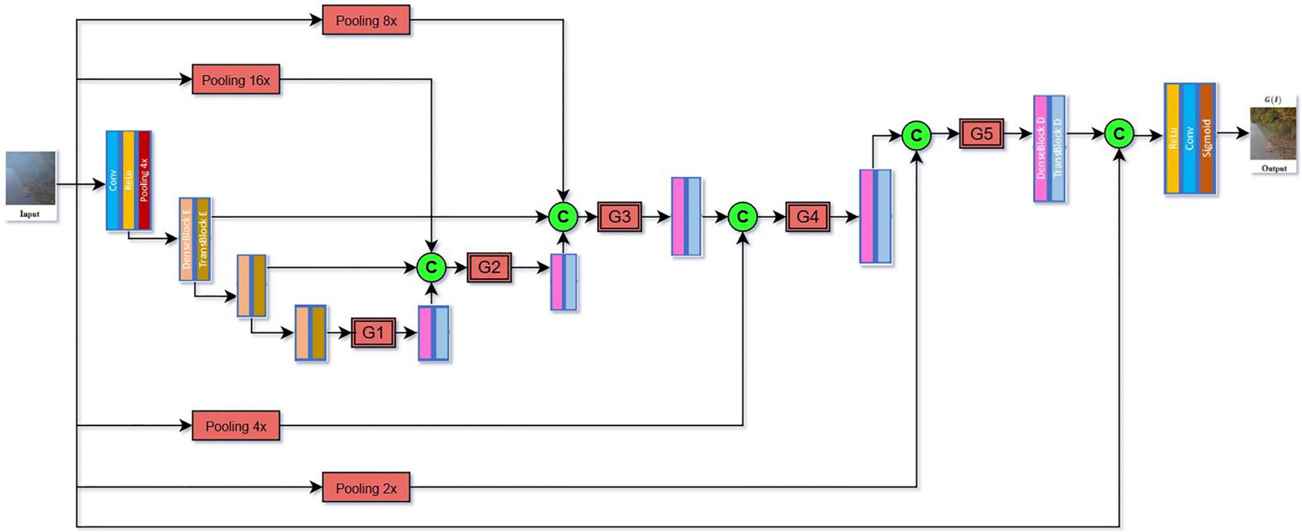

**Fig. 3** Network architecture

to learn more diverse and complex features. While the transition layer consists of a ReLU followed by a Conv layer which is then followed by a 2D transpose convolution for up sampling. The transition block reduces the number of channels and increases the spatial resolution by using a transpose convolution, which is a convolution that swaps the input and output dimensions.

Before being passed on to the group layer, the output from each up sampling block is concatenated with the corresponding feature maps from the down sampling layer and pyramid pooling layer. The pyramid pooling layer is a module that captures global context information by applying different levels of pooling and up sampling operations. The purpose of the pyramid pooling layer is to aggregate features from different regions of the image, which can help to preserve the details and avoid losing information due to down sampling.

The group layer structure is based on FFANet [15] and combines two basic residual block structures with a skip connection. The motivation behind the residual blocks is residual learning in ResNet [37], in which network layers learn residual functions with reference to the layer inputs rather than learning unreferenced functions. The residual blocks learn residual functions that are easier to optimize than unreferenced functions, and the skip connection helps to preserve the low-level information and avoid gradient vanishing.

A basic residual block structure, as shown in Fig. 4, consists of local residual learning and a Feature Attention (FA) module. Local residual learning allows less significant information to be transferred across several local residual connections, such as a low-frequency zone.

*Pixel and Channel Attention* Feature Attention is made up of channel attention and pixel attention, which can offer you more versatility when dealing with different forms of data. The concept of channel attention assumes that each channel feature has its own weighting regarding its importance in the data set. This is accomplished by gathering global spatial information from each individual channel and utilizing global average pooling to create a feature descriptor. The weights for these channels are then calculated using two convolution layers, followed by sigmoid or ReLu activation functions before being multiplied elementwise against the input features themselves. Pixel attention is a powerful tool for analyzing images that can provide more accurate results than traditional channel-wise analysis. By focusing on individual pixels, it allows us to identify important areas of an image with greater precision. This is especially useful when dealing with uneven haze distribution in an image, as the PA module enables the network to focus on thicker hazed pixels and high-frequency regions which contain more information about the scene or object being captured.

### 3.3.2 Discriminator

The discriminator in this model plays an important role in guiding the generator to generate more realistic dehazed results. Our discriminator is similar to that of [38], which consists of four convolutional layers with a stride of two and determines whether each $N \times N$ patch in an image is from the ground truth or the generator. Each convolutional feature map is fed into the next convolutional layer for regularization after passing through a group normalization layer and a Leaky-ReLu. The discriminator model is shown in Fig. 5.

