Towards Artifacts-free Image Defogging

Towards Artifacts-free Image Defogging

Gabriele Graffieti, Davide Maltoni
Department of Computer Science and
Engineering (DISI) – University of Bologna
Via dell'Università, 50, 47521 Cesena (FC), Italy
Emails: gabriele.graffieti@unibo.it, davide.maltoni@unibo.it

***Abstract*—In this paper we present a novel defogging technique, named *CurL-Defog*, aimed at minimizing the creation of unwanted artifacts during the defogging process. The majority of learning based defogging approaches rely on paired data (i.e., the same images with and without fog), where fog is artificially added to clear images: this often provides good results on mildly fogged images but does not generalize well to real difficult cases. On the other hand, the models trained with real unpaired data (e.g. CycleGAN) can provide visually impressive results but they often produce unwanted artifacts. In this paper we propose a curriculum learning strategy coupled with an enhanced CycleGAN model in order to reduce the number of produced artifacts, while maintaining state-of-the-art performance in terms of contrast enhancement and image reconstruction. We also introduce a new metric, called *HArD* (Hazy Artifact Detector) to numerically quantify the amount of artifacts in the defogged images, thus avoiding the tedious and subjective manual inspection of the results. The proposed approach compares favorably with state-of-the-art techniques on both real and synthetic datasets.**

I. Introduction

Images captured under bad weather conditions such as fog, mist or haze, suffer from limited visibility, poor contrast, faded colors and loss of sharpness. This not only makes the images aesthetically less pleasant, but can seriously deteriorate the accuracy of computer vision applications such as autonomous driving, which heavily relies on object detection, tracking, classification, and segmentation [19]. Defogging (or dehazing) is the task of removing the fog from a given image, aimed at reconstructing the same scene as if it were taken in good (or at least better) weather conditions.

In recent years, a plethora of defogging approaches have been proposed, based on both classical neural networks [5, 23, 16] and generative adversarial networks [22, 18, 9, 20]. Though these methods often outperform the preexisting state-of-the-art, they usually must be trained with paired data, namely the same scene acquired with and without fog. The impossibility to acquire perfectly paired data, alongside the increased need for big datasets, pushed the researchers to create and use synthetic data, where the fog is automatically inserted according to a physical model (see Equation 1) [17, 26, 8]. Moreover, the majority of the existing defogging technique cannot handle images with severe fog, where the visibility of the scene is highly compromised [1], limiting their practical application in real scenarios. When totally unpaired models (e.g., CycleGAN) are applied to images with severe fog, the results are visually pleasant but often affected by the presence of unwanted artifacts (see Figure 1), added by the "imagination" of the model. Such artifacts can be extremely dangerous in some applications: for example, non-existent pedestrians or obstacles added to a defogged road scene could lead an autonomous car to take wrong decisions.

(a) (b) (c)

Fig. 1: (a) real severe fogged images (b) defogging with a state-of-the-art unpaired approach [9] that exhibits clear artifacts in the sky (c) the results produced by the proposed CurL-Defog denote a good compromise between image enhancement and fidelity to the original scene. Better viewed if zoomed on a computer monitor.

In this paper we present a novel defogging technique, based on recent developments in generative adversarial networks and style transfer. To reduce the insertion of artifacts, we exploit the concept of *curriculum learning* [3], where a model is trained firstly on simpler tasks, and then the complexity of the examples is increased as the training progresses. We first train our model on artificial paired data, comparing the defogged image with the ground truth real scene. This highly penalize the insertion of artifacts, thus introducing a bias which is not destroyed in successive training phases when the model is exposed to real unpaired data to learn more difficult transformations.

As shown in section V, our model is able to perform defogging more effectively on real foggy images than the majority of model trained only on artificial data. In addition, the presence of artifacts is significantly reduced with respect to the

CycleGAN-based approaches.

In order to numerically quantify the amount of artifacts introduced by a defogging technique we also propose (in section IV) a novel referenceless metrics denoted as *HArD*.

II. Related Works

A. Single Image Defogging (or Dehazing)

In presence of fog or haze particles, the original irradiance received by the camera gets attenuated, proportionally to the distance of the objects. This effect is combined with the scattering of atmospheric light. A simple mathematical model can be formulated as [14, 21]:

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

where $I(x)$ and $J(x)$ are the hazy and clear images respectively, A is the global atmospheric light and $t(x)$ is the transmission map of the scene, defined as $t(x) = e^{-\beta d(x)}$, where $d(x)$ is the depth map of the scene and β the scattering coefficient of the atmosphere.

Some earlier methods directly tried to invert the aforementioned physical model, estimating the parameters directly from examples or exploiting some statistical properties of images [11, 31, 4]. In recent years, classical approaches have been surpassed by deep-learning techniques, often based on convolutional neural networks (CNNs). Some learning methods estimate the transmission map $t(x)$ and then invert the physical model to reconstruct the clear image [23, 5], others directly produce the clear image [16].

