Single image defogging and evaluation with deep neural networks without the need of ground truth

Gerard DeMas-Giménez[a], Pablo García-Gómez[b], Josep R. Casas[c], and Santiago Royo[a,b]

[a]Center for Sensors, Instruments and System Development (CD6), Universitat Politècnica de Catalunya (UPC), Rambla de Sant Nebridi 10, Terrassa, Spain
[b]Beamagine S.L., Bellesguard 16, Castellbisbal, Spain
[c]Image Processing Group (GPI), TSC Department, Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, Barcelona, Spain

ABSTRACT

Fog dramatically compromises the overall visibility of any scene, critically affecting features such as objects' illumination, contrast, and contours. The decrease in visibility compromises the performance of Computer Vision algorithms such as pattern recognition and segmentation, some of them very relevant to decision-making in the rise of autonomous-driven vehicles. Many dehazing methods have been proposed. However, to the best of our knowledge, all currently used metrics do compare the defogged image to its ground truth, usually the same scene on a non-foggy day, or estimate physical parameters from the scene. This hinders progress in the field, as obtaining proper ground truth images is not always possible and becomes costly and time-consuming because physical parameters greatly depend on the scene conditions. This work aims to tackle this issue by proposing a real-time operating defogging network that only takes an RGB image of the fogged scene as input, performs the defogging, and uses a contour-based metric for Single Image Defogging evaluation even when the ground truth is not available, which is the most common situation. The proposed metric only requires the original hazy image and the image after the defogging procedure. We trained our network using a novel two-stage pipeline with the DENSE dataset and compared our method and metric with currently used metrics and other defogging techniques with the NTIRE 2018 defogging challenge to prove their effectiveness.

Keywords: Image defogging, Image evaluation metrics, visual enhancement evaluation, edge detection, deep neural networks, generative adversarial networks, autonomous systems

1. INTRODUCTION AND STATE-OF-THE-ART

Adverse weather conditions such as fog, smoke, or haze critically compromise the visibility of any scene. Image processing algorithms perform poorly under such conditions. Furthermore, given the rise and advances in autonomous vehicles, there is a need to achieve a processing solution that reduces the effect of bad weather conditions, as the reduction of visibility can directly affect the judgment in an autonomous vehicle system. The process of enhancing the visibility in bad weather conditions is called defogging. The quantification of the defogging level attained is not an obvious process.

Nowadays there are several approaches to defog an image. Some of them are active and rely on using polarized light to get more information about the scene, or use non-visible cameras in the longer IR bands. There is also interest in applying Deep Neural Networks (DNNs) to these problems. Hence, this work offers a state-of-the-art solution to the fog problem in Computer Vision (CV) that relies on the information of an RGB image and the use of a DNN to be used in real-time applications. Later on, we will propose a metric to evaluate the improvements obtained by the DNN. Let us first understand the physics of the problem and how DNNs might be a solution for it.

Further author information: (Send correspondence to G.D.M.G)
G.D.M.G: E-mail: gerard.de.mas@upc.edu

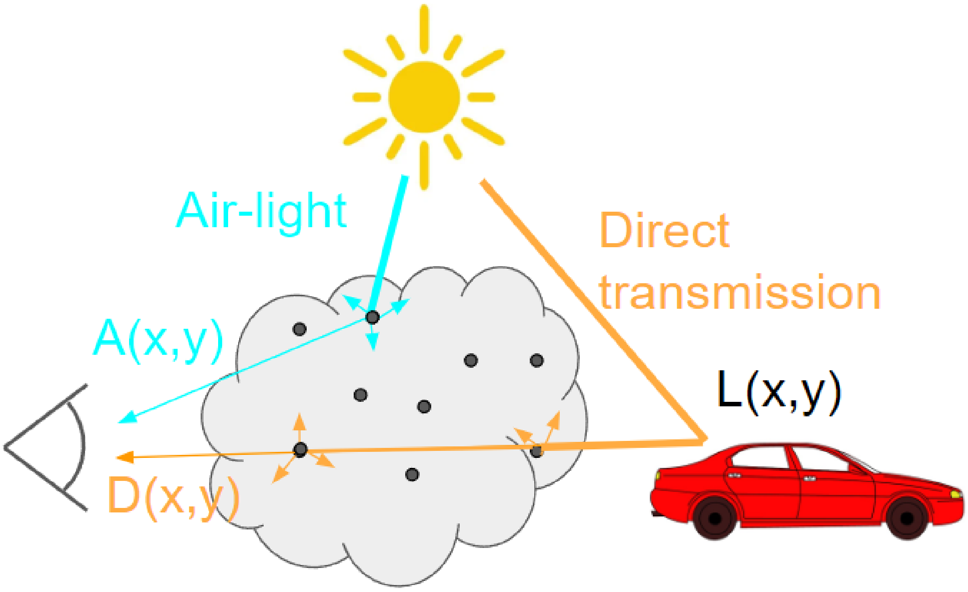

Figure 1. Image degradation model. Direct transmission correspond to the light coming from the object that has been attenuated due to the fog. Air-light correspond to the light that has been scattered directly from the fog particles.

1.1 Light scattering in fog

Fog is a common phenomenon that consists on tiny liquid water particles suspended in the air that absorb and scatter light resulting in reduced luminance and contrast, producing an overall degradation of the image. Imaging models through haze have been widely studied.[1] According to the image degradation model, the radiance reaching a detector in a foggy environment is a sum of two contributions, the direct transmission, $D(x, y)$ and the diffuse air-light field $A(x, y)$,

$$I(x, y) = D(x, y) + A(x, y). \tag{1}$$

As seen in Fig. 1, $D(x, y)$ contains information of the original scene, $L(x, y)$. Nevertheless, this information is still affected by the transmittance of the atmosphere $t(x, y)$. So,

$$D(x, y) = L(x, y) \cdot t(x, y). \tag{2}$$

Transmittance can be described as an attenuation that depends on the distance between the object and the observer, $t(x, y) = e^{-\beta d(x,y)}$, where, $d(x, y)$ is the distance between the object and the observer and β the attenuation coefficient of the atmosphere, which may vary depending on the weather conditions. On the other hand, the diffuse air-light field can be understood as,

$$A(x, y) = A_\infty[1 - t(x, y)], \tag{3}$$

where A_∞ is the radiance of an object at infinite distance. Knowing that, combining equations 1 , 2 and 3 we can estimate the real defogged image $L(x, y)$ as follows,

$$L(x, y) = \frac{I(x, y) - A(x, y)}{1 - A(x, y)/A_\infty}. \tag{4}$$

Therefore, the key to obtaining the defogged image is to properly estimate both $t(x, y)$ and A_∞. Unfortunately, estimating these parameters is not an easy task as they greatly depend on the conditions of the scene.

