Removing Atmospheric Turbulence via Deep Adversarial Learning

Shyam Nandan Rai and C. V. Jawahar, *Member, IEEE*

***Abstract*— Restoring images degraded due to atmospheric turbulence is challenging as it consists of several distortions. Several deep learning methods have been proposed to minimize atmospheric distortions that consist of a single-stage deep network. However, we find that a single-stage deep network is insufficient to remove the mixture of distortions caused by atmospheric turbulence. We propose a two-stage deep adversarial network that minimizes atmospheric turbulence to mitigate this. The first stage reduces the geometrical distortion and the second stage minimizes the image blur. We improve our network by adding channel attention and a proposed sub-pixel mechanism, which utilizes the information between the channels and further reduces the atmospheric turbulence at the finer level. Unlike previous methods, our approach neither uses any prior knowledge about atmospheric turbulence conditions at inference time nor requires the fusion of multiple images to get a single restored image. Our final restoration models DT-GAN+ and DTD-GAN+ outperform the general state-of-the-art image-to-image translation models and baseline restoration models. We synthesize turbulent image datasets to train the restoration models. Additionally, we also curate a natural turbulent dataset from YouTube to show the generalisability of the proposed model. We perform extensive experiments on restored images by utilizing them for downstream tasks such as classification, pose estimation, semantic keypoint estimation, and depth estimation. We observe that our restored images outperform turbulent images in downstream tasks by a significant margin demonstrating the restoration model's applicability in real-world problems.**

***Index Terms*— Atmospheric turbulence, image restoration, generative adversarial networks.**

I. Introduction

IMAGING through atmospheric turbulence has been a well-researched subject over the past few decades. Preliminary works on this topic were focused on astronomical applications [1]. Presently, image restoration from atmospheric turbulence has tremendous potential applications in long-range video surveillance, defence systems, and drone imaging systems. These systems capture images of an object at a distance measured in the order of several kilometres. Slight perturbations in atmospheric conditions caused by atmospheric turbulence can trigger significant changes in the object's geometrical and perceptual information present in such images.

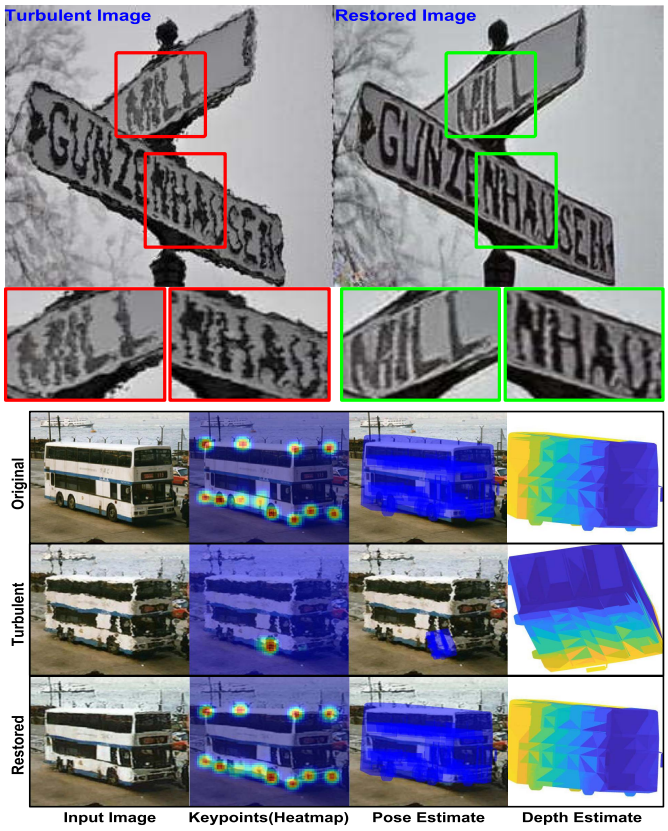

Fig. 1. Top: An example of image restoration by our framework. Magnified image patches of the turbulent image are red, and the restored image patches are in green. Bottom: Restored image got from our framework showing improvement over the turbulent image in various computer vision tasks. **(Best viewed when zoomed)**.