As shown in Fig. 5, the discriminator is a simple architecture that alternates between several layers. The first and the layer before the last conv layer are comprised of a Conv and Leaky-ReLu operations. The first layer is followed by

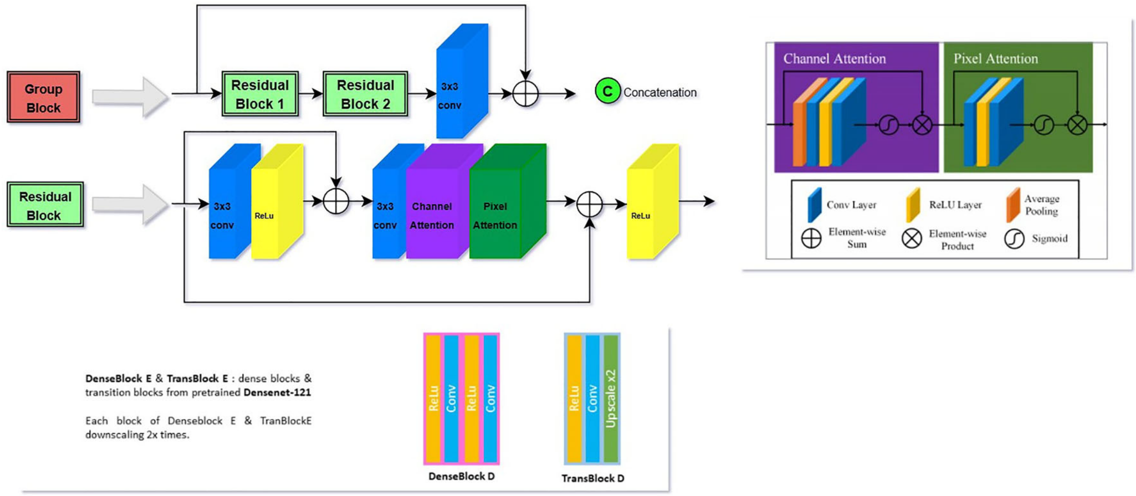

**Fig. 4** Building blocks explanation

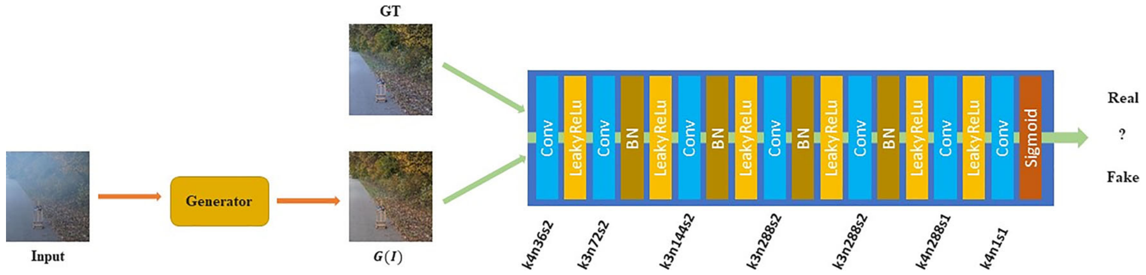

**Fig. 5** Discriminator

four layers that contain Conv, Batch Norm and Leaky-ReLu layers. The last Conv layer is followed by a sigmoid layer. The variables at the bottom of the figure, denoted by $k$, $n$ and $s$, represent the kernel size, number of channels and stride, respectively.

## 3.4 Losses and evaluation metrics

When training a CNN-based reconstruction network, it is important to choose the correct loss function because more conventional techniques like L2 error can result in hazy output. This problem has been addressed in recent works that makes use of new loss functions. During the experiment, we used an improvised loss to train the network on the two datasets and compare the results.

### 3.4.1 Improvised loss

For our improvised loss, we can start by denoting our dehaze image as $\hat{I}$. $I_{\text{gt}}$ and $I_{\text{hazy}}$ are, respectively, the ground truth image and hazy image. $G$ and $D$ represent the generator and discriminator respectively.

*Smooth L1 Loss* The smooth L1 loss is a robust version of the mean squared error (MSE) loss, which penalizes large errors less than small errors. This helps to avoid outliers and gradient explosion. The smooth L1 loss measures the pixel-level difference between the dehazed image and the ground truth image. The smooth L1 Loss can be described as:

$$\mathcal{L}_{\text{smooth}-L1} = \frac{1}{3N} \sum_{i=1} \sum_{c=1} \alpha\left(\hat{I}_c(i) - I_c^{\text{gt}}(i)\right) \tag{3}$$

where,

$$\alpha(e) = \begin{cases} 0.5e^2, & \text{if } |e| < 1 \\ |e| - 0.5, & \text{otherwise} \end{cases} \tag{4}$$

Here, $\hat{I}_c\ (i)$ and $I_c^{\text{gt}}(i)$ denote the intensity of the $c$-th channel of pixel $i$ in the dehazed image and in the ground truth image, respectively, and $N$ is the total number of pixels.

*Perceptual Loss* The perceptual loss is based on the feature maps extracted from a pre-trained VGG-19 network. The perceptual loss measures the semantic similarity between the dehazed image and the ground truth image, by comparing their high-level features. This helps to preserve the content and structure of the image. In addition to pixel-wise supervision, we utilize the VGG16 loss network pre-trained on

ImageNet to evaluate perceptual similarity.

$$\mathcal{L}_{\text{per}} = \sum_{j=1}^{3} \frac{1}{C_j H_j W_j} \phi_j \left\| \left(I^{gt}\right) - \phi_j\left(\hat{I}\right)_2^2 \right\| \tag{5}$$

where $H_j$, $W_j$, and $C_j$ represent the height, width, and channel of the feature map in the $j$-th layer of the backbone network, and finally, $\phi_{\text{j}}$ is the activation of the $j$-th layer.

