Towards Artifact-Free Image Defogging

Gabriele Graffieti, Davide Maltoni

e-mails: gabriele.graffieti@unibo.it, davide.maltoni@unibo.it

Departement of Computer Science and Engineering (DISI)
University of Bologna
Italy

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Problem

Defogging (or dehazing) is the task of removing the fog from an input image, aimed at reconstructing the same scene as if it were taken in good weather conditions.

Classical methods are often surpassed by DNN-based ones, using CNN or GANs. However, they often require paired data.

Problems of paired data:

- Impossible to obtain in natural conditions.
- Fog is inserted artificially.
  - No precision (depth map needed).
  - Synthetic fog ≠ real fog

Totally unpaired approaches show promising results in real fog conditions.

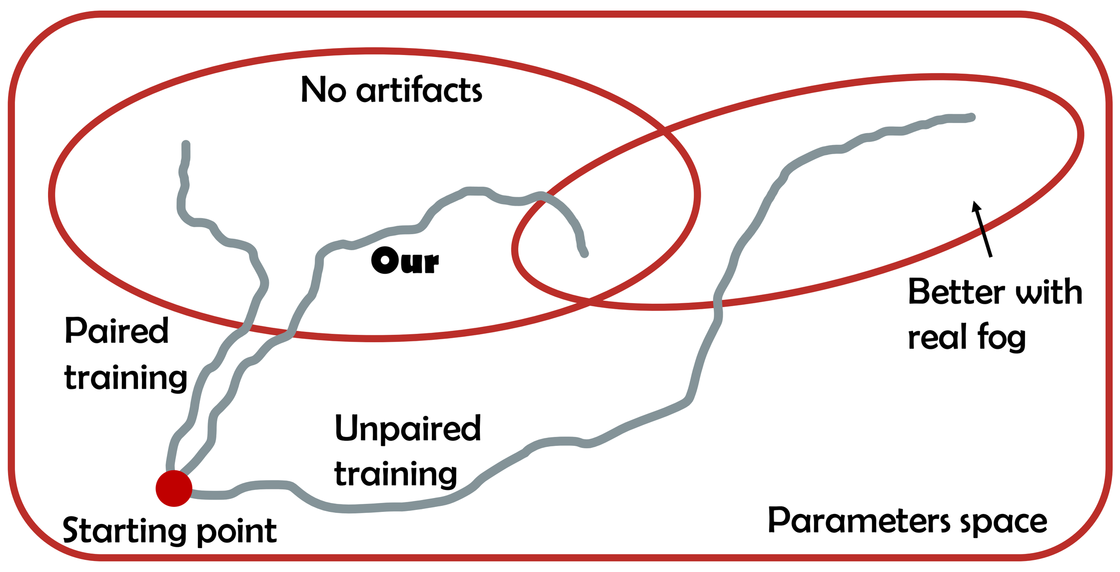
Problems of unpaired data:

- Models often insert unwanted artifacts in the defogged images.
  - Unrestricted nature of the unpaired training.

Idea

We could exploit paired data to force the model to not insert artifacts, and unpaired data to produce better results with real fog.