Hence, we need to develop restoration methods to minimize such adverse effects. Leveraging the recent developments in deep learning, we attempt to restore images under atmospheric turbulence and show improvement in various computer vision tasks, as shown in Figure 1.

Atmospheric turbulence [2] occurs when there is a slight perturbation in air pressure, temperature, and gaseous levels in the atmosphere. Temperature change is a major component in atmospheric turbulence near the earth's surface. A slight temperature change can cause random fluctuations in the refractive index along the camera's optical path, resulting in image blur and perceptual degradation of the imaged object. Besides, deep atmospheric turbulence introduces geometrical distortion [3] similar to the shearing effect, which is unevenly distributed into different parts of the image. These effects are commonly observed in the sandy desert and rocket exhaust.

Manuscript received June 26, 2019; revised January 25, 2021, June 30, 2021, September 3, 2021, and December 7, 2021; accepted February 17, 2022. Date of publication March 16, 2022; date of current version March 23, 2022. This work was supported in part by the Department of Science and Technology (DST), Government of India, through the IMPacting Research, INnovation and Technology (IMPRINT) Program. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hairong Qi. *(Corresponding author: Shyam Nandan Rai.)*

The authors are with the Center for Visual Information Technology, International Institute of Information Technology Hyderabad, Hyderabad 500032, India (e-mail: shyam.nandan@research.iiit.ac.in).

Digital Object Identifier 10.1109/TIP.2022.3158547

Mathematically, atmospheric turbulence in an image can be modeled [4] as:

$$T = HI + \epsilon \tag{1}$$

where T is the turbulent image, I is the original image, H is a transformation matrix, and ϵ includes other noises like camera sensor noise. H can be considered a combination of transformations caused by geometrical distortions and image blur in atmospheric turbulence. Equation 1 shows that the problem is an ill-posed inverse problem because the value of H and ϵ needs to be solved using a single equation. Therefore, instead of finding the exact solution, we approximate the results closest to the exact solution. Neural networks have proven to have sufficient capabilities to approximate [5]–[7] complex non-linear functions. Hence, employing the recent advancements in deep learning makes learning an approximate mapping from a turbulent image to an original image possible.

An atmospheric turbulent image contains a mixture of distortions that consists of geometrical distortion distributed irregularly in an image combined with spatial image blur. This kind of image degradation makes the restoration of images challenging and is distinct from other inverse problems such as deblurring [8] and denoising [9]. Figure 3(a) shows a real atmospheric turbulent image in which we can observe the geometrical distortions and the image blur. Whereas, Figure 3(b) displays a blurry image having inadequate high-frequency information, and Figure 3(c) shows a noisy image [10] occurs when an image pixel present in the camera sensor receives a varying amount of photons. But, image blurring or noise does not have geometrical distortions and suffers only from a single distortion form, i.e., noise or blur. Hence, using a deblurring or denoising network to minimize atmospheric turbulence will lead to sub-optimal solutions, as these networks are not intended to mitigate geometrical distortions. Only a few data-driven methods [11], [12] have been proposed to restore atmospheric turbulent images, making the problem statement relatively new and narrow.

A single-stage deep network will find it difficult to minimize atmospheric turbulence from an image as the network has to simultaneously remove the geometrical distortions and image blur. Hence, we employ a deep network consisting of two stages: a) In the first stage, we remove the geometrical distortions from the image using WarpNet, and b) In the next step, the image's blurriness is removed by ColorNet to improve the overall perceptual quality. Further, we add channel attention and sub-pixel mechanism to utilize the information between the channels and learn to remove atmospheric turbulence at the finer level. Our proposed model DTD-GAN+ outperforms the state-of-the-art image-to-image translation methods and prior atmospheric turbulence removal methods. We also show an empirical study in Sub-Section IV(H) that our proposed architecture is specifically intended for minimizing atmospheric turbulence.