Some alternative strategies have been proposed to achieve this. Polarized light has been proposed to physically achieve a defogged image.[2] There are different polarimetric methods for dehazing. Schechner[2] presented a method using polarized-difference imaging which was quite successful. It was based on the premise that it exists a best $(I_\parallel)$ and worst $(I_\perp)$ polarization state when it comes to transmission through fog. The parameter A_∞ is

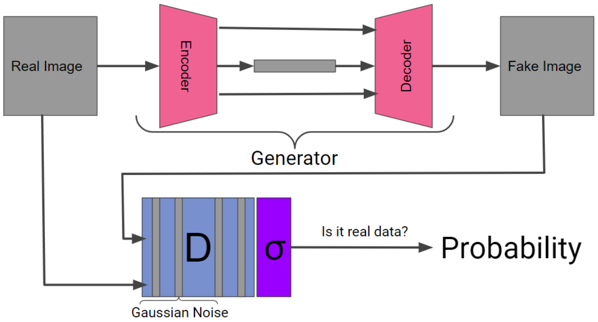

Figure 2. Schematic of our Network. The two components can be seen, the Generator with a U-Net schematic and the D, an encoder with Gaussian noise layers to decrease its performance.

computed as, $A_\infty = \frac{1}{2}(I_\parallel(sky) + I_\perp(sky))$. The issue with this method relies on the difficulty of finding a proper sky pixel in the image, and the acquisition of the best and worst polarization states, which often needs be done sequentially.

1.2 Deep Neural Networks

DNNs are Artificial Neural Networks (ANN) with many hidden layers between the input and output layers. These networks have been vastly used in CV and image processing among other fields. Due to the big number of connections between layers and neurons and the non-linear nature of the network, DNNs can find patterns from very complex data. The advancement in the Single Image Defogging (SID) field is usually evaluated in the NTIRE workshop.[3] This workshop proposes challenges in image and video processing in several fields, usually including homogeneous and non-homogeneous fog removal among the topics of interest. In these challenges, some research groups took advantage from previous information on the image and tried to evaluate the natural parameters (air-light and transmission map) through deep learning techniques.[4] Other groups took advantage of the generative capabilities of DNNs and used them to directly generate a defogged image from a foggy one without estimating any physical parameter.[5] The most common model used is based on the Generative Adversarial Network (GAN).[6] This network created a big impact in the field due to its ability to capture the training data distribution and generate new data from it. These models are made of two different networks, a Generator (G), and a Discriminator (D), and their usefulness arise from the adversarial competition between them. On the one hand, the main purpose of G is to produce "*fake*" data that resembles the training data. On the other hand, D's job is to identify if the input data is "*real*" or "*fake*". By "*real*" one means that it is part of the training dataset, and by "*fake*" that it is generated by G. During training, G continuously tries to fool D by generating better representations of the training data while D tries to adapt to these changes. GANs are extremely useful when it comes to generating "*fake*" data. The competition between G and D is done by a minmax function, where G wishes to minimize a cost function, whereas D tries to maximize it. Ideally, by the end of the training, this cost function should tend to $\frac{1}{2}$, as the probability of D getting the right output is completely random. At this point, our generated "*fake*" data should be indistinguishable from the "*real*" one.

The common minmax function between G and D is,

$$\textbf{minmax}\ V(D,G) = \log(D(x)) + \log(1 - D(G(z))) \tag{5}$$

Here, x represents the data of an image. Therefore, $D(x)$ is the probability that the image is "*real*", that we wish to be high, as we expect D to correctly guess "*real*" images as "*real*". On the other hand, z represents a latent vector. In the case that the input of G is also an image, then z and x would have the same dimensions. Knowing that, $G(z)$ is, therefore, a generated "*fake*" image. Then, $D(G(z))$ is the probability that D guesses a "*fake*" image as "*real*". Here is where the competition arises as G tries to bring this probability up, whereas D tries to bring it down.

1.3 Defogging metrics

In order to evaluate the effectiveness of the defogging networks, classical image processing metrics such as the Structural Similarity Index (SSIM) or the Peak Signal to Noise Ratio have been classically used to compare the defogged image with a Ground Truth (GT) of the scene. Nevertheless, classical CV algorithms for evaluation, as the mentioned above, perform poorly when it comes to quantifying an enhancement in the scene's visibility. Interestingly, these metrics need always a GT image (that is, the image of the same scene without fog) to be implemented. Needless to say, obtaining ground truth images in adverse weather conditions is costly, time-consuming and, often, simply unfeasible. In natural conditions fog is a time-variant and complex weather phenomenon. Reproducing the same scene for acquiring images without fog but with equivalent luminance, positioning of the objects, etc, is a very complex task in practice. Thus, research is often based on artificial fog generation in rather controlled environments, usually large-scale fog chambers. However, such artificially generated fog is not fully equivalent to natural fog in terms of homogeneity and distribution.[7] This problem is especially sensitive with DNNs because they need huge datasets to achieve good results and avoid overfitting. Even though there exist defogging DNNs which are trained in an unpaired manner,[8] the problem still perseveres when it comes to validation, as the most used evaluation metrics require a ground truth for comparison.

This is why there has been interest in defining a metric that properly defines a quantitative enhancement in a defogging procedure. There are different approaches used for evaluation of defogging algorithms. We can divide the evaluation methods into three groups.[9] The first two are called full-reference image quality assessment (FR-IQA) and no-reference image quality assessment (NR-IQA). The first group, FR-IQA, needs a ground truth image to evaluate quantitatively the defogging result. This is the case of SSIM and PSNR. On the contrary, NR-IQA metrics either do not need a reference or do not use a fog-free ground truth image for comparison. Our proposed metric falls into this category. The third group simulates hazy images from clear images based on Koschmieder's law[10] and then employs FR-IQA metrics to evaluate dehazing algorithms.

Hautière *et. al.*[11] and Pomerleau *et. al.*[12] presented different NR-IQA methods to evaluate the attenuation coefficient of the atmosphere by means of a single camera on a moving vehicle. Nevertheless, their method cannot be used as metric for a general single image visibility evaluator, as Pormeleau *et. al.* needed multiple images of the scene and Hautière *et. al.* required a road and the sky to be present in the scene.

A different NR-IQA method was presented by Liu *et. al.*[13] and consisted on the analysis of the histogram of the image on the HSV colourspace. First, the image is converted from the RGB colourspace to HSV. Then, fog detection is achieved by analyzing different features of the histogram of each channel: Hue (H), Saturation (S), and Value (V). They stated that the overall value of the three channels decreased due to scattering resulting from the fog, so the distribution was modified in presence of fog. Feature extraction of each histogram was performed by adding the values of the pixels of the image and normalizing to the number of pixels different of 0 in the channel. After that, a classification into different visibility categories was done by comparing the results obtained from the histogram with some empirical values. Even though Liu *et. al.* claimed good results with this method there is certain subjectivity in the choice of values of the thresholds for the classification.