The application of generative adversarial networks (GANs) to dehazing is even more recent [22]. The use of GANs as end-to-end models that directly produce the haze-free image without estimating the transmission map was developed by Li *et al.* [18]; in this model the discriminator receives a pair of images, the hazy image and the clear one, and its goal is to estimate the probability that the given pair is the ground truth, i.e. the haze-free image is the real one given the foggy picture, and it was not produced by the generator.

All the approaches reported above are trained on paired datasets of foggy and clear images. Unfortunately, these datasets are often synthetic and the quality of results may decrease if the resulting model is tested with real hazy photographs. Defogging with unpaired data was explored by Engin *et al.* [9], where an inception loss computed by features extracted from a VGG network [25] was used as a regularizer. In [20], a model similar to CycleGAN is used in conjunction with the physical model to apply fog to real images during the inverse mapping.

B. Defogging Metrics

Assessing the quality of defogging is a particularly difficult task, especially without information about the depth map of the scene. At today, human evaluation is often preferred w.r.t. automatic evaluation to assess the quality of defogging. However, human evaluations are subjective, tedious and impractical for large amount of data.

The most common metrics used to evaluate image defogging are *structural similarity (SSIM)* [29] and *peak signal-to-noise ratio (PSNR)*), both well-known in image processing (e.g. to estimate the quality of compression or deblurring techniques). The main problem with these metrics is the need of a reference clear image that, as stated before, is a nearly impossible to obtain in a real world scenario. In addition, SSIM and PSNR often do not correlate well with human judgment or with referenceless metrics [17].

Recently, some referenceless metrics specifically designed for defogging have been proposed [10, 7]. One of these "blind" metrics was introduced by Hautière *et al.* [10]; it consists of three different indicators: e, σ and $\bar{r}$. The value of e is based on the number of visible edges in the defogged image relative to the original foggy picture. The value of σ represents the percentage of pixels that become saturated (black or white) after defogging. Finally, $\bar{r}$ denotes the geometric mean of the ratio of the gradient at visible edges; in short it gives an indication of the amelioration of the contrast in the defogged image: high values of $\bar{r}$ mean a more effective defogging. Unfortunately, as discussed in section IV, some of these metrics are negatively affected by the presence of artifacts.

III. The CurL-Defog Model

In order to train a model with real images and, at the same time, reducing the artifacts inserted during the defogging, we propose *CurL-Defog*, an approach inspired by curriculum learning [3], where the model is first guided towards a desirable parameters-space region via a pix2pix-like [13] training using synthetic paired image. The model is then refined by a more complex CycleGAN-like training, where real unpaired images are progressively provided.

A. The model

The two training phases of the proposed model are not separated, but there is a gradual transition from the paired training to the unpaired one. Indeed, at each epoch the model is trained with some synthetic images and some real images. As the epochs goes on, the number of artificial images is reduced while the number of real examples grows, thus progressively increasing the influence of unpaired training.

During the paired training (where the model is trained with artificial data) we use two different pix2pix models, one for defogging fogged images and one for fogging clear images; these two submodels are trained separately in this phase. Conversely, during the unpaired training, the model can be seen as a CycleGAN-based model, where both the fogging and defogging networks are used in combination to enforce the cycle consistency loss.

Overall, CurL-Defog is composed of four networks: a generator G_{defog} used for defogging, a generator G_{fog} used for

fogging, a discriminator D_{clear} used to discriminate between real and defogged images, and a discriminator D_{fog} used to discriminate between real and generated foggy images. The overall approach is graphically shown in Figure 2, and the training algorithm is summarized in Algorithm 1.

Algorithm 1 CurL-Defog training algorithm. More details are provided in subsection III-C. The number of artificial and real images used in every epoch (lines 2-3) is discussed in section V

1: **for** number of epochs **do**
2: $k =$ nr. of artificial images in the current epoch
3: $j =$ nr. of real images in the current epoch
4: **for** k **do** ▷ paired training
5: Draw a pair (c, f) from the paired dataset.
6: $\hat{c} \leftarrow G_{defog}(f)$
7: $\hat{f} \leftarrow G_{fog}(c)$
8: Calculate $\mathcal{L}_{pair}$ using Equation 2.
9: BACK-PROPAGATE $\mathcal{L}_{pair}$
10: UPDATE $G_{defog}, G_{fog}, D_{clear}$ and D_{fog}.
11: **end for**
12: **for** j **do** ▷ unpaired training
13: Draw c from the real clear dataset.
14: Draw f from the real foggy dataset.
15: $\hat{c} \leftarrow G_{defog}(f)$
16: $\hat{f} \leftarrow G_{fog}(c)$
17: $f_{rec} \leftarrow G_{fog}(\hat{c})$
18: $c_{rec} \leftarrow G_{defog}(\hat{f})$
19: Calculate $\mathcal{L}_{unpair}$ using Equation 6.
20: BACK-PROPAGATE $\mathcal{L}_{unpair}$
21: UPDATE $G_{defog}, G_{fog}, D_{clear}$ and D_{fog}.
22: **end for**
23: **end for**