A deep network requires a large amount of turbulent and original image pairs to learn the mapping. However, there is no such large-scale dataset available to train the network. Moreover, rendering such images using computer graphics requires a high computational cost. To overcome this problem, we follow a systematic way by Schwartzman *et al.* [13] to inject atmospheric turbulence into images by a series of efficient 2D operations. We use this simulation method to benchmark restoration model performance over images on diverse simulated atmospheric turbulence levels. We also curate a dataset from YouTube consisting of natural atmospheric turbulence images to show our proposed restoration model's generalizability. We use the restored images in vision applications such as semantic keypoints detection, depth estimation, classification, and pose estimation, outperforming results on turbulent images. Additionally, we demonstrate the performance of restored images on datasets having diverse classes. Figure 2 illustrates the overall methodology. We will interchangeably use non-turbulent and original image terms to represent the ground-truth images.

To summarize, the major contributions of this paper are:

- We propose a deep adversarial network to minimize atmospheric turbulence. Unlike the earlier methods, our approach neither uses any prior knowledge about atmospheric turbulence at test time nor requires multiple images to reconstruct the restored image or struggle to remove finer atmospheric turbulence.
- Our proposed network use channel attention and sub-pixel mechanism to exploit the information between the channels and remove the atmospheric turbulence at the finer level achieving better restoration results.
- We synthesize a large-scale turbulence dataset comprised of original and turbulent image pairs for training the network. Additionally, we curate a dataset from YouTube consisting of natural atmospheric turbulence images.
- Our final restoration model achieves state-of-the-art performance over general image-to-image translation methods and prior atmospheric turbulence removal methods. Our model also gives impressive restoration results for natural turbulent images and synthetic turbulent images having diverse turbulence conditions.
- Extensive evaluations using restored images show significant improvement over turbulent images in the various vision tasks performance. Further, to show the restored image generalization ability, we evaluate several dataset's performances having a different number of classes for downstream tasks.

II. Related Work

Restoring images in atmospheric turbulence: Restoring images with atmospheric turbulence can be broadly categorized as a) Adaptive Optics b) Lucky Imaging. Adaptive optics requires expensive and massive instruments for removing atmospheric turbulence. They mainly used these methods for astronomical applications [14], [15]. On the other hand, lucky imaging algorithms take short exposed distorted images set from which lucky regions are selected and then fused to get the final image. Some of the preliminary work in lucky imaging [16] uses probabilistic analysis to restore the degraded images. Various methods [17], [18] have been proposed for enhancing images and videos which have been degraded due to atmospheric turbulence using lucky imaging.

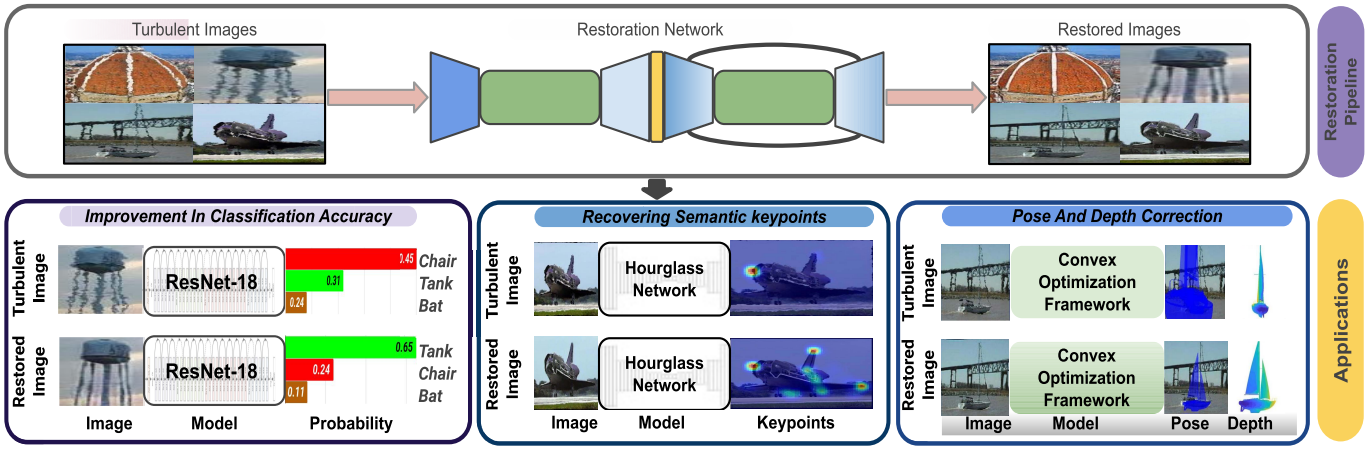

Fig. 2. Illustration of our proposed overall methodology. Top: Restoration pipeline for the turbulent images. Bottom: Application of restored images indicating improvements in tasks such as classification, keypoints detection, pose, and depth estimation.