Li *et. al.*[14] compared the results of two FR-IQA (SSIM and PSNR) with two NQ-IQA methods (spatial-spectral entropy-based quality - SSEQ)[15] and blind image integrity notator using DCT statistics (BLIINDS-II)[16]). However, their results do not bring to a general conclusion about which IQA method has a better judgement. Besides, BLIINDS-II[16] is based on the statistical behavior of a group of 100 people, so there is inherent subjectivity in the metric. Another case that uses statistical behaviour of human judgement of foggy scenes is Liu's *et. al.*[17] Fog-relevant Feature based SIMilarity index (FRFSIM).

Also, Choi *et. al.*[18] presented a reference-less prediction of perceptual fog density and perceptual image defogging based on natural scene statistics and fog-aware statistical features. Their proposed model, Fog Aware Density Evaluator (FADE), predicts the visibility of a foggy scene from a single image without reference to a corresponding fog-free image and without being trained on human-rated judgments. FADE only makes use of measurable deviations from statistical regularities observed in natural foggy and fog-free images. Even though FADE performs well in general scenarios, the usage of statistical data could introduce an unwanted bias that could lead to a poor judgement of some scenarios. Apart from that, they present a single image defogging network called DEFADE. More recently, Chen *et. al.*[19] presented a visibility detection algorithm of a single fog image based on the ratio of wavelength residual energy. Nevertheless, their algorithm uses the transmisivity map, which is obtained by estimating certain atmospheric parameters.

Other approaches have been trying to fix the method using edge detection metric evaluation,[20] as we propose. However, they are mostly focused on the evaluation of the edge detection method rather in an improvement of the visibility of a scene by gradient comparison. Moreover, these metrics require a ground truth edge image for a proper evaluation.

Currently, the most used metric in defogging challenges is SSIM.[21] This well-known metric takes into account different aspects of an image and directly compares them with a sample image. SSIM basically focuses on contrast, luminance and structure. In fact, these are some of the most affected image features when fog is present in a scene. Nevertheless, defogging techniques, usually, do not try to completely recreate the original image but rather to produce an enhancement in the visibility of the fogged image by adjusting luminance, contrast, and other aspects of the scene. This could lead to a defogging procedure being heavily punished for not being similar enough to its ground truth even if the defogging results are good. Still, the main drawback of the metric for defogging evaluation is the need of a ground truth. As mentioned earlier, obtaining a ground truth image of a natural foggy scene is difficult, and the issue becomes more relevant when DNNs are introduced.

Hence, this work presents a state-of-the-art fast DNN architecture for SID of natural scenes capable of defogging images in real time. Besides, we also present a novel gradient-based metric for SID that needs neither a GT image of the scene nor an evaluation of the physical parameters of the image. The proposed metric has been compared with SSIM and other FR-IQA and NR-IQA metrics on the O-Haze[22] dataset with some results of the NTIRE 2018 defogging challenge.[3]

2. METHOD

2.1 Our Defogging GAN

2.1.1 Our Generator

The G gets data as input and generates "*fake*" data from it. The input dimensions do not need to be equal to the output dimensions. Nevertheless, the data that generates G needs to be of the same type and with the same dimensions as the "*real*" data we feed into the D. The hidden layers of the G may vary depending on the task. Our architecture is based on the U-Net[23] structure. To fully understand the U-Net structure we have to take a look at Auto-Encoders (AE). An AE is a type of ANN that is used to learn efficient codings out of unlabeled data i.e. can do unsupervised learning. This means that AEs can reduce (encode) the information in the input data and amplify it (decode) to retrieve the original data. Obviously, during the process, some information is lost and the output does not exactly match the input. Nevertheless, the important features of the input are usually preserved. An AE can be understood by the combination of an encoder and a decoder. The U-Net, then, maintains the AE structure and adds connections between the layers of the encoder and the decoder. These connections help to transmit the general structure of the input data preserving more information about it. This is a requirement in our application as we aim to defog an image without changing any details, or adding new features, into it.

Figure 3 shows the architecture of our G. One can identify three main stages in the network: the encoder, the middle stage, and the decoder. The encoder part gets as input a 128×128 colored image and reduces its dimensions with convolutional and max pooling layers. The middle stage consists of performing two convolutional layers. Finally, the decoder stage gets the latent vector from the middle stage and recovers the input image dimensions by performing transverse convolutional layers. Some very important aspects of our network are the concatenate and dropout layers. These operations help to maintain the details and general structure of the input image. GANs are often used to generate completely new images with generated details. We cannot afford to generate "*fake*" objects such as cars, traffic signals or people on the scene as our main purpose for defogging is to effectively increase the performance of CV algorithms that would ultimately execute an action in an autonomous vehicle. Executing an action due to a *ghost* object can be very dangerous, so a lot of effort has been put to make sure our network does not introduce *ghost* features.

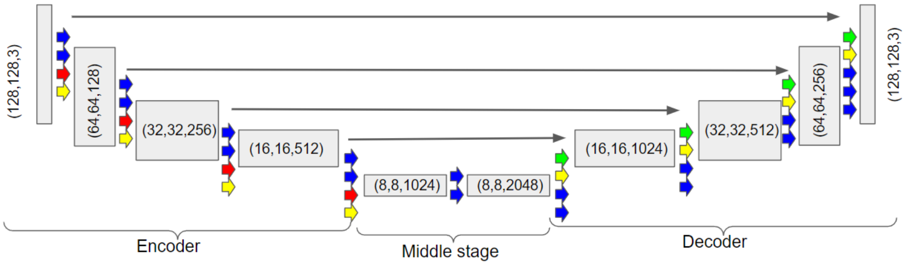

Figure 3. Generator architecture. The arrows represent the different layers applied: Convolutional (Blue), Max Pooling (Red), Dropout (Yellow), Convolutional Transpose (Green), and Concatenate (Black). Also, each convolutional layer had a ReLU activation function.

2.1.2 Our Discriminator

The D takes two sets of data, the "*real*" one and the "*fake*" generated by the U-Net. Randomly, we deliver these two sets of data to D. The input of D has to be properly labeled, i. e. indicating whether the image is "real" or "fake". The output layer of the D has to be a sigmoid (or equivalent) that transforms the previous data into a single scalar that represents the probability of the input data being "*real*" or "*fake*", between 0 and 1.

Figure 4 shows the architecture of our D. An image enters the network, its dimensions are reduced through convolutions until we get a flat vector in the latent space. After that, we apply one last dense layer with a sigmoid activation to retrieve a probability. We decided to add dropouts and Gaussian noise layers to the typical classifier structure. This may seem counterproductive as it lowers the detection capability of our network. However, one has to look at the bigger picture here and consider the network as a whole, including G. D's task, a binary classification, is way simpler than the defogging task performed by G. This led to D quickly understanding G's behavior and effectively identifying every image as "*real*" or "*fake*". Having a more advanced D led to a worse apprenticeship of G. By worsening D, we achieved a better joint learning process and better results.