B. Networks

1) Generator networks: The architecture of the generator is the same for both G_{defog} and G_{fog}. The networks contain three convolutional blocks in the encoding module and three convolutional blocks in the decoding module. Every convolutional block is followed by an instance normalization [27] layer and a ReLU activation layer. In the encoder, the fist layer has 64 filters with a kernel size of 7×7 and stride 1. The second and third layers have 128 and 256 filters respectively, with a kernel size of 3×3 and stride 2. Each layer of the decoder module has the same number of filters of its symmetric layer in the encoder module, but the convolutions have stride 1/2 (transposed convolution). To perform the translation, we have used 9 residual blocks [12], similar to the original CycleGAN implementation.

2) Discriminator networks: Both discriminators D_{clear} and D_{fog} are based on the same architecture: a 70×70 *patchGAN* [15] aimed at classifying 70×70 overlapping image patches as real or fake. Such model contains fewer parameters than a classical image classification network, and can operate on images of any size, due to its fully convolutional architecture.

C. Full objective

1) Paired training: During paired training the losses of the model are similar to the original pix2pix implementation [13]. The adversarial loss for defogging is joined with the L1 loss, calculated between the translated and the ground truth images as follow:

$$\mathcal{L}_{pair} = \mathcal{L}_{adv}^{defog} + \mathcal{L}_{adv}^{fog} + \lambda_{L1}\mathcal{L}_{L1} \tag{2}$$

where λ_{L1} controls the relative importance of the L1 loss.

2) Unpaired training: During unpaired training the model is trained with unpaired foggy and clear images, so the L1 loss cannot be used. Instead, we use the cycle consistency loss derived from the CycleGAN model [30]:

$$\begin{aligned}\mathcal{L}_{cyc} = &\|G_{fog}(G_{defog}(f)) - f\|_1 \\ &+ \|G_{defog}(G_{fog}(c)) - c\|_1\end{aligned} \tag{3}$$

where f and c represents the foggy and clear images respectively.

In order to make the defogging more effective and preserve details in the results, a cycle perceptual loss based on feature extracted with a pretrained VGG-16 network [25] is introduced, following the work of Engin *et al*. [9]. The cycle perceptual loss is defined as:

$$\begin{aligned}\mathcal{L}_{perc} = &\|\phi(f) - \phi(G_{fog}(G_{defog}(f)))\|_2^2 \\ &+ \|\phi(c) - \phi(G_{defog}(G_{fog}(c)))\|_2^2\end{aligned} \tag{4}$$

where $\phi(\cdot)$ represents features extracted from the 2nd and the 5th pooling layers of the VGG-16 network.

Furthermore, similar to the original CycleGAN implementation [30], we also include an identity loss, aimed to preserve the original tint and color of the images. The identity loss is defined as:

$$\mathcal{L}_{idt} = \|G_{fog}(f) - f\|_1 + \|G_{defog}(c) - c\|_1 \tag{5}$$

Finally, the overall objective of the model during the unpaired training can be expressed as:

$$\begin{aligned}\mathcal{L}_{unpair} = &\mathcal{L}_{adv}^{defog} + \mathcal{L}_{adv}^{fog} + \lambda_{cyc}\mathcal{L}_{cyc} \\ &+ \lambda_{perc}\mathcal{L}_{perc} + \lambda_{idt}\mathcal{L}_{idt}\end{aligned} \tag{6}$$

where $\lambda_{cyc}, \lambda_{perc}$ and λ_{idt} control the relative importance of cycle consistency loss, perceptual loss and identity loss respectively.

IV. THE HARD METRIC

As discussed in section II, some ad-hoc metrics have been proposed in the literature for assessing the quality of defogging techniques, each focusing on a different aspect, such as the level of fog [7] or the amelioration of contrast [10].

However, the aforementioned metrics do not take into account the presence of artifacts in the defogged image, which is a critical issue especially when unpaired image-to-image translation models are used. For example, the metrics proposed in