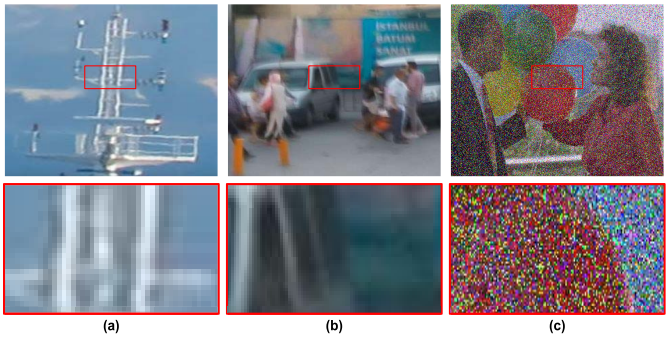

Fig. 3. Examples of various distortions occur in an image. (a) An image captured in atmospheric turbulence, due to which it suffers from geometrical distortions and image blur. (b) A blurry image having blurry edges. (c) Noisy image caused due to varying amount of photon received by the camera sensor. This shows that atmospheric turbulence consists of various distortions and cannot be addressed through deblurring and denoising networks.

In image enhancement, these methods use a multi-frame image reconstruction approach to correct geometric distortion and reduce the blur introduced by atmospheric turbulence. For video enhancement, it is performed by fusing a stream of randomly distorted images. Statistical methods and Fourier analysis techniques [19], [20] were also attempted to remove atmospheric turbulence.

Zhu *et al.* [21] proposed a machine learning method that first registers each frame to suppress geometric distortion through B-spline based on nonrigid registration. Next, a temporal regression process is carried out to produce an image from the registered frames. It can be viewed as being convolved with space invariant near-diffraction-limited blur. Finally, a blind deconvolution algorithm is implemented to deblur the fused image and generate the final output. The method suffered from several limitations. For example, the process uses the temporal mean to calculate the reference image, which results in poor registration. Xie *et al.* [22] proposed an improved method that first constructs a high-quality reference image from the set of observed frames using low-rank matrix decomposition. The reference image is iteratively optimized to further improve the registered sequence using a variational framework containing a new spatiotemporal regularization. All of the above methods fail to restore an image using a single turbulent image since multiple turbulent images are required to obtain a single restored image. Recently, deep learning methods have been proposed by Gao *et al.* [12] and Chak *et al.* [11]. These methods either struggle to remove finer atmospheric turbulence or take multi-frame input to get single restored images. On the other hand, our proposed method can minimize finer geometrical distortions while requiring only a single turbulent image to get the restored image.

Image-to-Image transformation via deep learning: The recent success of deep neural networks have drastically improved the performance for image classification [23]–[25], object detection [26], [27], and object segmentation [28]. These networks can also be employed for generating natural images using deep generative networks, such as Generative Adversarial Network(GAN) [29], [30], VAE [31], [32], PixelRNN [33], and PixelCNN [34] which involves mapping from a highly non-linear manifold to another. GAN are the most successful among the generative networks due to their photo-realistic output [35], [36]. Several image inverse problems such as super-resolution [37], deblurring [38], and denoising [39] extensively uses GAN to produce clean images. Particularly, GAN based image denoising methods is used in tomography [40] and microscopy [41].

Apart from generating realistic images, deep neural network architectures can also manipulate geometric structures. These architectures have been used in gaze manipulation [42], image matching [43], image registration [44], image transformation [45], and restoring geometrical distortions in an image caused due to turbulent water [46].

Attention mechanism in deep learning: Attention can be interpreted as selectively concentrating on a particular piece of information while ignoring the other perceivable information. In other words, it can also be viewed as allocating the majority weight to the essential information of an input. Attention mechanism has been widely used in deep learning methods related to computer vision [47], [48], and natural language processing [49]. In parallel, several works [50], [51] has been done to combine spatial attention with channel attention to improve models on various vision tasks.