*MS-SSIM Loss* The MS-SSIM loss measures the contrast and luminance similarity between the dehazed image and the ground truth image, by comparing their local statistics at multiple scales. This helps to preserve the contrast and brightness of the image. Let's use the terms $O$ and $G$ to represent two windows with a common size that are each centered at pixel $i$ in the haze-free image and the dehazed image, respectively. After using a Gaussian filter to $O$ and $G$, we can compute the resulting means $\mu_O$, $\mu_G$, standard deviations $\sigma_O, \sigma_G$ and covariance $\sigma_{OG}$. After defining the terms, the SSIM for pixel i can be defined as:

$$\begin{aligned}\text{SSIM}(i) &= \frac{2\mu_O\mu_C + C_1}{\mu_O^2 + \mu_G^2 + C_1} \cdot \frac{2\sigma_{OG} + C_2}{\sigma_O^2 + \sigma_G^2 + C_2} \\ &= l(i) \cdot \text{cs}(i)\end{aligned} \tag{6}$$

where *l(i)* represents luminance, and cs*(i)* represents contract, and structure measures, $C_1$, $C_2$, are two variables to stabilize the division with weak denominator. Finally, MS-SSIM loss is computed using $M$ levels of SSIM, which is defined as:

$$\mathcal{L}_{\text{MS}} - \text{SSIM} = 1 - \text{MS} - \text{SSIM} \tag{7}$$

where

$$\text{MS} - \text{SSIM} = l_M^{\alpha}(i) \cdot \prod_{m=1}^{M} \text{cs}_m^{\beta_m}(i) \tag{8}$$

with $\alpha$ and $\beta m$ being default parameters. In the Multi-Scale Structural Similarity Index (MS-SSIM) formula, the parameters $\alpha$ and $\beta m$ play a significant role in determining the weights of different components of the index. The $\alpha$ parameter controls the influence of the luminance comparison on the overall index. A higher value of $\alpha$ gives more weight to the luminance component, making it more important in the final score. The $\beta m$ parameters control the influence of the contrast comparison at different scales on the overall index. Each βm value corresponds to a specific scale, with m ranging from 1 to $M$. Higher values of βm give more weight to the contrast component at that scale, making it more important in the final score. By adjusting these parameters, you can customize the MS-SSIM index to emphasize certain aspects of image quality over others.

*Adversarial Loss* The adversarial loss $l_{\text{adv}}$ is defined using the discriminator.

*D(G(*$I^{\text{hazy}}$*))* probabilities across all training samples. It is described as:

$$\mathcal{L}_{\text{adv}} = \sum_{n=1}^{N} -\log D\left(G\left(I^{\text{hazy}}\right)\right) \tag{9}$$

where *D(G(*$I_{\text{hazy}}$*))* is the probability of reconstructed image *G(*$I_{\text{hazy}}$*)* to be a haze-free image.

*WGAN-GP Loss* The WGAN-GP loss is an enhanced version of the Wasserstein GAN (WGAN) loss, utilizing a gradient penalty term to enforce a Lipschitz constraint on the discriminator, thereby minimizing the Wasserstein distance between real and fake image distributions. This helps to stabilize the training process and avoid mode collapse. To obtain the WGAN value function, the Kantorovich–Rubinstein duality is used.

$$\min_G \max_{D\in\mathcal{D}} \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)] - \mathop{\mathbb{E}}_{\tilde{\mathbb{X}}\sim\mathbb{P}_g}\left[D(\tilde{x})\right] \tag{10}$$

where $D$ is the collection of 1-Lipschitz functions, and $P_g$ is once more the distribution of the model that is implicitly specified by $\hat{x} = G(z)$. Because the gradient of the critic function produced by the WGAN value function behaves better with regard to its input than that of the GAN counterpart, it is simpler to optimize the generator. The interaction between the weight constraint and the loss function complicates WGAN training and results in exploding or vanishing gradients. Gradient Penalty works by enforcing a constraint that requires the gradients of the critic's output with regard to the inputs to have a unit norm.

$$\begin{aligned}&L_{\text{wgan - gP}} \\ &= \underbrace{\mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}\left[D(\tilde{x})\right] - \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(\boldsymbol{x})]}_{\text{Original critic loss}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_{\mathbf{x}}}\left[\left(\nabla_{\hat{x}} D(\hat{x})_2 - 1\right)^2\right]}_{\text{Gradient penalty}}\end{aligned} \tag{11}$$

*Total Loss* To supervise the training of our dehazing network, we integrate the smooth L1 loss, perceptual loss, MS-SSIM loss, adversarial loss, and WGAN-GP loss.