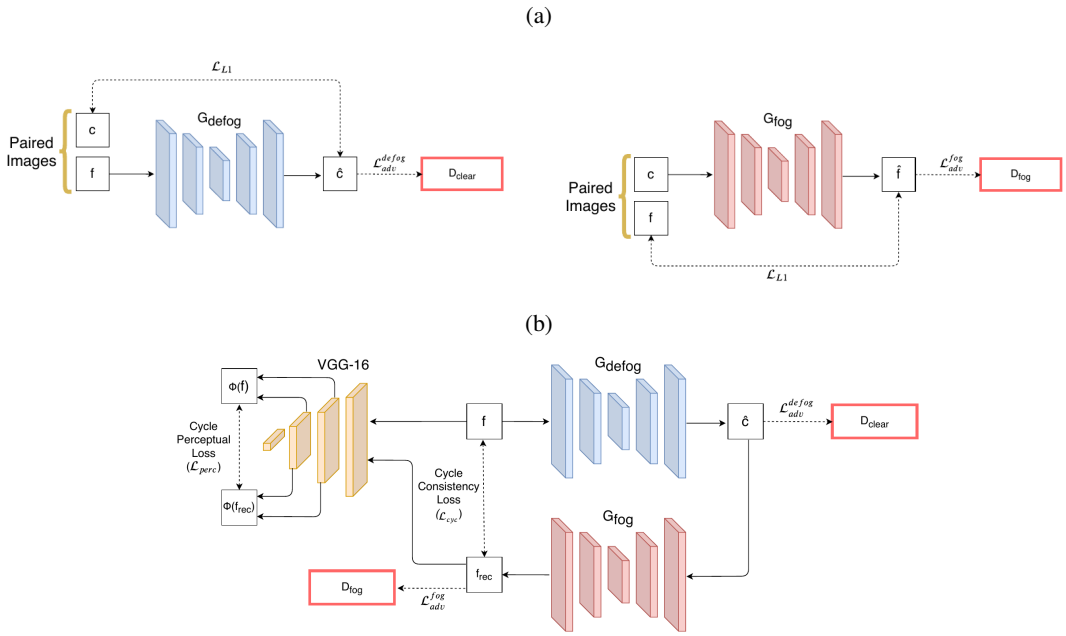


Fig. 2: CurL-Defog architecture. (a) During the paired training: two separate pix2pix models are used, one for defogging and one for fogging images. (b) During the unpaired training the two generators are used in combination to enforce the cycle consistency loss and the cycle perceptual loss; to keep the notation simple in (b) only the defogging cycle is displayed, since the foggy cycle (starting from the clear image) is symmetrical and can be easily derived. For each training epoch the model behaves as (a), for a subset of the training iterations, and as (b) for the rest of the iterations.

[10] can be deceived by the presence of artifacts (especially the descriptor e), since the addition of nonexistent objects can raise the number of visible edges in the defogged image.

Here we propose *HArD (Haze Artifact Detector)* a new referenceless metrics to quantify the amount of artifacts in a defogged image, given the corresponding foggy scene. HArD is based on the assumption that if a region in the original foggy image does not present any edge (i.e. is a region of constant intensity), the defogging method should not introduce any object in that region, since there is no information to exploit for reconstructing the scene. The pseudocode of the HArD metric is illustrated in Algorithm 2.

HArD can be implemented through a sequence of image processing steps (see Algorithm 2): first, a map of edges concentration in the foggy and defogged images are computed by determining the module of the gradient (Prewitt operator) and smoothing it by convolution with a Gaussian. The maps are then normalized in the range [0,1] using statistics derived from the entire training dataset. After that, both the maps are saturated multiplying them by a constant and applying the hyperbolic tangent function: this step is aimed at making the edge regions in the two images comparable even if the defogged image is usually much more contrasted. Finally, the artifact regions in the defogged images are determined by subtracting the saturated foggy edge density map from the defogged edge density map. The obtained difference map has high values in regions denoted by the presence of edges in the defogged image but not in the fogged one. An example is shown in Figure 3.

Algorithm 2 HArD metric calculation.

Require: f = original foggy image in range [0,1]
Require: d = automatically defogged image in range [0,1]

procedure HARD(f, d)
  $f' \leftarrow \text{PREWITT}(f_{gray})$
  $d' \leftarrow \text{PREWITT}(d_{gray})$
  $f'_{smooth} \leftarrow \text{GAUSSIANFILTER}(f')$ ▷ $\sigma = 20$
  $d'_{smooth} \leftarrow \text{GAUSSIANFILTER}(d')$ ▷ $\sigma = 20$
  $f'_{scaled} \leftarrow \text{NORMALIZE}(f'_{smooth})$
  $f'_{sat} \leftarrow \tanh(\nu_{fog} \cdot f'_{scaled})$ ▷ $\nu_{fog} = 7.5$
  $d'_{scaled} \leftarrow \text{NORMALIZE}(d'_{smooth})$
  $d'_{sat} \leftarrow \tanh(\nu_{defog} \cdot d'_{scaled})$ ▷ $\nu_{defog} = 1.5$
  $diff \leftarrow \min(0, d'_{scaled} - f'_{sat})$
  return $\text{MEAN}(diff)$
end procedure

V. Experiments and Results

In this section we evaluate the proposed CurL-Defog approach and compare it with some state-of-the-art methods. We report experiments both on artificial and real foggy images, with a particular emphasis on scenes with severe fog and reduced visibility.

