Unsupervised Non-Rigid Image Distortion Removal via Grid Deformation
Supplementary Material

Nianyi Li[1,3], Simron Thapa[1], Cameron Whyte[2], Albert Reed[2], Suren Jayasuriya[2], and Jinwei Ye[1]

[1]Louisiana State University, Baton Rouge, LA 70803, USA
[2]Arizona State University, Tempe, AZ 85281, USA
[3]Clemson University, Clemson, SC 29634, USA

In this supplementary material, we provide details on our network architecture (Section 1), and synthetic air turbulence simulation (Section 2). We show more ablation study results on different numbers of inputs and the Fourier feature mapping parameters (Section 3). Lastly, we include more visual comparison results with the state-of-the-arts on the synthetic air and water turbulence (Section 4).

1. Network Architecture

In this section, we provide the detailed architecture of our non-rigid distortion removal network. Our network has two main component the grid deformer $\mathcal{G}_\theta$ and the image generator $\mathcal{I}_\phi$. Table 1 provides detailed network architecture of the two subnets. In the tables, Conv, BN, ReLU refer to convolution layers, batch normalization and Rectified Linear Unit; γ refers to the GRFF position encoding component.

	Input	Filters	Output Shape
$\mathcal{G}_\theta$	Input		$128 \times 128 \times 2$
	Conv, ReLU, BN	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, ReLU	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, ReLU	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, Tanh	$2@1 \times 1$	$128 \times 128 \times 2$
$\mathcal{I}_\phi$	Input		$128 \times 128 \times 2$
	γ (GRFF)		$128 \times 128 \times 256$
	Conv, ReLU, BN	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, ReLU	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, ReLU	$256@1 \times 1$	$128 \times 128 \times 256$
	Conv, Sigmoid	$3@1 \times 1$	$128 \times 128 \times 3$

Table 1. Detailed architecture of grid deformer $\mathcal{G}_\theta$ and the image generator $\mathcal{I}_\phi$. Here, ReLU and BN stands for Rectified Linear Unit and Batch Normalization, respectively. We use the input image size $128 \times 128 \times 3$ as an illustrative example.

2. Air turbulence Simulation Details

We use the physics-based simulation software presented in [5] to render images affected by the air turbulence. Fig. 1 illustrates the simulation setup and the respective parameters used in the simulator. Numerical values of the parameters are given in Table 2.

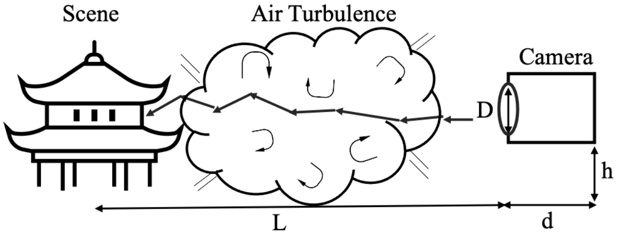

Figure 1. Air turbulence simulation setup. We simulate the distorted image of the scene through the turbulent air. Here L is the path length from camera to scene; d is the camera's focal length; h is the camera height; and D is the camera's aperture size.

Parameter		Value
Path length L		2km
Height h		4m
Aperture Diameter D		0.08m
Focal Length d		0.3m
Wavelength λ		550nm
Turbulence	Weak	1×10^{-14} m$^{-2/3}$
Strength	Medium	1×10^{-13} m$^{-2/3}$
C_n^2	Strong	1×10^{-12} m$^{-2/3}$

Table 2. Air turbulence simulation parameters

The simulator use the refractive image constant (C_n^2) to control the strength of the air turbulence. Stronger turbulence results in more distorted images. In our simulation,

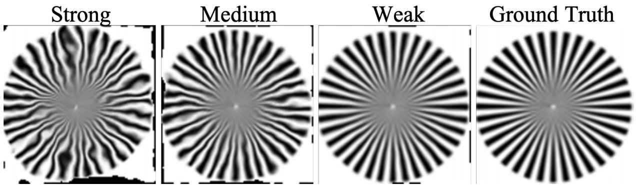

Figure 2. Exemplary images of the three turbulent strength levels (weak, medium, and strong) in comparison with the distortion-free image (taken from Open Turbulent Image Set (OTIS) [2]).

we use three levels of C_n^2 to render images under weak, medium, and strong air turbulence. Fig. 2 shows exemplary images of the three turbulent strengths.