$$\begin{aligned}L_{\text{improv}} &= L_{\text{smooth}-\text{L1}} + \alpha L_{\text{MS}-\text{SSIM}} + \beta L_{\text{per}} \\ &\quad + \gamma L_{\text{adv}} + \phi L_{\text{wgan}-\text{gp}}\end{aligned} \tag{12}$$

where $\alpha = 0.2$, $\beta = 0.5$ and $\gamma = 0.0005$, and $\phi = 1$ are the hyperparameters weighting for each loss functions. The hyperparameters weighing for each loss function in the $L_{\text{improv}}$ equation are selected based on the specific requirements of the dehazing network and the training data after performing several experiments. In general, these weights are chosen to balance the contribution of each loss function to the overall objective of the network. A lower value of $\alpha$ means that its associated loss has less impact on the final score. A higher value of $\beta$ means that its associated loss has more impact on the final score. A lower value of $\gamma$ means that its loss has less impact on the final score. It is important to note that these values are not fixed and can be adjusted based on your specific use case and requirements. The main motivation for combining these different loss functions is to leverage their complementary strengths and overcome their limitations. By using multiple criteria to evaluate the dehazed image, we can ensure that it has high quality in terms of pixel-level accuracy, perceptual similarity, structural similarity, realism, and diversity. Moreover, by balancing these criteria with appropriate weights, we can avoid overfitting or underfitting to any specific aspect of the image.

### 3.4.2 Evaluation metrics

Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM), which are typically used as a criterion to evaluate image quality in the image dehazing task, are the evaluation metrics we adopt to evaluate the performance of our method.

*PSNR* PSNR is defined as the ratio of a signal's maximum possible power to the power of corrupting noise that influences the fidelity of its representation. It is commonly used to regulate the quality of digital signal transmission. In the case of images, each pixel can be thought of as a component of an 8-bit RGB signal. It is described as:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}_I^2}{\mathrm{MSE}}\right) \tag{13}$$

Here, $\mathrm{MAX}_I$ is the maximum valid value for a pixel, and MSE represents the Mean Squared error between the output image and the target image.

*SSIM* SSIM is a metric that attempts to emulate the operation of the human visual system (HVS color model). It is built around three components: correlation, luminance distortion, and contrast distortion. Instead of a direct pixel-by-pixel comparison, this index is generated on several image windows. If *x, y* are windows of size $N \times N$ in images:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{14}$$

where $\mu_x$ is the average of $x$; $\mu_y$ is the average of $y$; ${\sigma_x}^2$ is the variance of $x$; ${\sigma_y}^2$ is the variance of $y$; $\sigma_{xy}$ is the covariance of $x$ and $y$; $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ are two variables to stabilize the division with weak denominator; $L$ is the dynamic range of the pixel-values (typically this is $2^{\#\text{bits per pixel}}$—1); and $k_1$ = 0.01 and $k_2$ = 0.03 by default.

**Table 2** List of hyperparameters used to train the model

| Hyperparameters | Values |
|---|---|
| Input size | 512 × 512 × 3 |
| Learning rate | 1.00E−04 |
| Weight decay | 1.00E−02 |
| Batch size | 1 |
| Hidden layer activation | ReLu for Gen./Leaky-ReLu for disc |
| Normalization layer | Group norm |
| Optimizer | Lion |
| Loss function | Improvised loss |
| Metrics | PSNR and SSIM |
| No. of epochs | 200 |
| Generator output size | 512 × 512 × 3 Discriminator output size 2 |

### 3.4.3 Implementation details

The proposed model was implemented using the Python programming language and PyTorch Version 1.12.1 as its backend. Due to constrained memory and computational resources, the model was trained for 200 epochs with a batch size of 1. Due to this reason, each layer of the network was normalized and made more stable by using group norm instead of batch norm. Lion (EvoLved Sign Momentum) optimizer with a learning rate of 0.0001 and weight decay of 0.01 was used to replace Adam. It is a new optimizer discovered by Google Brain that is purportedly to be better than Adam(w), in Pytorch. We used StepLR as a scheduler to adjust the learning rate. StepLR is a scheduler used to adjust the learning rate of each parameter group. It works by decaying the learning rate every 20 epochs, with a decay factor of gamma = 0.5. This decay can happen in addition to any other changes made to the learning rate from outside sources. The detailed information about the hyperparameter values that were provided during the training of the model is given in Table 2.

## 3.5 Results and discussion

In this section, we conduct a series of experiments on two synthetic datasets (NTIRE 2018 and SOTS) to demonstrate the effectiveness of the proposed methodology. These datasets are artificially generated, so we have access to the ground truth images for the validation sets, which enable us to assess the performance both qualitatively and quantitatively. This is a valuable tool for determining how well our proposed methodologies can be applied in real world scenarios.

The first experiment conducted was done using the NTIRE 2018 dataset where we trained and tested both the indoor and outdoor datasets, using our own novel approach with improvised loss. The second experiment was performed using the SOTS dataset where again we trained and tested both the indoor and outdoor datasets using the proposed model.

One possible reason to use SOTS and NTIRE 2018 datasets instead of more recent ones like NTIRE 2021 to train our image dehazing model is that they have more diverse and realistic images of hazy scenes. SOTS contains both indoor and outdoor images, while NTIRE 2018 covers various weather conditions and haze levels. These datasets can help our model learn more generalizable features and avoid overfitting to a specific domain or scenario. On the other hand, NTIRE 2021 focuses on extreme weather conditions, such as heavy fog, snow, and rain, which may not be representative of the common cases of image dehazing. Moreover, NTIRE 2021 has fewer images than SOTS and NTIRE 2018, which may limit the amount of data available for training and testing our model.