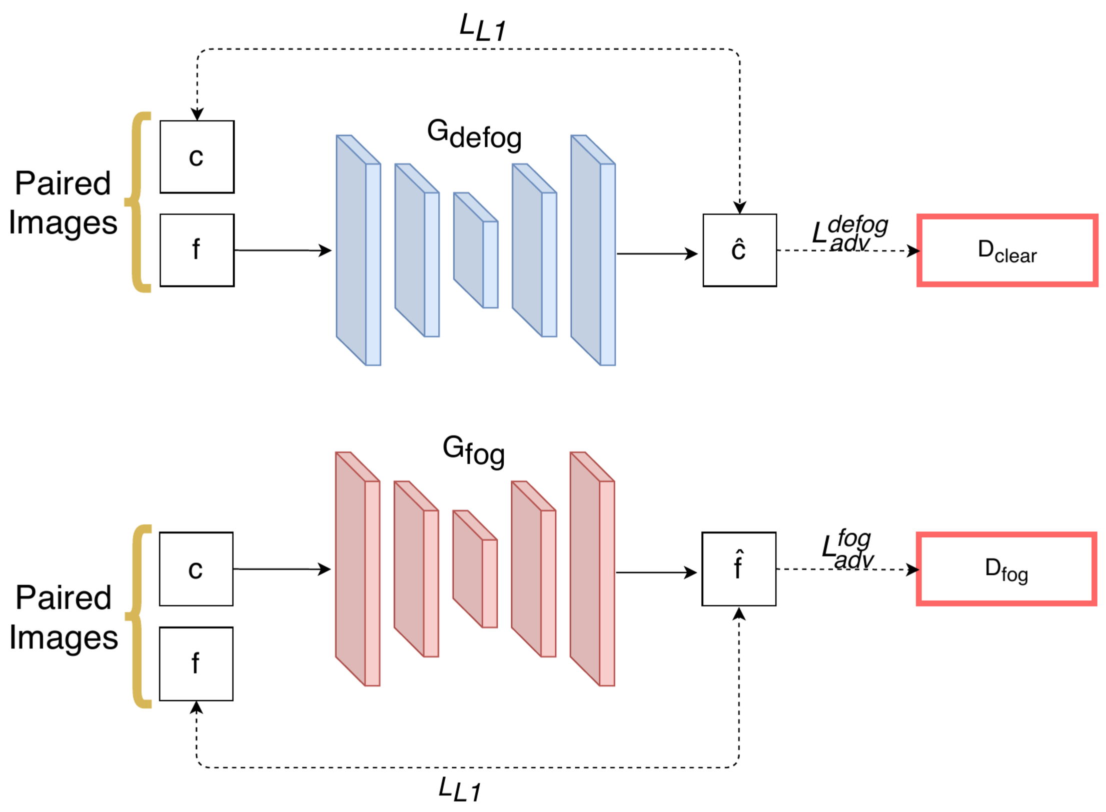
- Find the parameters that reduce artifacts insertion and optimize results on real fog.

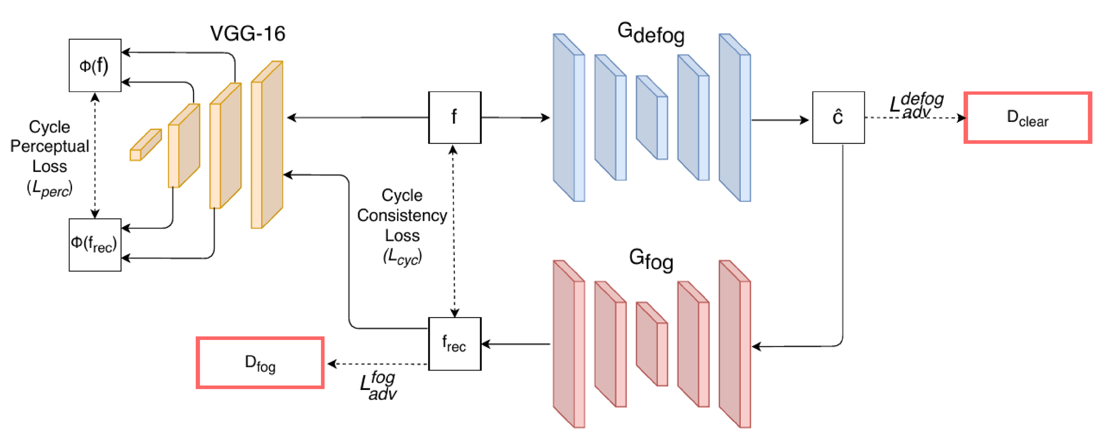


This is similar to curriculum learning, when a model is first trained on simpler data and then refined on harder one.