TABLE I: Average SSIM and PSNR scores of the dehazing on the HSTS testing set.

	DCP [11]	CAP [31]	NLD [4]	DehazeNet [5]	MSCNN [23]	AOD-Net [16]	Pix2Pix [13]	**CurL-Defog**
PSNR	14.84	21.53	18.92	24.48	18.64	20.55	24.22	**24.83**
SSIM	0.7609	0.8727	0.7411	**0.9183**	0.8168	0.8973	0.8991	0.9037

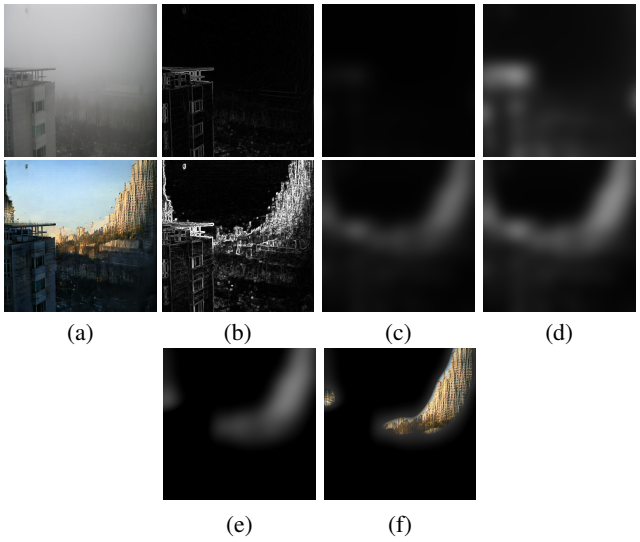


Fig. 3: Overall sequence of steps in HArD metrics calculation. (a) input images: top row is the real fogged image, second row is the same image defogged by a pure unpaired approach [30]. (b) gradient computation by the Prewitt operator. (c) smoothing with a Gaussian filter and scaling in the interval [0,1]. (d) saturation of the maps through the hyperbolic tangent. (e) difference between the two maps in (d). (f) the regions where the metrics detected the presence of artifacts superimposed over the defogged image. Better viewed if zoomed on a computer monitor.

A. Training details

During training all the images were scaled to 286×286 pixel, using a bicubic interpolation, and then a random crop of size 256×256 was taken and used as input to the networks. To reduce oscillation when the model was trained in the unpaired training phase, we followed the strategy proposed by Shrivastava *et al*. [24] where the discriminators are updated by using a history of generated images instead of the last produced by the generators. We kept an image pool of the last 50 generated images and randomly chose one of them to train the discriminators. For all the experiments we maintained the same parameters:

$$\lambda_{L1} = 100 \qquad \lambda_{cyc} = 10$$
$$\lambda_{perc} = 0.1 \qquad \lambda_{idt} = 0.5 \cdot \lambda_{cyc}$$

These parameter values have been manually selected, without performing any systematic grid search: so we believe that accuracy can be further improved. In all the experiments the model was trained for 200 epochs. The learning rate was initially set to 0.0002, kept unaltered for the first 100 epochs and then linearly decayed to zero in the last 100 epochs.

Curriculum Learning Strategy

The definition of the curriculum learning strategy is not explicitly specified in CurL-Defog, so many approaches could be used. The simplest strategy, denoted as linear, works as follows: at the first epoch the model is trained only with artificial data; as the epochs progresses, the number of artificial images is linearly decreased, reaching zero at the last epoch; conversely, the number of real images is linearly increased, reaching the maximum in the last epoch.

We experimented other possible strategies, in particular other two approaches denominated linear-saturate and step. In the step strategy only artificial images are used for the first half of the epochs, and only real images for the remaining half. The linear saturate strategy is similar to the linear strategy, but the model is trained only with real data for the last k epochs. These two alternatives did not show any advantage over the simple linear strategy, so the linear strategy was used for all the subsequent experiments.

B. Experiments on synthetic data

To assess the performance of CurL-Defog on synthetic data we have used the classical SSIM and PSNR metrics. In fact, even if improved metrics have been introduced (such as MS-SSIM [28]), SSIM and PSNR have been used by most of the methods we considered for comparison. In this experiment we compare our approach with both classical methods [11, 31, 4] and machine learning techniques [5, 23, 16]. We also include the pix2pix model [13] in our experiments. The datasets used are:

- Synthetic paired dataset (training): *Outdoor Training Set (OTS)*, included in the RESIDE dataset [17]. The dataset is composed of 2,061 clear images where, for each image, 35 different levels of haze are applied (varying the parameters A and β of Equation 1), for a total of 72,135 training images.
- Real unpaired dataset (training): *LIVE image defogging* [6], composed of 500 clear and 500 foggy real photographs. The major issues with LIVE dataset is the low number of images, and the marked difference between foggy and clear photographs. Hence, we substitute the 500 clear images with 2.651 clear photographs taken from the RESIDE dataset [17] (with no intersection between the images used for the synthetic dataset experiments). The clear images were manually selected in order to include only daytime photographs, with clear skies and good lighting.