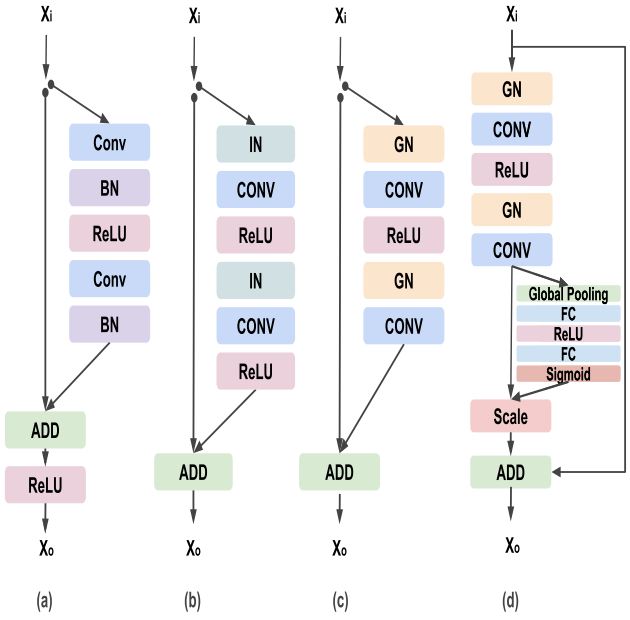

Fig. 4. Architectural comparison of varying residual blocks: (a) He *et al.* [25], (b) Li *et al.* [46], (c) Baseline1 (ours), and (d) DTD-GAN (ours).

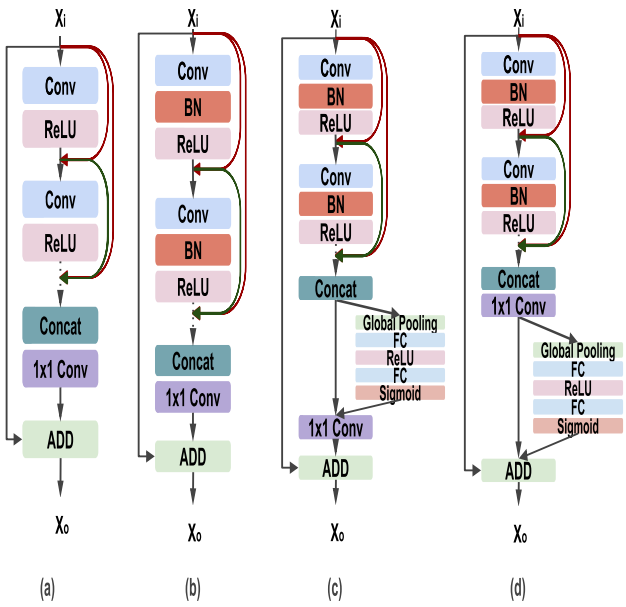

Fig. 5. Architectural comparison of varying dense residual blocks: (a) Zhang *et al.* [58], (b) Baseline2 (ours), (c) DTD-GAN(pre) (ours), and (d) DTD-GAN (ours).

Hu *et al.* [52] recently introduced a channel attention module for improving the quality of representations. It explicitly models the interdependencies between the channels and convolutional features of a deep network. This module has been extensively used in improving denoising [9], super-resolution [53], [54], deblurring [8], and dehazing [55]. Hence, we incorporated channel attention in our residual blocks by observing their efficacy in improving various computer vision tasks.

III. Method

In this section, we will discuss the architectural details of our proposed model. Our proposed model is inspired by Li *et al.* [46]; however, it significantly differs in the following ways: a) We introduce an array of residual blocks and dense residual blocks, suitably modified for minimizing atmospheric turbulence. We add a channel attention module into these blocks to improve the restoration. b) Next, we propose a sub-pixel mechanism that removes finer geometrical distortions from a turbulent image and produces a high-quality restored image. c) Finally, we improve the overall network's performance by adding a new loss to the objective function that minimizes loss between WarpNet output and ground truth image, acting as additional supervision. Now, we will describe our proposed method in the following subsections.