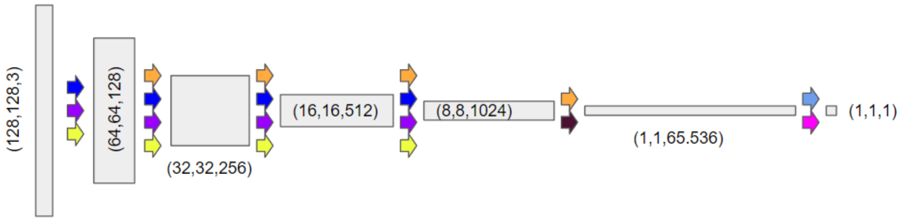

Figure 4. Discriminator architecture. The arrows represent the different layers applied: Convolutional (Blue), Leaky ReLU (Purple), Dropout (Yellow), Gaussian noise (Orange), Flatten (Brown), Dense (Cyan), and Sigmoid (Pink).

2.1.3 Our training loop

One of the most complex parts of adjusting the GAN is the training procedure. Both G and D learn from each other at each step. This symbiotic relationship only produces proper results when a fine equilibrium is achieved between the evolution rate of both networks. If G is significantly more advanced than D it might find patterns that fool D even if the results are not good. On the contrary, if D is far more advanced, G will not have sufficient information to produce good results and will output random noise. We also noticed that trying to defog an image directly from the initial weights is a too complex task for our network, which quickly saturated at low epochs. To avoid saturation, we thought of a more complex training loop. First, we trained our network to properly replicate fog-free images. This task is way easier than trying to defog the image from the beginning so our network completed the task with ease. This step was also useful for learning not to generate *ghost* objects. After that, we re-train our network from these weights to perform defogging.

A general training loop for a GAN can be seen in Algorithm 2. In our case, we took $k = 3$ for the reconstruct stage and $k = 1$ for the defogging stage as it led to the best results. The difference in training between the replicate stage and the defog stage only relies on the input data and its labeling. For the replicating stage, only fog-free images enter the G whereas both fog-free and generated images are taken as input for D. In D, fog-free images are labeled as *real* and generated images as *fake*. On the other hand, for the defogging stage, only fogged images enter G whereas fogged, fog-free, and generated images enter D. Only fog-free images are labeled as "*real*", the others, fogged and the generated defogged images, are labeled as "*fake*". The initial weights for the replicate stage are taken from a glorot uniform distribution. Both networks have an Adam optimizer with an initial learning rate of $7 \cdot 10^{-5}$. The G for the defogging stage has an initial learning rate an order of magnitude smaller for better refining of the images. The first learning stage takes us to a local minimum, a smaller learning rate for the second stage results in a fine tuning of the results, i. e. a realistic defogged image with the same features as the original.

Algorithm 1: General training loop for GANs.

```
for Number of training iterations do
    for k steps do
        ∘ Sample batch $m$ of real labeled data;
        ∘ Sample batch $m$ of fake labeled data;
        ∘ Update the Discriminator by ascending its stochastic gradient:
            $$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[ \log D\left(x^{(i)}\right) + \log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)\right] \quad (6)$$
    ∘ Sample batch $m$ of real or fake labeled data;
    ∘ Update the Generator by descending its stochastic gradient:
        $$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right) \quad (7)$$
```

2.2 Contour-based metric without ground truth

As Fig. 5 shows, the main effect that hazy weather has on a scene is decreased luminance and contrast, that dramatically reduce the contours and textures of the scene. Maintaining defined contours in adverse weather conditions is key to proper object recognition and segmentation, which are the basis of several applications. The visibility metric we present is based on gradient detection for image defogging evaluation. Our approach compares the gradient of the foggy image to the gradient of its defogged counterpart, i. e. after the defogging procedure is done. Hence, there is no need for a ground truth. Besides that, our method does not need to estimate any atmospheric parameter, which is difficult from a single RGB image and, in general, requires the sky to be present in the image.

Thus, as a first step, we need to obtain the derivative of both images (original and defogged), as can be seen in Fig. 5. There are several well-known image processing operators to perform such procedure. Some of the most used are Canny,[24] Roberts, Prewitt, and Sobel.[25] For our method, we used the Sobel edge detector[26] due to its simplicity. The horizontal and vertical derivatives are obtained by respectively applying the horizontal and vertical kernels on the image, as shown in Eq. 8,

$$F_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \circledast I \quad ; \quad F_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \circledast I, \tag{8}$$

where F_x and F_y are the corresponding horizontal and vertical derivatives of the image I. The image integrating all gradients is retrieved following

$$F = \sqrt{F_x^2 + F_y^2}. \tag{9}$$

Note that in any image, most of the pixels do not represent an edge, yielding small values in the processed gradient image. This can be appreciated in Fig. 5 where the white pixels that represent null or negligible gradients

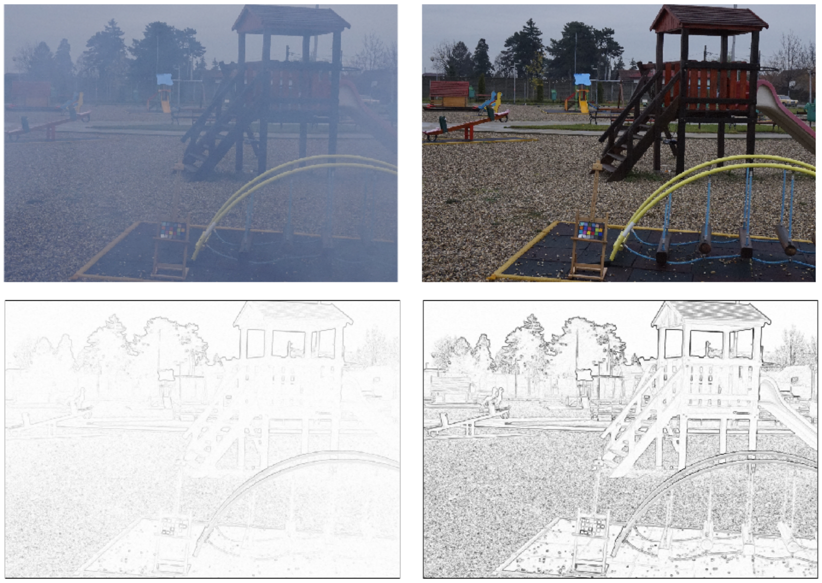

Figure 5. Gradient comparison between a fogged image (left) and its fog-free ground truth (right). Both colour images are presented on top with their associated edge images below.

are dominant in the image. Hence, we define a threshold value for the gradient values in order to differentiate the gradients of interest from the background (white).