- Test dataset: *Hybrid Subjective Testing Set (HSTS)*, included in RESIDE dataset and used as a test dataset in the recent benchmark by Li *et al*. [17].

The results are reported in Table I. It is worth noting that our approach is in line with state-of-the-art models, reaching the first score for PSNR and the second for SSIM, even if Curl-Defog was not designed to rival existing state-of-the-art approaches on synthetic data.

C. Experiments on real images

Real data is intrinsically unpaired, so SSIM and PSNR cannot be used to assess the effectiveness of defogging. Therefore, the metric proposed by Hautière *et al*. [10] was used to evaluate the results, alongside the proposed HArD metric to detect the presence of artifacts. CurL-Defog is here compared against the Cycle-Dehaze unsupervised approach [9] and a pix2pix model [13]. The datasets used in this experiment are:

- Synthetic paired dataset (training): *Outdoor Training Set (OTS)*, included in the RESIDE dataset [17].
- Real unpaired dataset (training): *LIVE image defogging* [6], enhanced with more images as described in the experiment on synthetic data.
- Test dataset: test set of the LIVE image defogging dataset (100 images) [6].

The results are reported in Table II and some examples are shown in Figure 4. As we can see from Figure 4, our defogging method often produces more realistic results compared to Cycle-Dehaze [9] with almost no artifacts in the defogged image, even if the contrast and color saturation are lower. At the same time the amount of details, especially on very foggy regions, is better than pix2pix. The HArD values in Table II support these observations. Regarding the Hautière *et al*. metrics e and $\bar{r}$, these value are affected by the insertion of artifacts, which produce edges that contribute to increase them: therefore, they cannot be considered in isolation.

TABLE II: Indicators e and $\bar{r}$ from [10] and HArD metric calculated on the LIVE test set. ↑ indicates that a higher value is better, ↓ indicates that a lower value is better.

	CycleDehaze [9]	Pix2Pix [13]	**CurL-Defog**
$\boldsymbol{e}$ (↑)	32.70	25.74	28.41
$\boldsymbol{\bar{r}}$ (↑)	3.290	2.135	2.636
HArD (↓)	2.535	0.3786	1.374

D. Experiments on Severe Fog

To further assess the quality of the proposed approach in presence of dense fog, we tested our model on the recently released Dense-Haze dataset [1], which is composed of 55 images where fog is artificially inserted with fog-machines in controlled conditions. This makes possible a direct comparison (in terms of classical PSNR and SSIM) with clear images of the same scene. The level of fog of the images is very high, and the visibility is almost zero in all the photographs. We compared our method with several state-of-the-art learning based defogging techniques [23, 5, 9, 13]. The datasets used in this experiment are:

- Synthetic paired dataset (training): *Outdoor Training Set (OTS)*, included in the RESIDE dataset [17].
- Real unpaired dataset (training): O-Haze dataset [2], which is composed of 45 high resolution pairs of fog and real images. The fog is inserted with fog machines and its very similar to real fog. We randomly crop each of the 45 images 45 times, producing an unpaired dataset of 2,025 images.
- Test dataset: Dense-Haze dataset (55 images with severe fog) [1].