A. Residual Blocks

Residual networks perform well on various computer vision tasks to efficiently carry forward information deep into the network with residual connections. Figure 4 and 5 shows the various residual blocks used in our experiments. Initially, we began our experiment with the residual block of Li *et al.* [46] shown in Figure 4(b), that removed distortions in the turbulent water. We empirically find that replacing Instance Normalization (IN) [56] layers with Group Normalization (GN) [57] layers shown in Figure 4(c) improved our performance. This is because GN layers work better on a broader range of batch sizes and show better training stability and accuracy. Dense residual block [58] shown in Figure 5(a) perform better than the residual block as its dense connection can access all the subsequent layers in a block and pass on information that needs to be preserved within a block. Hence, we replace the residual block with a dense residual block and add Batch Normalization (BN) [59] layers, shown in Figure 5(b). Adding the BN layers helps stabilize the network training. We also tried to include GN layers, but it gave slightly poorer restoration results than the BN layer.

Channel Attention: The above discussed residual blocks do not focus on convolutional feature's channel relationships. So, we incorporate the channel attention module by Hu *et al.* [52] into our residual blocks. The channel attention models the interdependencies between convolutional features and channels, which improves the network's quality of representations. Figure 4(d), 5(c) and (d) show the residual blocks, including channel attention. In Figure 5(c) and (d), we add the channel attention mechanism either before or after the concatenation to empirically explore our network's performance.

B. Network Architecture

Figure 6 shows our adversarial network's generator architecture, divided into two sub-networks: WarpNet and ColorNet. WarpNet removes the geometrical distortion introduced by atmospheric turbulence by learning a warping field applied to the input image, resulting in a warped image as output. The WarpNet output is blurry and lacks high-frequency details due to atmospheric blur, and one-to-one mapping is not ensured in the forward mapping of WarpNet. So, ColorNet is appended after the WarpNet, which restores the perceptual details and minimizes the image blur. The detailed network architecture of WarpNet and ColorNet is discussed below.

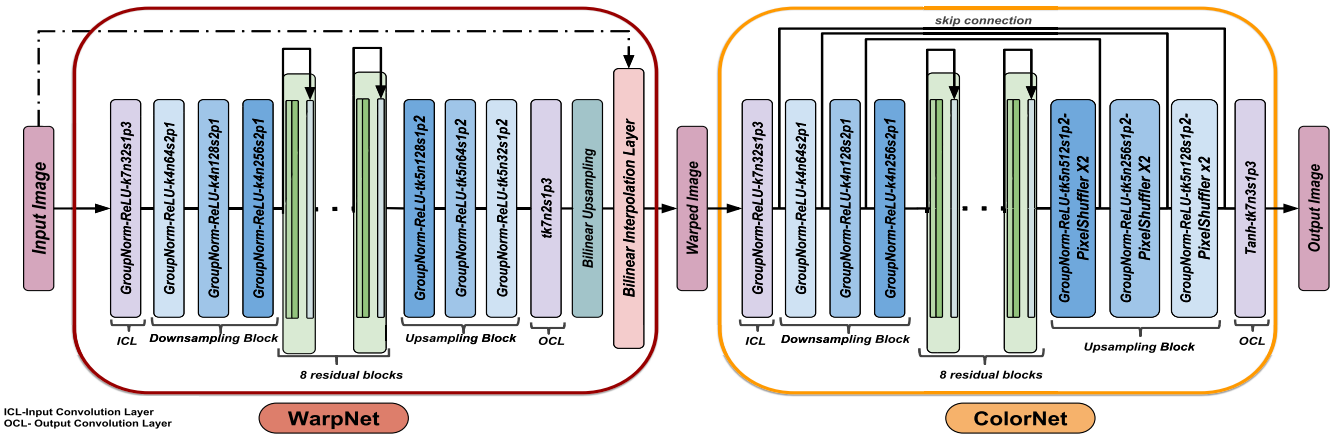

Fig. 6. The general architecture of the generator for adversarial training. The generator architecture of DT-GAN and DTD-GAN are formed by using the modified residual blocks of Figure 4(d) and 5(d) respectively. In Figure, k is the kernel size, n is the number of feature maps, and s is the stride in each convolutional layer with p as padding.