Table 3 shows the quantitative performance results (SSIM and PSNR) using both datasets. As can be observed from the table, our proposed method achieves best performance for the outdoor images and comparable performance in the indoor images on the NTIRE 2018 dataset. The model achieved a PSNR and SSIM of 14.7 and 0.54 for the indoor images and 16.54 and 0.54 for the outdoor images, respectively. For the SOTS dataset, the model achieved a PSNR and SSIM of 23.98 and 0.87 for the indoor images, and 19.88 and 0.83 for the outdoor images, respectively. Figure 6 shows the dehazed results of indoor and outdoor images from the NTIRE 2018 and SOTS datasets, respectively.

As shown in Fig. 6, the performance of the method varies depending on the type of hazy image. Specifically, the method works better on outdoor hazy images than on indoor hazy images. This can be explained by the fact that outdoor hazy images have more homogeneous and uniform haze distribution. On the other hand, indoor hazy images have more complex and varying haze distribution. Therefore, the proposed method may introduce artifacts or over-enhancement on indoor hazy images, while preserving the naturalness and details of outdoor hazy images. To improve the method, one possible direction is to use a deep learning model to learn the transmission map from a large dataset of hazy and clear images.

### 3.6 Ablation study

To conduct an ablation study, we compare our full model with different variants that remove one or more components of the improved loss function. We also compare our model with some baseline models that use different architectures or loss functions. The following table summarizes the results of our ablation study on two benchmark datasets: NTIRE 2018 and SOTS.

From this Table 4, we can see that our full model achieves the best performance on both datasets in terms of PSNR and SSIM, which demonstrates the effectiveness of our network and our improved loss function. We can also see that each component of the improved loss function contributes to the performance improvement, as removing any of them leads to a drop in the metrics. Among the five components, the WGAN-GP loss has the most impact, as it improves the stability and convergence of the GAN training and prevents mode collapse. The perceptual loss and the MS-SSIM loss also have significant effects, as they capture the high-level features and the structural details of the images. The smooth L1 loss and the adversarial loss have less influence, but they still help to reduce the pixel-wise and feature-wise errors.

We can also see that our model outperforms the baseline models that use different architectures or loss functions. Baseline 1 uses a U-Net architecture and a L2 loss function, which is a common choice for image restoration tasks. However, this model produces blurry images with low contrast and saturation. Baseline 2 uses a U-Net architecture and a Smooth L1 loss function, which is slightly better than L2 in preserving edges and details. However, this model still suffers from color distortion and haze residue. Baseline 3 uses a U-Net architecture and a perceptual loss function, which is based on the features extracted by a pre-trained VGG-19 network. This model improves the perceptual quality of the images, but it also introduces some artifacts and noise. Baseline 4 uses a U-Net architecture and a MS-SSIM loss function, which is based on the structural similarity between images at multiple scales. This model enhances the structural details of the images, but it also amplifies some haze and reduces the contrast. Baseline 7 uses a U-Net architecture and our improved loss function, which is a combination of L1, perceptual, MS-SSIM, adversarial, and WGAN-GP losses. This model achieves better results than the previous baselines, but it still has some limitations in terms of image quality and diversity. Baseline 8 uses a ResNet architecture and our improved loss function, which is similar to our model except for the generator architecture. This model produces clearer images than baseline 5, but it still lags behind our model in terms of PSNR and SSIM.

### 3.7 Comparison with state-of-the-art studies

In this section, we will quantitatively evaluate the performance of DC-GAN with previous state-of-the-art image dehazing techniques. Table 5 compares the quantitative performance of the proposed technique to other recent methods on indoor and outdoor images from the NTIRE 2018 dataset.

**Table 3** Performance of DC-GAN on the two datasets

| Dataset | Indoor | | Outdoor | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| NTIRE 2018 | 14.7 | 0.54 | 16.54 | 0.54 |
| SOTS | 23.98 | 0.87 | 19.88 | 0.83 |