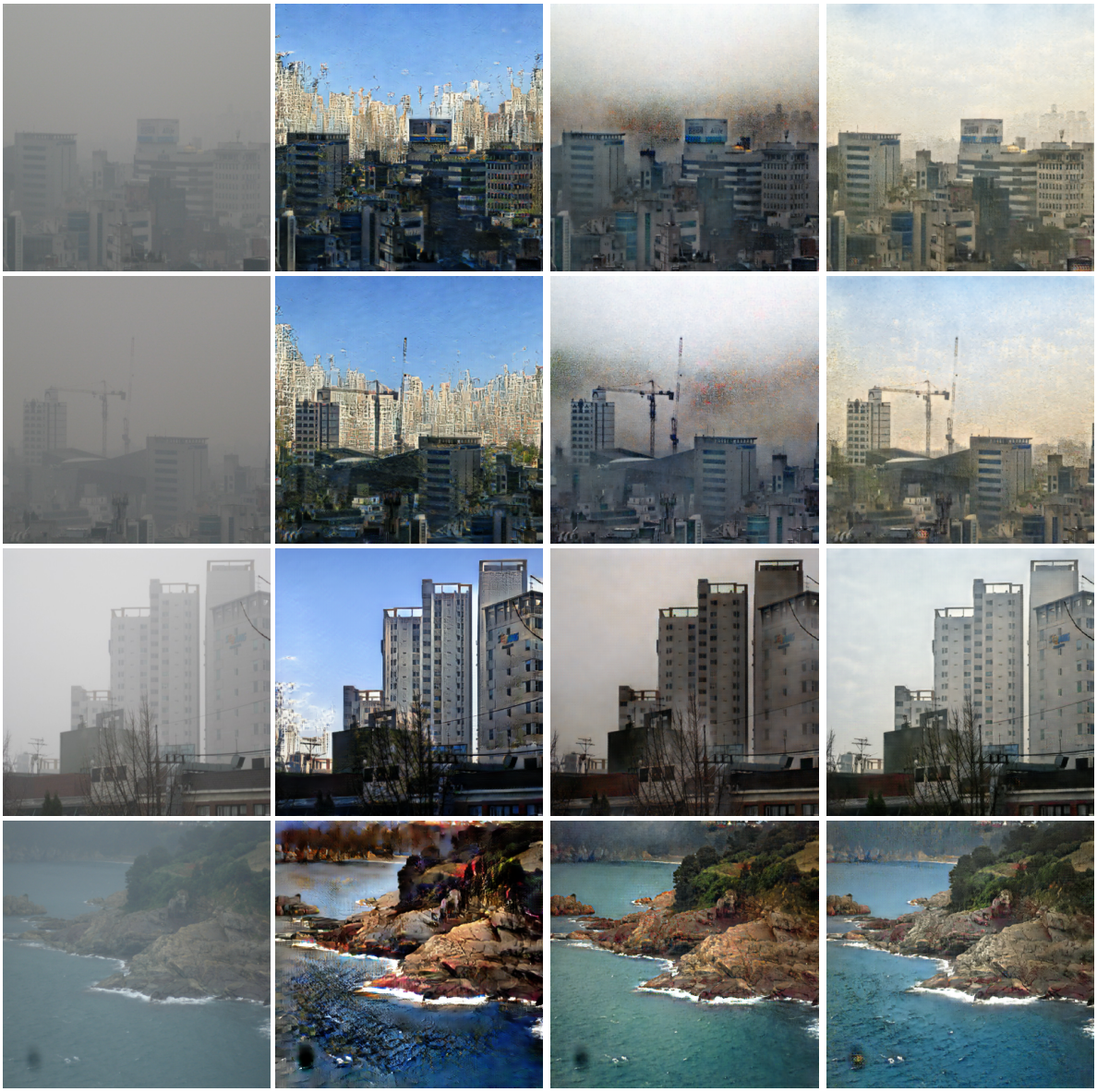
The results are reported in Table III. Note that our method was not trained with images of the Dense-Haze dataset, nor with images with a comparable high level of fog. The results in Table III demonstrate the effectiveness of CurL-Defog: our approach shows similar performance on the PSNR metrics w.r.t. the best baseline, but greatly outperform all the methods on the SSIM metric.

TABLE III: Average SSIM and PSNR scores of the dehazing on the Dense-Haze dataset

	PSNR	SSIM
MSCNN [23]	**12.52**	0.369
DehazeNet [5]	11.36	0.374
CycleDehaze [9]	10.54	0.261
Pix2Pix [13]	10.55	0.311
CurL-Defog	12.24	**0.469**

VI. Conclusions

In this work, we proposed CurL-Defog, a curriculum based approach that exploits both paired and unpaired data to effectively defog images even in the presence of severe fog. The main goal of CurL-Defog is to reduce the amount of artifacts inserted by unpaired defogging methods, while maintaining the same performance in terms of contrast enhancement and edge restoration. To numerically estimate the amount of artifacts in the defogged images we proposed HArD, a referenceless metrics that automatically detect regions in the resulting images that contains unwanted artifacts. We performed experiments both on synthetic and real fog, obtaining results in line with state-of-the-art approaches but with significantly less artifacts. Our study confirmed that the paired information provided by synthetic data are very useful to guide the model towards regions of the parameter space that minimize the insertion of artifacts especially in the case of severe fog. In the future, we intend to combine Hautière *et al*. metrics [10] with HArD in order to define artifact-immune referenceless metrics for image defogging.



(a) (b) (c) (d)

Fig. 4: Qualitative comparison of defogging on real images with severe fog. (a) foggy images, (b) unpaired approach [9] (note the heavy insertion of artifacts), (c) paired approach [13], (d) CurL-Defog.

References

[1] Codruta O. Ancuti et al. "Dense Haze: A Benchmark for Image Dehazing with Dense-Haze and Haze-Free Images". In: *IEEE International Conference on Image Processing (ICIP)*. IEEE ICIP 2019. Taipei, Taiwan, 2019.

[2] Codruta O Ancuti et al. "O-HAZE: a Dehazing Benchmark with Real Hazy and Haze-Free Outdoor Images". In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2018, pp. 754–762.

[3] Yoshua Bengio et al. "Curriculum Learning". In: *Proceedings of the 26th International Conference on Machine Learning*. 2009, pp. 41–48.