WarpNet consists of the input convolutional layer, upsampling block, eight residual blocks, downsampling block, and output convolutional layer. The input convolutional layer projects the input image to feature space, passed on to downsampling blocks. The downsampling block consists of 3 convolutional layers, 3 ReLU layers, and 3 Group Normalization layers. Each convolutional layer comprises of filter size 4×4 with the stride of 2 and padding 1. The depth of the output channel doubles after each convolution layer. The downsampling block's output is then passed through residual blocks and further upsampled using upsampling block. The structure of the upsampling block is similar to the downsampling block. It uses a deconvolutional layer with a bi-linear strategy for upsampling the feature map. The output channel's depth decreases from 256 to 32 output channels across the downsampling block's deconvolution layers. At last, we use the output convolutional layer to get the final warping field. The architecture of ColorNet is slightly different from WarpNet, as ColorNet has skip-connections [60] which help in recovering information lost during downsampling.

Sub-pixel mechanism: Accurate motion estimation is made possible by calculating the pixel's motion at sub-pixel accuracy. We inculcate this idea into the WarpNet, which allows it to learn smaller geometrical distortions. We introduced this notion by adding a bilinear-upsampling layer into the WarpNet. The bilinear-upsampling layer doubles the predicted warp field's size, allowing it to predict the smaller pixel movement caused by the finer atmospheric turbulence. The warp field is further applied to the input image to remove the geometrical distortions. The final output is then downsampled by half and given as an input to the ColorNet. Next, in the upsampling block of ColorNet, we replace the bi-linear interpolation layers, which learn a single filter to upsample the features with an efficient sub-pixel convolution layer [61] that learns a collection of upscaling filters to upsample the features. We call the above-discussed process the sub-pixel mechanism.

Baselines: Figure 4(c) shows our proposed Baseline1. We add channel attention and sub-pixel mechanism in Baseline1 to construct De-Turbulence Generative Adversarial Networks (**DT-GAN**), shown in Figure 4(d). DT-GAN demonstrates the restoration improvement over Li *et al.* [46] by using enhanced residual structure along with sub-pixel mechanism and channel-attention. Next, we create Baseline2 by adding a batch normalization layer to a dense residual block, as shown in Figure 5(b). We further add channel attention and sub-pixel mechanism into Baseline2 to construct De-Turbulence Dense Generative Adversarial Networks (**DTD-GAN**) in Figure 5(d). In section IV (Experiments), we empirically find that the post concatenation model: DTD-GAN shown in Figure 5(d) performs better than the pre concatenation model: DTD-GAN(pre) shown in Figure 5(c). We also adopt a self-ensemble strategy [62] to maximize the performance of our model. In this method, we create multiple copies of the input image via data augmentation. We then inverse augment all the corresponding outputs to align with the input image. The final result is the average of all the aligned outputs. The self-ensemble models are denoted by adding $'+'$ as post-fix to the model name. Table I summarizes all the above-discussed model configurations.

TABLE I

CONFIGURATIONS OF PROPOSED BASELINES AND THEIR FINAL MODELS

Model	Dense Residual Block	Channel Attention	Sub-Pixel Mechanism	Self-Ensemble
Baseline1	✗	✗	✗	✗
DT-GAN	✗	✓	✓	✗
DT-GAN+	✗	✓	✓	✓
Baseline2	✓	✗	✗	✗
DTD-GAN	✓	✓	✓	✗
DTD-GAN+	✓	✓	✓	✓

C. Objective Functions

Our network objective is to minimize the loss between the original image I and the network output taking corresponding turbulent image I^D as an input image. To achieve the same, we train the generator consisting of WarpNet (W^r_θ) parameterised by learnable parameters (θ) and ColorNet (C^o_ϕ) parameterised by learnable parameters (ϕ). We train the network on two objective functions: $l_{generator}$ which minimizes Generator output with the original image, and $l_{WarpNet}$ minimizing WarpNet output with the original image. We discuss the two objective functions in detail below. $\mathbf{l_{generator}}$: The loss function