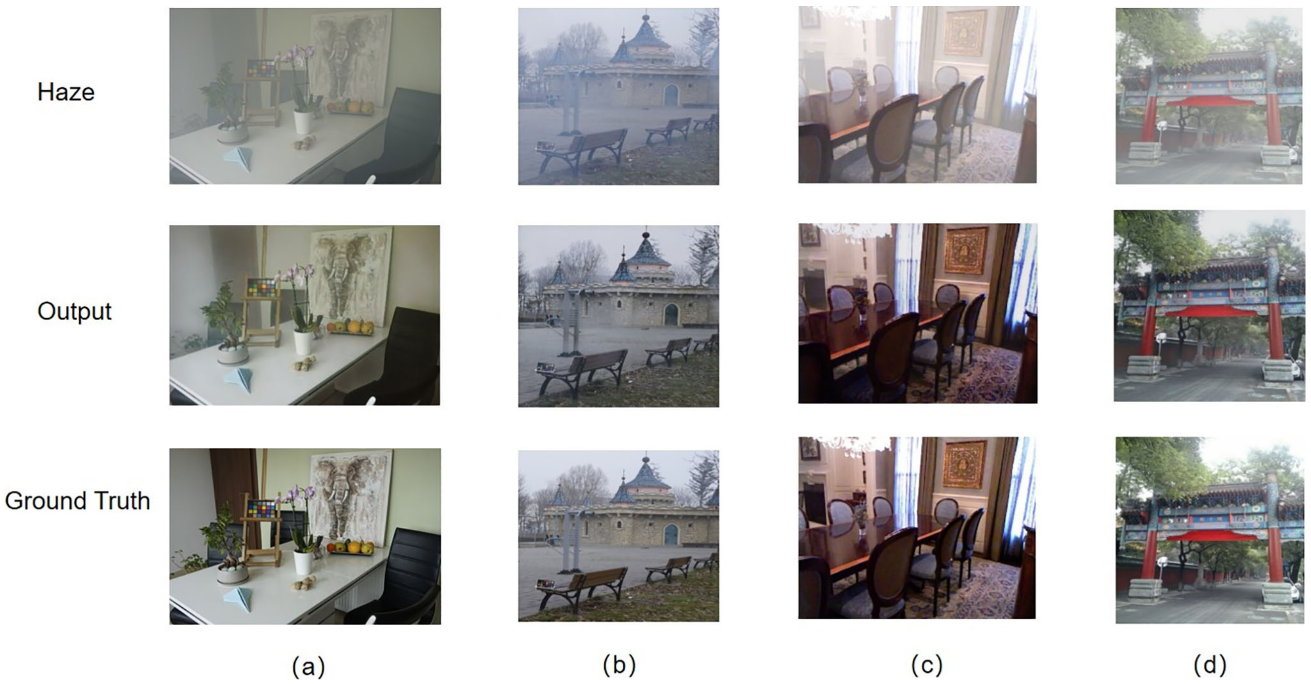

**Fig. 6** Qualitative results of DC-GAN on NTIRE 2018 and SOTS datasets. **a** and **b** represent indoor and outdoor images from NTIRE 2018 dataset, whereas **c** and **d** represent indoor and outdoor images from SOTS dataset, respectively

**Table 4** Ablation study using NTIRE 2018 and SOTS datasets for different losses on outdoor images

| Method | NTIRE 2018 | | SOTS | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| Baseline 1 (U-Net + L2) | 14.70 | 0.28 | 16.18 | 0.57 |
| Baseline 2 (U-Net + $L_{\text{smooth L1}}$) | 15.10 | 0.31 | 16.22 | 0.59 |
| Baseline 3 (U-Net + $L_{\text{per}}$) | 15.22 | 0.36 | 17.34 | 0.64 |
| Baseline 4 (U-Net + $L_{\text{MS-SSIM}}$) | 16.01 | 0.42 | 17.68 | 0.69 |
| Baseline 5 (U-Net + $L_{\text{adv}}$) | 16.08 | 0.44 | 17.73 | 0.74 |
| Baseline 6 (U-Net + $L_{\text{wgan-gp}}$) | 16.12 | 0.47 | 18.53 | 0.76 |
| Baseline 7 (U-Net + $L_{\text{improv}}$) | 16.26 | 0.49 | 19.03 | 0.80 |
| Baseline 8 (ResNet + $L_{\text{improv}}$) | 16.40 | 0.51 | 19.45 | 0.81 |
| Ours—$L_{\text{smooth L1}}$ | 15.67 | 0.45 | 19.23 | 0.79 |
| Ours—$L_{\text{per}}$ | 15.55 | 0.38 | 17.64 | 0.74 |
| Ours—$L_{\text{MS-SSIM}}$ | 15.10 | 0.35 | 17.42 | 0.71 |
| Ours—$L_{\text{adv}}$ | 15.92 | 0.42 | 18.80 | 0.78 |
| Ours—$L_{\text{wgan-gp}}$ | 14.50 | 0.31 | 16.72 | 0.65 |

**Table 5** Quantitative comparisons on NTIRE 2018 for different methods

| Method | Indoor | | Outdoor | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| He et al. (CVPR'09) | 14.43 | 0.75 | 16.78 | 0.65 |
| Zhu et al. (TIP'15) | 12.24 | 0.61 | 16.08 | 0.59 |
| Ren et al. (ECCV'16) | 15.22 | 0.75 | 17.56 | 0.64 |
| Berman et al. (CVPR'16) | 14.12 | 0.65 | 15.98 | 0.58 |
| Li et al. (ICCV'17) | 13.98 | 0.73 | 15.03 | 0.53 |
| Ours | 14.7 | 0.54 | 16.54 | 0.54 |

The method proposed by Ren et al. (ECCV'16) achieves the highest PSNR and SSIM values for both indoor and outdoor images. Their method achieves a PSNR of 15.22, SSIM of 0.75 for indoor images and PSNR of 17.56, SSIM of 0.64 for outdoor images, while our model achieves PSNR of 14.7, SSIM of 0.54 for indoor images, PSNR of 16.54 and SSIM of 0.54 for outdoor images on the NTIRE 2018 dataset. However, the difference between their method and our model is not very large, especially for outdoor images. Table 5 shows that the performance of different methods varies depending on the type and quality of the images. Therefore, it is not possible to draw a definitive conclusion about which method is superior to the others for all kinds of images. One of the main limitations of our method is that it relies on a fixed set of parameters that are tuned for a specific dataset. This makes it difficult to generalize to new images that have different characteristics or noise levels. Another limitation is that our method does not explicitly handle occlusions or missing data, which can degrade the quality of the reconstructed images. Furthermore, our method does not incorporate any semantic information or prior knowledge about the scene, which could potentially improve the results.