- First model is trained with paired synthetic data -> artifact insertion highly penalized.
- Then real images are used -> high quality on real fog.
- Gradual transition between paired and unpaired data.

The model (CurL-Defog)





Paired training: model acts as 2 different pix2pix models (one for defogging and one for "fogging").

Unpaired taining: models acts as a single cycleGAN model, with some additional loss terms:

$$\mathcal{L}_{perc} = \|\phi(f) - \phi(G_{fog}(G_{defog}(f)))\|_2^2 + \|\phi(c) - \phi(G_{defog}(G_{fog}(c)))\|_2^2$$

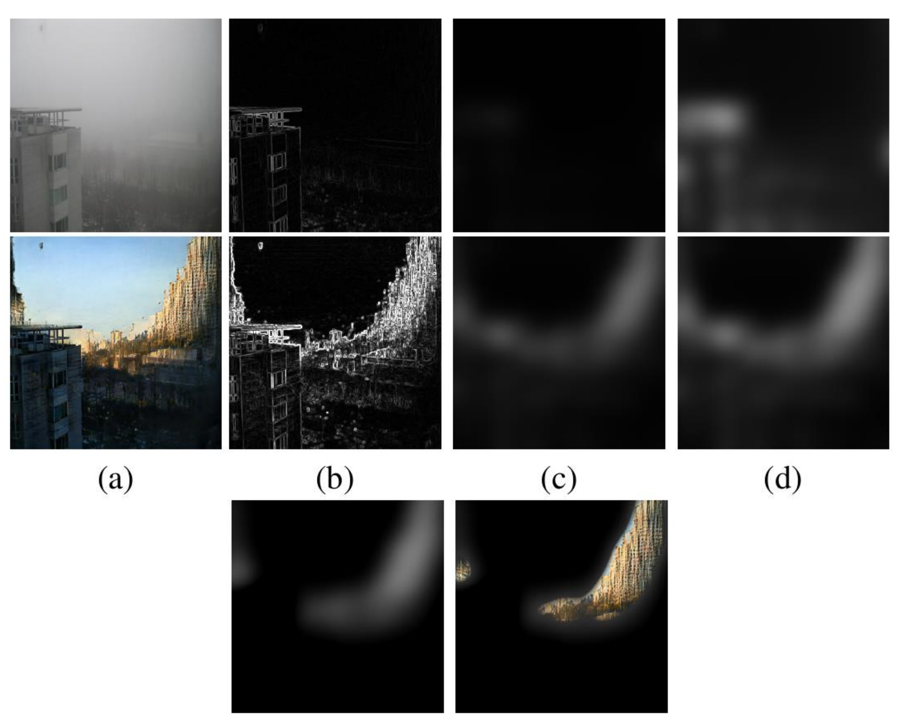
$$\mathcal{L}_{idt} = \|G_{fog}(f) - f\|_1 + \|G_{defog}(c) - c\|_1$$

Haze Artifact Detection (HArD)

- No defogging metrics takes into account the presence of artifacts.
- Some metrics may be deceived by the presence of artifacts, e.g., the ones that counts the number of visible edges.

Idea: no edges in the foggy image -> no edges in the defogged image.

- Find the regions in both images that contains edges and compare them.
- Regions with edges in the defogged image and not in the foggy one -> artifacts.



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

(e) (f)

HArD algorithm scketch

(a) input images (b) gradient computation by the Prewitt operator (c) smoothing with a Gaussian filter and scaling in the interval [0,1] (d) multiplication by a predetermined constant and saturation through the hyperbolic tangent. (e) and (f) difference between the two maps in (d).

Experiments and results

Experiments on synthetic data

- Synthetic dataset: OTS dataset.
- Real dataset: LIVE Image Defogging.
- Test set: HSTS dataset.