3. More Results on Ablation Study

3.1. Effect of number of input images

In the main paper, we show the quantitative comparison results of different number of input images. Here Fig. 4 shows the visual comparison results. We can see that the qualitative improvement of the predicted distortion-free image becomes margin when the number of input images is greater than 10. We therefore take 10 input images as the default setting to balance performance and efficiency.

3.2. Effect of GRFF parameters

We evaluate the impact of the bandwidth-related scale factor κ for Fourier feature mapping. We show the quantitative comparison results (average PSNR/SSIM) in Table 3, and qualitative comparison results on distortion-free image and distortion field prediction in Fig. 4.

κ	0.1	1	8	10	50	100
PSNR	19.38	19.29	20.28	20.01	16.24	13.89
SSIM	0.627	0.800	0.796	0.754	0.373	0.372

Table 3. Quantitative comparison on varied Fourier feature mapping parameters κ. Red and Blue refer to the top and second best performance respectively.

3.3. Effect of regularization term $R(I_k)$

We conducted an ablation study by removing $R(I_k)$. The qualitative comparison results are shown Fig. 3. We can see that without using $R(I_k)$, the output image appear blurry. The quantitative comparison results are shown in Table 4. Both PSNR and SSIM are higher when $R(I_k)$ is added.

	Without $R(I_k)$	With $R(I_k)$
PSNR	19.54	**20.48**
SSIM	0.772	**0.795**

Table 4. Quantitative comparison on $R(I_k)$.

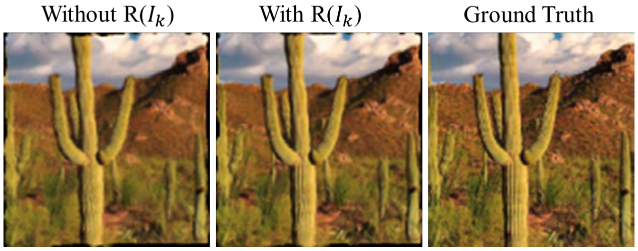

Figure 3. Qualitative comparison on regularization term $R(I_k)$.

4. More Results on Synthetic and Real Data

4.1. Air turbulence results

We show additional visual results on simulated air turbulence in Fig. 5. We compare on simulated air turbulence of three strength levels: weak, medium and strong. We compare with the state-of-the-art method CLEAR [1], whose source code is available. We also compare with the average image of the entire input sequence, as well as our initialization result.

We show more results on the real captured hot-air turbulence scenes in the supplementary video. The videos are filmed by imaging through the hot air generated by a lit gas stove. We include video clips of two scenes. Each scene has 50 frames. As recovering all frames together is computationally expensive, we divide the 50 frames into three batches: 20, 20 and 10. We then feed these three batches into our network to predict the distortion-free image and the distortion fields. In the videos, we show the predicted distortion-free image from the last batch.

4.2. Water turbulence results

We show additional visual results on simulated water turbulence in Fig. 6. We compare on simulated water turbulence of three types: ripple, ocean and Gaussian. We compare with the state-of-the-art methods Tian *et al.* [7], Oreifej *et al.* [4], and Li *et al.* [3].

We show more results on the real captured water turbulence scenes in the supplementary video. The real captured turbulent videos are provided by [6]. The videos are captured through a wavy water surface. We choose two different types of waves with different background patterns. The video processing method is similar to the air turbulence case. Each scene has 50 frames. We divide the 50 frames into three batches: 20, 20 and 10. We then feed these three batches into our network to predict the distortion-free image and the distortion fields. In the videos, we show the predicted distortion-free image from the last batch.