The quantitative comparisons using the SOTS dataset are shown in Table 6. In terms of PSNR and SSIM metrics, our proposed DC-GAN outperforms the majority of the state-of-the-art methods.

As shown in Table 6, our model outperforms the existing methods on the indoor images, achieving the highest PSNR and SSIM values. This indicates that our model can effectively remove the haze and preserve the details and colors of the indoor images. We evaluated our model on the SOTS dataset, which is a benchmark dataset for image dehazing. The dataset contains indoor and outdoor images with different levels of haze. We initially compared our model with several benchmarks, such as DCP, DehazeNet, AOD-Net, GFN. However, all the approaches are very old. To further demonstrate the effectiveness of our model, we also compared our model with some newer methods that have been presented in recent years, such as MSPCNN, DCPDN, FPCNet, and GCANet. These methods are based on more advanced techniques, such as multi-scale processing, dense connections, feature pyramid fusion, and graph convolutional networks. Our model still achieves the best performance on the indoor images, surpassing the newer methods by a large margin in terms of PSNR and comparable SSIM values. However, on the outdoor images, our model falls significantly behind the newer methods, indicating that our model has some limitations for outdoor scenes. We attribute this gap to two main reasons: first, our model does not exploit multi-scale information or feature fusion techniques, which may help to capture more details and context information from different scales; second, our model does not incorporate any prior knowledge or physical constraints, which may help to reduce the ambiguity and noise in outdoor scenes. To address these issues, we plan to extend our model in future work by incorporating multi-scale processing and feature fusion techniques into our network architecture, as well as introducing some prior knowledge or physical constraints into our loss function or regularization term.

## 4 Conclusion

In this paper, a Densely Connected-GAN with feature attention was proposed for the task of image dehazing. Dense blocks offer the benefit of reducing network parameters, deepening network layers, improving feature propagation, avoiding the vanishing-gradient problem, and expanding receptive fields. They also handle the issue of decreasing network performance as network depth increases. We also proposed feature attention units, which use inter-spatial and inter-channel feature correlations to generate spatial and channel attention. Channel attention is based on the assumption that each channel feature has its own weighting, and pixel attention is a useful approach for studying images that can provide more accurate results than conventional channel-wise analysis. Finally, an improvised loss-based WGAN-GP was utilized to train the model using the NTIRE 2018 and SOTS datasets. Our model has a considerable advantage in restoring image detail and color fidelity, and it is expected

**Table 6** Quantitative comparisons on SOTS for different methods

| Methods | Indoor | | Outdoor | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| DCP | 16.62 | 0.81 | 19.13 | 0.81 |
| AOD-Net | 19.06 | 0.85 | 20.29 | 0.87 |
| DehazeNet | 21.14 | 0.84 | 22.46 | 0.85 |
| GFN | 22.30 | 0.88 | 21.55 | 0.84 |
| MSPCNN | 22.63 | 0.88 | 21.76 | 0.86 |
| DCPDN | 23.15 | 0.89 | 22.47 | 0.87 |
| FPCNet | 23.45 | 0.90 | 22.66 | 0.88 |
| GCANet | 23.67 | 0.91 | 22.83 | 0.89 |
| Ours | 23.98 | 0.87 | 19.88 | 0.83 |

to address a wide range of low-level vision issues such as super-resolution and denoising.

**Acknowledgements** We want to thank Northwestern Polytechical University for helping us to conduct research in this area and providing resources.

**Author contributions** Tewodros Tassew wrote the main manuscript text and Nie Xuan contributed and lead the research and help review the manuscipt.

**Availability of data and materials** The NTIRE 2018 and RESIDE datasets are both publicly available on (NTIRE 2018: https://data.vision.ee.ethz.ch/cvl/ntire18/; SOTS: https://sites.google.com/view/reside-dehaze-datasets/reside-standard?authuser=3D0).

**Code availability** Code that supports the findings of this study will be available for noncommercial academic purposes and will require a formal code use agreement. Please contact ted.meg1234@mail.nwpu.edu.cn for access.