	DCP	CAP	NLD	DehazeNet
PSNR	14.84	21.53	18.92	24.48
SSIM	0.7609	0.8727	0.7411	**0.9183**

	MSCNN	AOD-Net	Pix2Pix	**CurL-Defog**
PSNR	18.64	20.55	24.22	**24.83**
SSIM	0.8168	0.8973	0.8991	0.9037

Experiments on real data

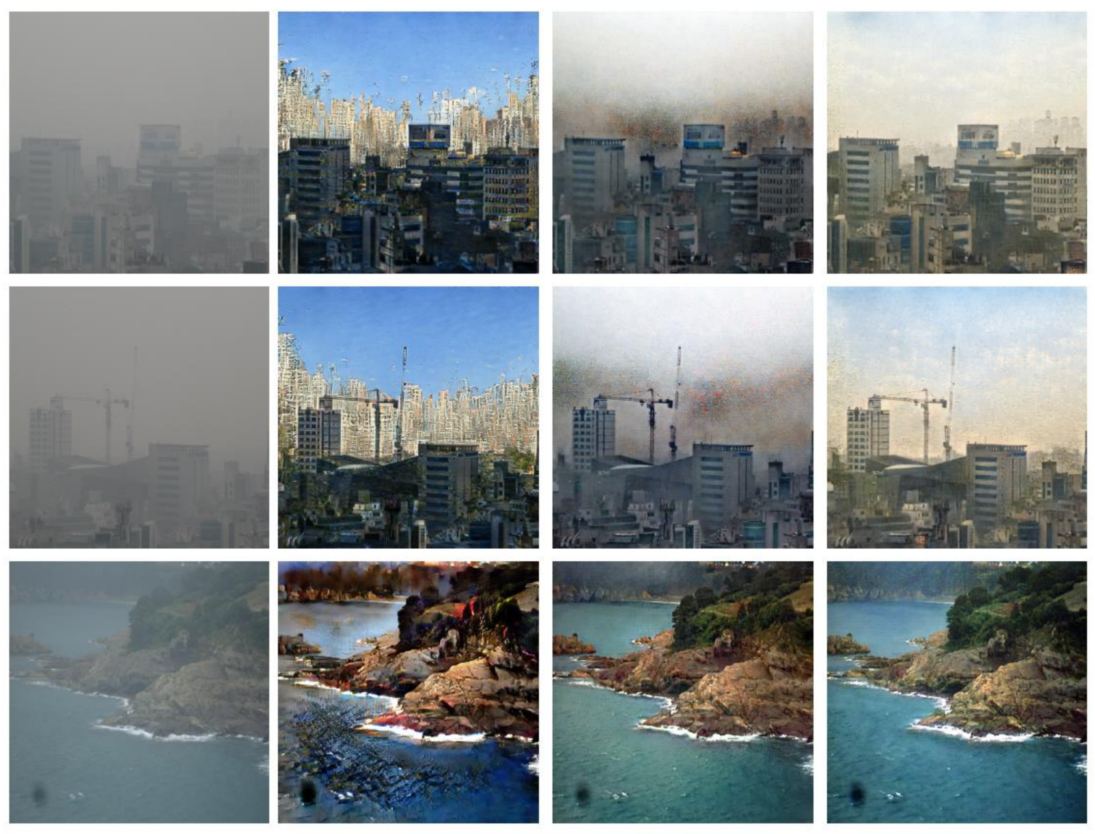
- Test set: LIVE Image Defogging test set.

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

Experiments on severe fog

- Real Dataset: O-Haze dataset
- Test set: Dense-Haze dataset

	MSCNN	DehazeNet	CycleDehaze	Pix2Pix	**CurL-Defog**
PSNR	**12.52**	11.36	10.54	10.55	12.24
SSIM	0.369	0.374	0.261	0.311	**0.469**



Foggy image CycleDehaze Pix2Pix **CurL-Defog**

Conclusions

- CurL-Defog: a curriculum learning-based novel defogging method that minimize the number of inserted artifacts.
- Effective defogging even with severe fog.
- A new referenceless metric (HArD) to numerically estimate the amount of defogging artifacts.