Gradient images are normalized to one, so our proposed threshold value is to use 5% of the maximum edge value present in the image to still keep all the relevant information related to edges while disregarding the background data. This threshold will be further discussed after presenting Eq. 11. Such 5% value could be optimized for a different dataset if needed. After obtaining the derivative of each image, we perform the relative difference between the gradient images of the fogged and its defogged counterpart pixel by pixel, as stated in Eq. 10,

$$RD(x,y) = \begin{cases} \dfrac{d_e(x,y) - f_e(x,y)}{f_e(x,y)} & d_e(x,y), f_e(x,y) > \text{threshold} \\ 0 \quad \text{otherwise} & \end{cases}, \tag{10}$$

where $RD(x,y)$ is the relative difference computed at pixel (x,y), $d_e(x,y)$ is the defogged gradient image and $f_e(x,y)$ is the fogged gradient image. Let us take a moment to analyze the "*relative difference image*" so obtained. This image has the same dimensions as both input images. Each pixel stores the relative difference between the corresponding pixel of both input gradient images. If the value of a pixel in the relative difference image is positive, the strength of the gradients in the defogged image has improved, i. e. the gradient value in the defogged image is larger than the gradient value in the original image. Otherwise, if the value of a pixel in the relative difference image is negative, the strength of the gradient has decreased after the defogging algorithm. Therefore, the value of the difference quantifies the improvement in gradient strength obtained after the defogging process.

Once we compute the relative difference image, we perform the histogram of said image excluding the background pixels of the image, that is, the null values corresponding to those pixels below the threshold value. Fig. 6 presents the resulting histogram of the defogged image from the previous Fig. 9. The vast majority of edges in this image are better defined when fog is not present on the scene because of the defogging algorithm, as we would expect. Negative values close to 0 in the histogram correspond to some regions that have not been really affected by fog, or that in such areas the defogging process has introduced small variations in the gradient strength for these regions. These pixels, however, are quite residual compared to the rest. Note that positive pixels can reach values as large as 6, meaning a 6-fold improvement in the gradient strength.

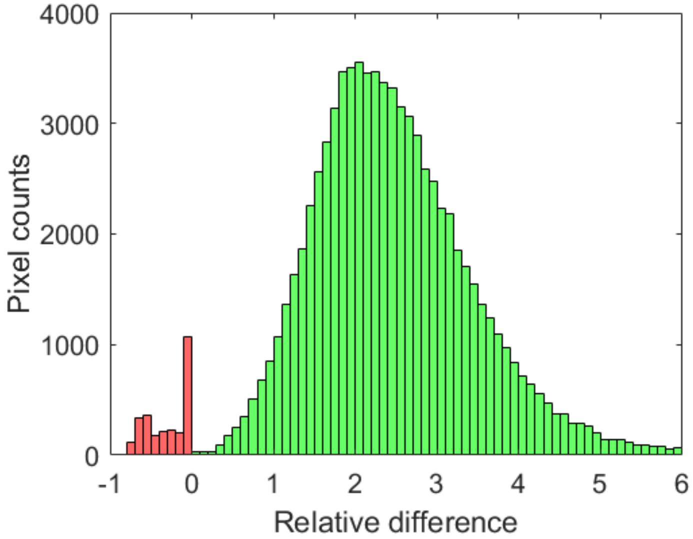

Figure 6. Histogram of the relative difference between the gradient images of the ground truth and the fogged image shown in Fig. 9. The positive values of the histogram are shown in green whereas negative values are shown in red.

At this point, the strategy of the gradient-based metric becomes clear. However, we still need a scalar value to quantify the enhancement of the defogging procedure consistent with the information that can be graphically observed in the histogram presented in Fig. 6. There are several options to obtain this numerical value. Our proposal consists on calculating the weighted ratio between the positive part of the histogram and the whole one. Mathematically,

$$R = \frac{\sum_{i=0}^{\infty} r_i^+ \cdot h(r_i^+) - \sum_{i=0}^{\infty} |r_i^-| \cdot h(r_i^-)}{\sum_{i=0}^{\infty} r_i^+ \cdot h(r_i^+) + \sum_{i=0}^{\infty} |r_i^-| \cdot h(r_i^-)} \tag{11}$$

where $r_i^{\pm}$ is the value of the relative difference, either positive or negative, and $h(r_i^{\pm})$ corresponds to the histogram value of $r_i^{\pm}$, so the total counts on the gradient image of such a value. R can take values from -1 to 1, being 1 when all the gradients have been enhanced and -1 when the defogging procedure has worsened every gradient of the image. The weighted character of the metric is used to strengthen the gradient that have been greatly improved or worsened. If we compute the proposed metric value for the example images shown in Fig. 9 we get $R = 0.9732$. This is a reasonable result since we are comparing a fogged image directly with its fog-free ground truth, mimicking an ideal defogging algorithm.

As we previously mentioned, the threshold value in Eq. 10 has been empirically fixed. We noted that setting a threshold above 8% disregards low intensity gradients, and decreases the value of the metric. On the other side, a threshold smaller than 5% considers a gradient almost any variation due to noise. We shall recall the reader that this threshold was introduced to separate the background pixels, where there are no contours, edges, or textures, from the low intensity edges.

An additional remark must be made. As previously discussed, DNNs, and especially GANs,[6] are nowadays used to tackle defogging. GANs are very useful when it comes to generating new data that resembles the data distribution it has learned from. This means that these networks tend to generate new features in the images, which leads to new contours producing better results in our metric even if the defogging is poor.

These situations may occur with images lacking edge information. Under this condition, two scenarios could happen. First, the original haze-free might does not have any contours. In this case, fog will not be a problem since no information would be hidden due to fog. Moreover, once the defogging procedure is done the resulting image will be very similar to the original hazy one because there is no element on the scene that needs to be improved. Second, the original haze-free scene has contours, but the fog is so dense that there is no visibility.

This is a more delicate case since there are elements in the image that could be improved. Nevertheless, no realistic defogging method could recover any information under such conditions. Any contour generated under extremely low visibility can in practice be considered a "ghost" object as long as it is appearing in the image from nothing.

In our opinion, generating these "*ghost*" features in the image should directly discard the defogging method. Defogging is especially useful to increase the performance of object detection and image segmentation, which will ultimately execute an action in an autonomous vehicle. Executing an action due to a "*ghost*" feature could be extremely dangerous. So our metric works under the premise that no new features are added to the defogged image during the defogging procedure, and only already existing features are highlighted.

In 2008, Hautiére *et. al.*[27] presented a reference-less metric that was based on a gradient comparison between the original hazy image and the defogged one. Specifically, it focuses on the new visible gradients that have appeared after the visibility enhancement. We hypothesize that any defogging method that generates new contours or gradients should be discarded. This decision is based purely on safety measurements as the authors believe that the main application of defogging algorithms is autonomous systems. Among other differences in the algorithm, our metric differs from Hautiére in the sense that it deals with the up-to-date problems of the defogging issue.

A complete algorithm for the metric computation is presented in Algorithm 2[*].

Algorithm 2: Algorithm to compute the gradient-based metric for image defogging without ground truth.