## Declarations

**Competing interests** The authors declare no competing interests.

## References

1. Fabio, D.R., Fabio, D., Carlo, P.: Profiling core-periphery network structure by random walkers. Sci. Rep. **3**, 1467 (2013)
2. Jobson, D.J., Rahman, Z.U., Woodell, G.A.: Properties and performance of a center/surround retinex. IEEE Trans. Image Process. **6**, 451–462 (1997)
3. Rahman, Z., Jobson, D.J., Woodell, G.A.: Image enhancement, image quality, and noise. Proc. SPIE Int. Soc. Opt. Eng. **6**, 451–462 (2005)
4. He, K., Tang, X.: Single image haze removal using dark channel prior. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
5. Fattal, R.: Single image dehazing. ACM transactions on graphics. IEEE Conf. Comput. Vis. Patt. Recogn. **27**, 547–555 (2008)
6. Xu, H., Guo, J., Liu, Q.: Fast image dehazing using improved dark channel prior. IEEE **27**, 663–667 (2012)
7. Khatun, A., Haque, M.R., Basri, R., Uddin, M.S.: Single image dehazing: An analysis on generative adversarial network. Journal of Computer and Communications **8** (2020)
8. Yang, F., Zhang, Q.: Depth aware image dehazing. Vis. Comput. **38**, 1–9 (2021)
9. Liu, Z., Xiao, B., Alrabeiah, M., Wang, K., Chen, J.: Generic model-agnostic convolutional neural network for single image dehazing. arXiv preprint arXiv:1810.02862 (2018)
10. Cheng, Z., You, S., Ila, V., Li, H.: Semantic single-image dehazing. http://arxiv.org/abs/1804.05624 (2018)
11. Cai, B., Xu, X., Jia, K., Qing, C., Tao, D.: Dehazenet: an end-to-end system for single image haze removal. IEEE Trans. Image Process. **25**, 5187–5198 (2016)
12. Li, B., Peng, X., Wang, Z., Xu, J., Feng, D.: AOD-Net: All-in-one dehazing network. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 22–29 (2017)
13. Li, B., Peng, X., Wang, Z., Xu, J., Feng, D.: End-to-end united video dehazing and detection. In: AAAI (2017)
14. Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., Hua, G.: Gated context aggregation network for image dehazing and deraining. In: IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1375–1383 (2019)
15. Qin, X., Wang, Z., Bai, Y., Xie, X., Jia, H.: FFA-Net: feature fusion attention network for single image dehazing. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11908–11915 (2020)
16. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y.: Generative Adversarial Nets. arXiv:1406.2661v1 (2014)
17. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: International Conference on Learning Representations, ICLR (2016)
18. Arjovsky, C.S..B.L. M.: Wasserstein generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning, Proceedings of Machine Learning Research, vol. 70, pp. 214–223 (2017)
19. Mirza, M., Osindero, S.: Conditional Generative Adversarial Nets. arXiv:1411.1784 (2014)
20. Zhu, J., Park, T., Isola, P., Efros, A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2242–2251 (2017)
21. Malav, R., Kim, A., Sahoo, S.R., Pandey, G.: DHSGAN: An End-to-end dehazing network for fog and smoke. In: Computer Vision-ACCV, pp. 593–608 (2018)

22. Yang, X., Xu, Z., Luo, J.: Towards perceptual image dehazing by physics-based disentanglement and adversarial training. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence (2018)
23. Engin, D., Genc, A., Ekenel, H.K.: Cycle-dehaze: Enhanced cycle gan for single image dehazing. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 938–9388 (2018)
24. Liu, W., Hou, X., Duan, J., Qiu, G.: End-to-end single image fog removal using enhanced cycle consistent adversarial networks. In: IEEE Transactions on Image Processing, vol. 29, pp. 7819–7833 (2020)
25. Li, R., Pan, J., Li, Z., Tang, J.: Single image dehazing via conditional generative adversarial network. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8202–8211 (2018)
26. Dong, Y., Liu, Y., Zhang, H., Chen, S., Qiao, Y.: FD-GAN: generative Adver-sarial networks with fusion-discriminator for single image dehazing. In: Proceedings of the AAAI, pp. 10729–10736 (2020)
27. Mehta, A., Sinha, H., Narang, P., Mandal, M.: Hidegan: a hyperspectral-guided image dehazing gan. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 846–856 (2020)
28. Yifan, L., Siyuan, F., Zhang, X., Xie, N.: Denoising Monte Carlo renderings via a multi-scale featured dual-residual GAN. Vis. Comput. **37**, 09 (2021)
29. Wang, C., Xing, X., Yao, G., Zhixun, S.: Single image deraining via deep shared pyramid network. Vis. Comput. **37**, 07 (2021)
30. Zhang, H., Sindagi, V., Patel, V.M.: Image de-raining using a conditional generative adversarial network. IEEE Trans. Circuits Syst. Video Technol. **30**(11), 3943–3956 (2020)
31. Amaranageswarao, G., Deivalakshmi, S., Ko, S.: Joint restoration convolutional neural network for low-quality image super resolution. Vis. Comput. **38**, 31–50 (2020)
32. Ma, T., Tian, W.: Back-projection-based progressive growing generative adversarial network for single image super-resolution. Vis. Comput. **37**, 05 (2021)
33. Wenlong, Z., Yihao, L., Dong, C., Qiao, Y.: Ranksrgan: generative adversarial networks with ranker for image super-resolution. IEEE Trans. Pattern Anal. Mach. Intell. **1**, 1 (2019)
34. Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., Wang, Z.: Benchmarking Single Image Dehazing and Beyond. ArXiv e-prints (2017)
35. Ancuti, C.O., Ancuti, C., R., T., De Vleeschouwer, C.: I-haze: A Dehazing Benchmark with Real Hazy and Haze-Free Indoor Images. ArXiv e-prints (2018)
36. Ancuti, C.O., Ancuti, C., R., T., De Vleeschouwer, C.: O-haze: A Dehazing Benchmark with Real Hazy and Haze-Free Outdoor Images. ArXiv e-prints (2018)
37. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
38. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1125–1134 (2017)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.