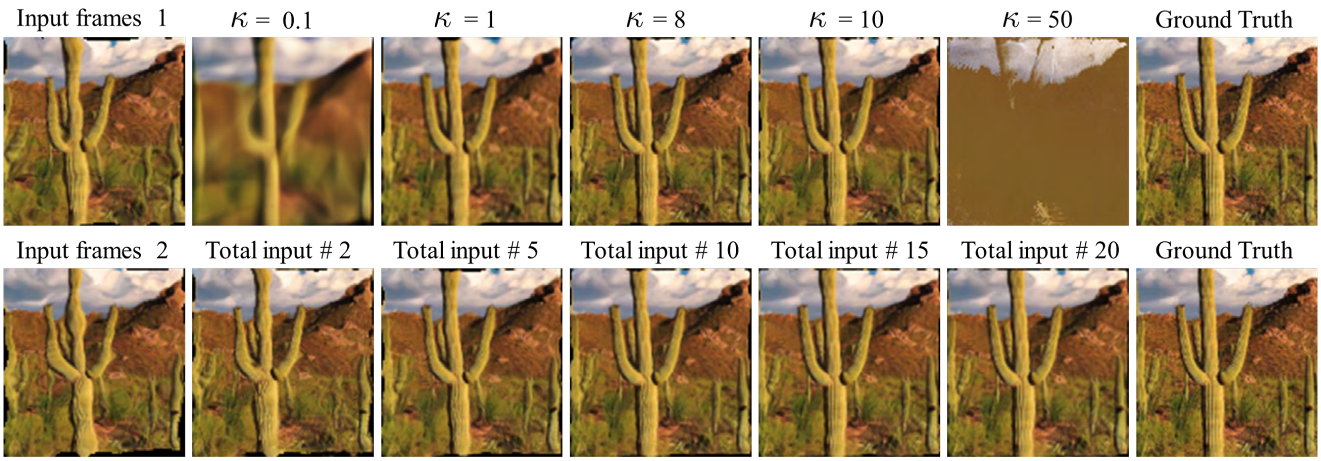

Figure 4. Qualitative comparison on varied values for $\kappa_{\mathcal{I}}$ and different numbers of input images (1, 5, 10, 15, and 20).

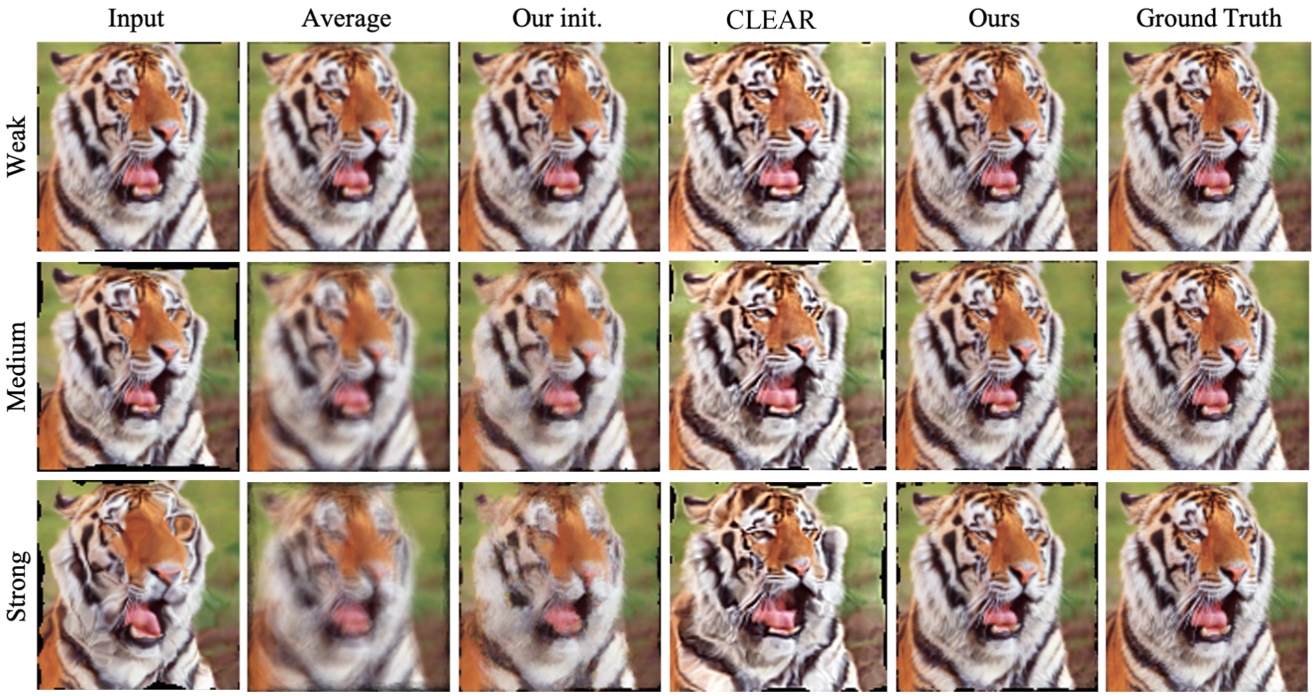

Figure 5. Qualitative comparison on simulated air turbulence images with various strengths (weak, medium, and strong).