○ Obtain the gradient images of both Fogged and Defogged images:

$$F_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \circledast I \quad ; \quad F_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \circledast I$$

$$F = \sqrt{F_x^2 + F_y^2}$$

○ Obtain the *relative difference image* from the gradient images:

$$RD(x,y) = \begin{cases} \dfrac{d_e(x,y) - f_e(x,y)}{f_e(x,y)} & d_e(x,y), f_e(x,y) > \text{threshold} \\ 0 \quad \text{otherwise} \end{cases}$$

○ Perform the histogram of the *relative difference image*;

○ Obtain the ratio of enhancement:

$$R = \frac{\sum_{i=0}^{\infty} r_i^+ \cdot h(r_i^+) - \sum_{i=0}^{\infty} |r_i^-| \cdot h(r_i^-)}{\sum_{i=0}^{\infty} r_i^+ \cdot h(r_i^+) + \sum_{i=0}^{\infty} |r_i^-| \cdot h(r_i^-)}$$

3. EXPERIMENTS AND RESULTS

In this section we present a quantitative comparison between our proposed metric with several state-of-the-art methods to prove its effectiveness. Afer that, we will present the results[†] of our defogging network on the DENSE dataset.[28] We also present quantitative results of our network on our proposed metric and compare them with SSIM, the metric used in the official challenge, on the O-Haze dataset[22] using some results of the NTIRE 2018 challenge.[3]

[*] A Matlab implementation of the metric can be found at the following GitHub repository.

[†] Additional material can be found in the following google drive link, where the results are separated in folders with their corresponding *readme* files.

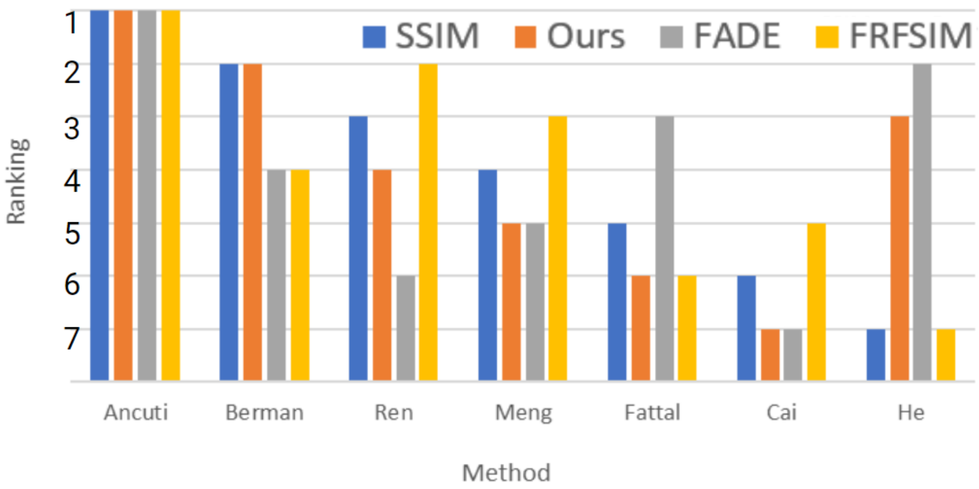

Figure 7. Classification of the mean over the 45 images of the O-Haze dataset for SSIM (FR-IQA), our proposed metric (NR-IQA), FADE (NR-IQA), and FRFSIM (FR-IQA).

3.1 Comparison of our proposed metric with the state-of-the-art

To validate our proposed metric, we tested it on the O-Haze dataset.[22] This dataset was used in the NTIRE 2018 challenge.[3] It consists of 45 outdoor scenes. Each fogged scene has its ground truth counterpart. Apart from that, the results of seven defogging methods provided by seven researchers were also facilitated in the dataset. Fig. 9 shows some examples of the O-Haze dataset as well as the seven mentioned results of the defogging methods. We used our metric to compare the results of some groups who participated in the challenge. During the NTIRE'18 defogging challenge, the groups received 35 fogged images, with their respective ground truth for training their networks. They also received 5 more images for validation purposes and 5 more for testing, which were evaluated by the jury. Again, the last ten images had their respective ground truths delivered. In order to fully validate the effectiveness of our metric, we used the 45 scenes with every defogging method available, reaching up to 405 images. Apart from that, we also tested two state-of-the-art defogging evaluation metrics, FRFSIM,[17] an FR-IQA metric based on statistical behavior over human judgment on foggy scenes, and FADE,[18] an NR-IQA fog density prediction model based on natural scene statistics, on the O-Haze dataset and compared the results with our own.

As mentioned above, the metrics used for evaluation in the NTIRE 2018 challenge were SSIM and PSNR, calculated relative to the ground truth image. The defogged images have 800 pixels of height or width at most whereas both the ground truth and the original hazy images have greater resolutions so we resized them to match the dimensions of the defogged image, to enable proper comparison. The resize method used was the bi-cubic algorithm. After resizing, we computed the value of the SSIM, FADE, FRFSIM, and our proposed metric for each scene and method. After that, we computed the mean over the 45 scenes to obtain a mean value of the defogging method for each criterion. Numerical values are shown in Table 1, where the worst and best values of each metric are plotted in red and green, respectively. The classification according to their ranking can be seen in Fig. 7.

Table 1 and Fig. 7 show relevant information. First, every metric considers Ancuti's as the best performing defogging method. There is a dispute over the last place. On the one hand, our metric and FADE, both NR-IQA,

Table 1. Mean over the 45 images of the O-Haze[22] dataset of SSIM (FR-IQA), our proposed metric (NR-IQA), FADE (NR-IQA with natural scene statistics) and FRFSIM (FR-IQA with human judgement). The best and worst performing results are colored in green and red respectively for each metric.

	He *et. al.*	Meng *et. al.*	Fattal *et. al.*	Bermann *et. al.*	Cai *et. al.*	Ren *et. al.*	Ancuti *et. al.*
SSIM↑	**0.399**	0.498	0.441	0.545	0.433	0.519	**0.573**
Ours↑	0.911	0.872	0.769	0.944	**0.747**	0.889	**0.975**
FADE↓	0.256	0.288	0.258	0.262	**0.642**	0.503	**0.252**
FRFSIM↑	**0.340**	0.461	0.352	0.443	0.352	0.468	**0.480**

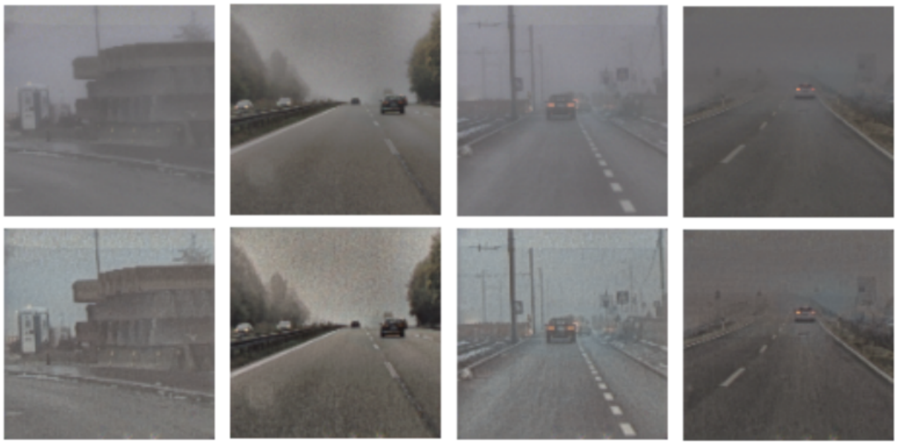

Figure 8. Defogging results of our network. The first row contains the original images and the second the results after the GAN is applied.