[4] Dana Berman, Shai Avidan, et al. "Non-Local Image Dehazing". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1674–1682.

[5] Bolun Cai et al. "Dehazenet: An End-to-End System for Single Image Haze Removal". In: *IEEE Transactions on Image Processing* 25.11 (2016), pp. 5187–5198.

[6] L. K. Choi, J. You, and A. C. Bovik. *LIVE Image Defogging Database*. http://live.ece.utexas.edu/research/fog/fade_defade.html. Accessed: 2019-05-30.

[7] Lark Kwon Choi, Jaehee You, and Alan Conrad Bovik. "Referenceless Prediction of Perceptual Fog Density and Perceptual Image Defogging". In: *IEEE Transactions on Image Processing* 24.11 (2015), pp. 3888–3901.

[8] Christophe De Vleeschouwer Cosmin Ancuti Codruta O. Ancuti. "D-Hazy: a Dataset to Evaluate Quantitatively Dehazing Algorithms". In: *IEEE International Conference on Image Processing (ICIP)*. ICIP'16. Pheonix, USA, 2016.

[9] Deniz Engin, Anıl Genç, and Hazım Kemal Ekenel. "Cycle-Dehaze: Enhanced CycleGAN for Single Image Dehazing". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2018.

[10] Nicolas Hautiere et al. "Blind Contrast Enhancement Assessment by Gradient Ratioing at Visible Edges". In: *Image Analysis & Stereology* 27.2 (2011), pp. 87–95.

[11] Kaiming He, Jian Sun, and Xiaoou Tang. "Single Image Haze Removal using Dark Channel Prior". In: *IEEE transactions on pattern analysis and machine intelligence* 33.12 (2010), pp. 2341–2353.

[12] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 770–778.

[13] Phillip Isola et al. "Image-to-Image Translation with Conditional Adversarial Networks". In: *arxiv* (2016).

[14] Harald Koschmieder. "Theorie der horizontalen Sichtweite". In: *Beitrage zur Physik der freien Atmosphare* (1924), pp. 33–53.

[15] Christian Ledig et al. "Photo-Realistic Single Image Super-Resolution using a Generative Adversarial Network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4681–4690.

[16] Boyi Li et al. "Aod-net: All-in-One Dehazing Network". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 4770–4778.

[17] Boyi Li et al. "Benchmarking Single-Image Dehazing and Beyond". In: *IEEE Transactions on Image Processing* 28.1 (2019), pp. 492–505.

[18] R. Li et al. "Single Image Dehazing via Conditional Generative Adversarial Network". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. June 2018, pp. 8202–8211.

[19] Yu Li et al. "Haze Visibility Enhancement: A Survey and Quantitative Benchmarking". In: *Computer Vision and Image Understanding* 165 (2017), pp. 1–16.

[20] Wei Liu et al. "End-to-End Single Image Fog Removal using Enhanced Cycle Consistent Adversarial Networks". In: *ArXiv* (2019).

[21] Earl J McCartney. "Optics of the Atmosphere: Scattering by Molecules and Particles". In: *New York, John Wiley and Sons, Inc., 1976. 421 p.* (1976).

[22] Y. Pang, J. Xie, and X. Li. "Visual Haze Removal by a Unified Generative Adversarial Network". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2018), pp. 1–1. ISSN: 1051-8215.

[23] Wenqi Ren et al. "Single Image Dehazing via Multi-Scale Convolutional Neural Networks". In: *European conference on computer vision*. Springer. 2016, pp. 154–169.

[24] Ashish Shrivastava et al. "Learning from Simulated and Unsupervised Images through Adversarial Training". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2107–2116.

[25] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *arXiv e-prints* (Sept. 2014). arXiv: 1409.1556.

[26] Jean-Philippe Tarel et al. "Vision Enhancement in Homogeneous and Heterogeneous Fog". In: *IEEE Intelligent Transportation Systems Magazine* 4.2 (2012), pp. 6–20.

[27] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. "Instance Normalization: The Missing Ingredient for Fast Stylization". In: *CoRR* abs/1607.08022 (2016).

[28] Z. Wang, E. P. Simoncelli, and A. C. Bovik. "Multiscale Structural Similarity for Image Quality Assessment". In: *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*. Vol. 2. 2003, 1398–1402 Vol.2.

[29] Zhou Wang et al. "Image Quality Assessment: from Error Visibility to Structural Similarity". In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.

[30] Jun-Yan Zhu et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks". In: *Computer Vision (ICCV), 2017 IEEE International Conference on Computer Vision*. 2017.

[31] Qingsong Zhu, Jiaming Mai, and Ling Shao. "A Fast Single Image Haze Removal Algorithm Using Color Attenuation Prior". In: *IEEE transactions on image processing* 24.11 (2015), pp. 3522–3533.