References

[1] Nantheera Anantrasirichai, Alin Achim, and David Bull. Atmospheric turbulence mitigation for sequences with moving objects using recursive image fusion. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2895–2899. IEEE, 2018. 2

[2] Jérôme Gilles, Tristan Dagobert, and Carlo De Franchis. Atmospheric turbulence restoration by diffeomorphic image registration and blind deconvolution. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 400–409. Springer, 2008. 2

[3] Zhengqin Li, Zak Murez, David Kriegman, Ravi Ramamoorthi, and Manmohan Chandraker. Learning to see through turbulent water. In *Winter Conference on Applications of Computer Vision*, pages 512–520, 2018. 2

[4] Omar Oreifej, Guang Shu, Teresa Pace, and Mubarak Shah. A two-stage reconstruction approach for seeing through water. In *CVPR 2011*, pages 1153–1160. IEEE, 2011. 2

[5] Armin Schwartzman, Marina Alterman, Rotem Zamir, and Yoav Y Schechner. Turbulence-induced 2d correlated image distortion. In *2017 IEEE International Conference on Com-*

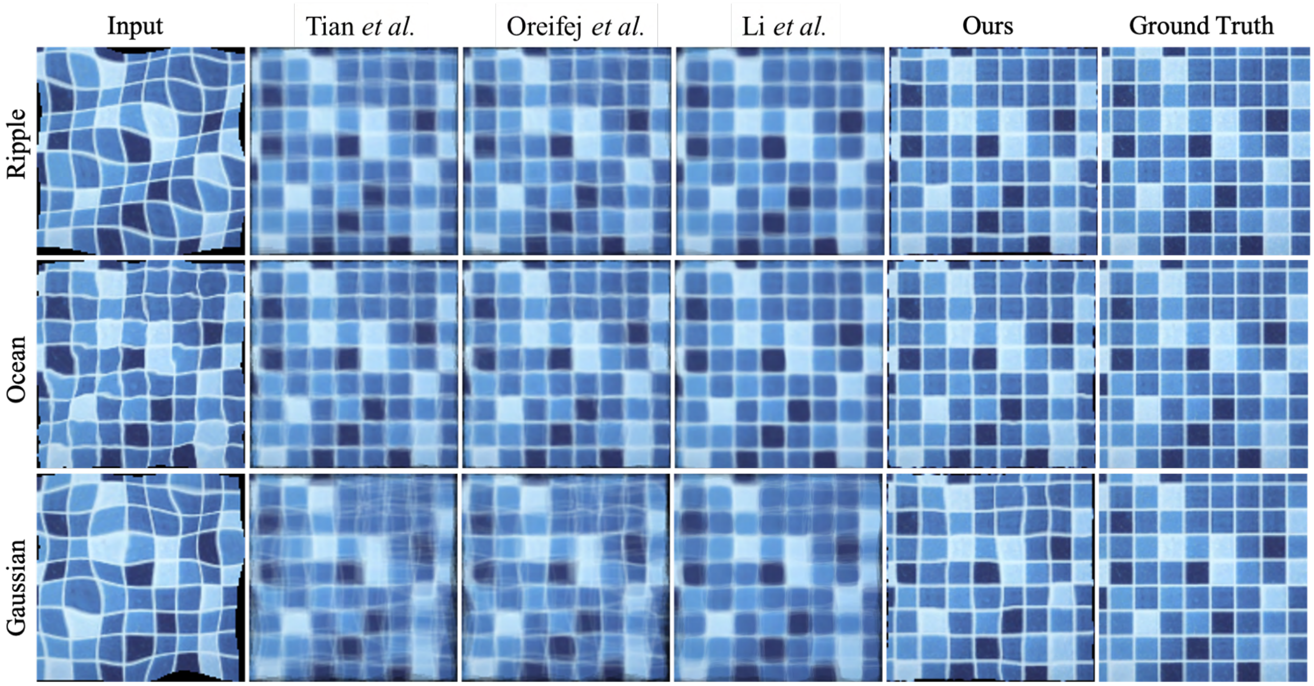

Figure 6. Qualitative comparison on simulated water turbulence images with various types of waves (ripple, ocean, and Gaussian).

putational Photography (ICCP), pages 1–13. IEEE, 2017. 1

[6] Simron Thapa, Nianyi Li, and Jinwei Ye. Dynamic fluid surface reconstruction unsing deep neural network. In *Conference on Computer Vision and Pattern Recognition*, 2020. 2

[7] Yuandong Tian and Srinivasa G Narasimhan. Seeing through water: Image restoration using model-based tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2303–2310. IEEE, 2009. 2