judge Cai's as the worst method. On the other hand, SSIM and FRFSSIM state that He is actually the worst defogging procedure. Let us look closely at He's case. When it comes to defogging and, especially, differentiating objects, He's results are visibly better than Meng's, Cai's, or even Ren's. Nevertheless, all previous groups are ahead of them when SSIM is applied. This can be explained by looking at the colors of each image and comparing them to the ground truth. The color aberration introduced by He is measured by SSIM and FRFSIM as a bad defogging method. On the contrary, our metric strictly considers one of the most affected features by fog, the edges of objects, leading to a more reasonable position of He's defogging method even without the need for a ground truth comparison.

As mentioned above, the metric used in the NTIRE'18 defogging challenge[3] was SSIM. From the metrics used in the paper, our proposed one is the one that better resembles SSIM's behavior. From SSIM's perspective, FADE and FRFSIM are too harsh on Berman and give too much credit to Fattal or Cai. Yet, in our case, the only discrepancy with SSIM is the He exception mentioned in the paragraph above.

3.2 Qualitative results from the defogging network

Finding a proper dataset is key to getting good results in a deep learning project, which usually needs a large set of images to see the network converge. Unfortunately, finding a large dataset of haze and its associated GT is a difficult task. For this project, we used some RGB images of the DENSE dataset.[28] The dataset used contained 3.366 haze-free images and 732 images with dense fog for training, 25 dense fog images for testing, and over 20 images of the same scene for validation in a fog chamber with its corresponding GT. This is still a very small dataset for a conventional deep learning project so we had to find an efficient training loop and constantly check the results with the test and validation datasets. Some defogging results can be seen in Fig. 8. We see how the GAN focuses on increasing the contrast and changing the illumination of the scene. Another interesting aspect of our network is its response time. Counting pre- and post-processing and inference time our network can work at a frame rate of over 45 fps, making it suitable for real-time applications. The calculations have been done with a Nvidia GeForce RTX 3070 and Python's TensorFlow environment.

3.3 Quantitative results on our proposed metric

In Fig. 9 a comparison of different defogging algorithms on the O-Haze dataset can be seen. Note that not every method presented in Fig. 9 is based on deep learning. Also, some of them estimate the physical parameters of the scene. Besides, these methods have been specially designed to perform well in this dataset, as it was part of the NTIRE 2018 challenge. Our model, instead, was trained on a different dataset which clearly differs from this one, especially in fog, as in the latter it is artificially created, so the distribution of fog in the scene is very different from what our network has learned. We compared our metric results with the metric used in the official challenge, SSIM, on every method available, including our defoggign results. Results are summarized in

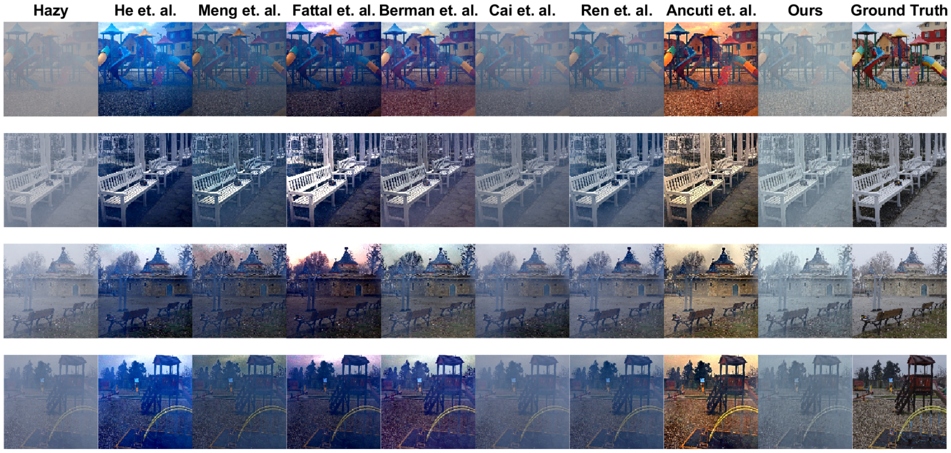

Figure 9. Comparison of different defogging algorithms on the O-Haze dataset. From left to right, the hazy scene, other proposed methods, in order, He, Meng, Fattal, Berman, Cai, Ren, Ancuti,[4,5,29–33] ours, and the ground truth.

Table 2. We can see that, first, our metric behaves similarly to the SSIM despite it works without any reference. The only clear difference between the two metrics is in He's case. He's method performs poorly on SSIM and average on our metric. If we look at Fig. 9 we can see that He's results are visibly better than Cai's or Meng's. However, He's drastically modified the image's colours resulting in a low SSIM, nevertheless, it increases the contours which leads to better performance on our metric. Second, even if our results in Fig. 9 do not look as good as Fig. 8 our method achieves good results on both metrics as it focuses on a better edge difference and contrast which leads to better identification of the objects. The difference in the results in Fig. 9 and Fig. 8 for our network can be explained through the structural difference between datasets. Our proposed metric allows to quantify the amount of defogging in a scene, without the need of a GT.

Table 2. Mean over the 45 images of the O-Haze[22] dataset of SSIM and our proposed metric.

	He *et. al.*	Meng *et. al.*	Fattal *et. al.*	Berman *et. al.*	Cai *et. al.*	Ren *et. al.*	Ancuti *et. al.*	Ours
SSIM	0.43684	0.51198	0.48097	0.57125	0.43986	0.53451	**0.60856**	0.53588
Ours	0.75652	0.63628	0.65295	0.80755	0.50801	0.71127	**0.91184**	0.90606

4. CONCLUSIONS AND FUTURE WORK

We proposed a fast defogging network with a GAN-like architecture that only takes the information of an RGB image of the fogged scene and does not estimate any physical parameter of the scene. We thoroughly described each part, its training loop, and our stretch to preserve the general structure of the input images to avoid the generation of *ghost* objects. We also described a gradient-based metric for SID that does not need a GT image. Lastly, we compared our proposed metric with the current metric of evaluation for defogging challenges, SSIM, as well as other state-of-the-art metrics through the O-Haze dataset. We concluded that the proposed metric properly judges visual enhancement of SID without any reference other than the original RGB fogged scene. In future work, we intend to train defogging networks with a polarized dataset in natural hazy scenes, despite the difficulties of producing such data, as polarized light is less affected by fog than unpolarized light.

ACKNOWLEDGMENTS

This work has been possible thanks to the project PID2020-119484RB-I00 and TED2021-132338B-I00 funded by Ministerio de Ciencia e Innovación de España. This work is part of a Ph.D. thesis with grant number 2023 FI-1 00229 co-funded by the European Union.

REFERENCES

[1] Zhang, W., Liang, J., Wang, G., Zhang, H., and Fu, S., "Review of passive polarimetric dehazing methods," *Optical Engineering* **60** (2021).

[2] Schechner, Y. Y., Narasimhan, S. G., and Nayar, S. K., "Polarization-based vision through haze," *Applied Optics* **42** (2003).

[3] Ancuti, C., Ancuti, C. O., Timofte, R., Van Gool, L., Zhang, L., Yang, M.-H., Patel, V. M., Zhang, H., Sindagi, V. A., Zhao, R., Ma, X., Qin, Y., Jia, L., Friedel, K., Ki, S., Sim, H., Choi, J.-S., Kim, S., Seo, S., Kim, S., Kim, M., Mondal, R., Santra, S., Chanda, B., Liu, J., Mei, K., Li, J., Luyao, Fang, F., Jiang, A., Qu, X., Liu, T., Wang, P., Sun, B., Deng, J., Zhao, Y., Hong, M., Huang, J., Chen, Y., Chen, E., Yu, X., Wu, T., Genc, A., Engin, D., Ekenel, H. K., Liu, W., Tong, T., Li, G., Gao, Q., Li, Z., Tang, D., Chen, Y., Huo, Z., Alvarez-Gila, A., Galdran, A., Bria, A., Vazquez-Corral, J., Bertalmo, M., Demir, H. S., Adil, O. F., Phung, H. X., Jin, X., Chen, J., Shan, C., and Chen, Z., "Ntire 2018 challenge on image dehazing: Methods and results," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*], 891–901 (2018).

[4] He, K., Sun, J., and Tang, X., "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33** (2011).

[5] Cai, B., Xu, X., Jia, K., Qing, C., and Tao, D., "DehazeNet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing* **25**, 5187–5198 (nov 2016).

[6] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative adversarial nets," (2014).

[7] Duthon, P., Colomb, M., and Bernardin, F., "Fog classification by their droplet size distributions: Application to the characterization of cerema's platform," *Atmosphere* **11** (2020).

[8] Engin, D., Genc, A., and Ekenel, H. K., "Cycle-dehaze: Enhanced cyclegan for single image dehazing," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* **2018-June** (2018).

[9] Zhao, S., Zhang, L., Huang, S., Shen, Y., and Zhao, S., "Dehazing evaluation: Real-world benchmark datasets, criteria, and baselines," *IEEE Transactions on Image Processing* **29** (2020).

[10] Middleton, W. E. K. and Twersky, V., "Vision through the atmosphere," *Physics Today* **7** (1954).

[11] Hautiére, N., Tarel, J. P., Lavenant, J., and Aubert, D., "Automatic fog detection and estimation of visibility distance through use of an onboard camera," *Machine Vision and Applications* **17** (2006).

[12] Pomerleau, D., "Visibility estimation from a moving vehicle using the ralph vision system," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC* (1997).

[13] Liu, C., Lu, X., Ji, S., and Geng, W., "A fog level detection method based on image hsv color histogram," *PIC 2014 - Proceedings of 2014 IEEE International Conference on Progress in Informatics and Computing* (2014).

[14] Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., and Wang, Z., "Benchmarking single-image dehazing and beyond," *IEEE Transactions on Image Processing* **28** (2019).

[15] Liu, L., Liu, B., Huang, H., and Bovik, A. C., "No-reference image quality assessment based on spatial and spectral entropies," *Signal Processing: Image Communication* **29** (2014).

[16] Saad, M. A., Bovik, A. C., and Charrier, C., "Blind image quality assessment: A natural scene statistics approach in the dct domain," *IEEE Transactions on Image Processing* **21** (2012).

[17] Liu, W., Zhou, F., Lu, T., Duan, J., and Qiu, G., "Image defogging quality assessment: Real-world database and method," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* **30** (2021).

[18] Choi, L. K., You, J., and Bovik, A. C., "Referenceless prediction of perceptual fog density and perceptual image defogging," *IEEE Transactions on Image Processing* **24** (2015).

[19] Chen, Z. and Ou, B., "Visibility detection algorithm of single fog image based on the ratio of wavelength residual energy," *Mathematical Problems in Engineering* **2021** (2021).

[20] Magnier, B., Abdulrahman, H., and Montesinos, P., "A review of supervised edge detection evaluation methods and an objective comparison of filtering gradient computations using hysteresis thresholds," *Journal of Imaging* **4** (2018).

[21] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P., "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing* **13** (2004).

[22] Ancuti, C. O., Ancuti, C., Timofte, R., and De Vleeschouwer, C., "O-haze: a dehazing benchmark with real hazy and haze-free outdoor images," in [*IEEE Conference on Computer Vision and Pattern Recognition, NTIRE Workshop*], *NTIRE CVPR'18* (2018).

[23] Ronneberger, O., Fischer, P., and Brox, T., "U-net: Convolutional networks for biomedical image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **9351** (2015).

[24] Canny, J., "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-8** (1986).

[25] Chaple, G. N., Daruwala, R. D., and Gofane, M. S., "Comparisions of robert, prewitt, sobel operator based edge detection methods for real time uses on fpga," *Proceedings - International Conference on Technologies for Sustainable Development, ICTSD 2015* (2015).

[26] Vincent, O. and Folorunso, O., "A descriptive algorithm for sobel image edge detection," *Proceedings of the 2009 InSITE Conference* (2009).

[27] Hautière, N., Tarel, J.-P., Didier, A., and Dumont, E., "Blind contrast enhancement assessment by gradient ratioing at visible edges," *Image Analysis and Stereology* **27** (06 2008).

[28] Bijelic, M., Mannan, F., Gruber, T., Ritter, W., Dietmayer, K., and Heide, F., "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2020* (2020).

[29] Meng, G., Wang, Y., Duan, J., Xiang, S., and Pan, C., "Efficient image dehazing with boundary constraint and contextual regularization," *Proceedings of the IEEE International Conference on Computer Vision* (2013).

[30] Fattal, R., "Dehazing using color-lines," *ACM Transactions on Graphics* **34** (2014).

[31] Berman, D., Treibitz, T., and Avidan, S., "Non-local image dehazing," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2016-December** (2016).

[32] Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., and Yang, M.-H., "Single image dehazing via multi-scale convolutional neural networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **9906 LNCS** (2016).

[33] Ancuti, C. O. and Ancuti, C., "Single image dehazing by multi-scale fusion," *IEEE Transactions on Image Processing* **22** (2013).