Fig. 7. Sample images from the synthetically generated turbulent image dataset. **(Best viewed when zoomed)**.

consists of a linear combination of content loss, perceptual loss, and adversarial loss. The intuition behind adding adversarial loss into the loss function is to produce images in the natural image manifold. The architecture of the discriminator is the same as used for training DCGAN [30]. Instead of training discriminator with the sigmoid cross-entropy loss function, we adopted the least square loss function [63] which resulted in more stable training and high-quality image generation. The loss function for the generator and discriminator can be formulated as:

$$\begin{aligned} l_{generator} &= \lambda_2[D_\gamma(C^o_\phi(I^\omega)) - 1]^2 \\ &+ \sum_{i=1}^{N}\sum_{j=1}^{M}|I_{i,j} - C^o_\phi(I^\omega)_{i,j}| \\ &+ \lambda_1 \sum_{x=1}^{w_{4,3}}\sum_{y=1}^{h_{4,3}}|\psi_{4,3}(I)_{x,y} - \psi_{4,3}(C^o_\phi(I^\omega))_{x,y}| \end{aligned} \tag{2}$$

$$l_{discriminator} = [D_\gamma(C^o_\phi(I^\omega))^2 + (D_\gamma(I) - 1)^2] \tag{3}$$

where, D_γ represents the discriminator with γ as the trainable parameter and the values of λ_1 and λ_2 were empirically found during training the network.

$\mathbf{l_{WarpNet}}$: The loss between WarpNet output and original image acts as additional supervision to the network, which is trained by minimizing the sum of perceptual loss [64] and content loss. The content loss is formulated by taking the pixel-wise L1 distance between the original image and the reconstructed image from WarpNet. We use a perceptual loss to measure the perceptual similarity between images by taking the L1 distance between feature representation of Conv4_3 of VGG16 [24] of the original and restored image by WarpNet.

Fig. 8. Example images from real atmospheric dataset.**(Best viewed when zoomed)**.

The overall loss function for WarpNet can be described as:

$$\begin{aligned} l_{WarpNet} &= \sum_{i=1}^{N}\sum_{j=1}^{M}|I_{i,j} - I^\omega_{i,j}| \\ &+ \sum_{x=1}^{w_{4,3}}\sum_{y=1}^{h_{4,3}}|\psi_{4,3}(I)_{x,y} - \psi_{4,3}(I^\omega)_{x,y}| \end{aligned} \tag{4}$$

where $\psi_{4,3}$ is the feature map and $w_{4,3}$, $h_{4,3}$ are the dimensions of the feature map of VGG16 at Conv4_3 layer output. I and I^ω are the original and restored image from the WarpNet receptively.

IV. Experimentation

A. Dataset

Simulated Atmospheric Turbulent Dataset: Rendering turbulent images using computer graphics utilizes high computational power and time. To overcome this computational burden, we used an efficient way of rendering turbulent images derived by [13]. This method bypasses 3D numerical calculations and relies on a closed-form model for creating a computationally inexpensive and faster way of simulating turbulent fields. The virtual camera parameters used for creating the simulated atmospheric turbulence dataset are focal distance $= 300mm$ with a lens diameter $\approx 5.357cm$ and pixel size of $4e-3mm$. We place the virtual camera at an elevation of $4m$ with an object distance of $2km$. The value for structure constant C_n^2 which expresses the turbulent strength is $3e-13m^{-2/3}$. The light traveling from an object point is assumed to be having a spherical wavefront with a wavelength of $550nm$. Using the above parameters, we rendered turbulent images by applying simulated turbulent fields on the ImageNet dataset. The synthesized dataset consists of 400,000 turbulent and original image pairs selected from 1000 classes of ImageNet for training the restoration network. We validate all the restoration network performance on ImageNet validation dataset images consisting of 5995 pairs of synthesized turbulent and original images. Figure 7 shows a few examples of synthesized samples.

Real Atmospheric Turbulent Dataset: To test the restoration model's generalization ability, we created a test dataset of images having natural atmospheric turbulence. The images of the real turbulent dataset were curated from YouTube videos,

TABLE II

QUANTITATIVE COMPARISON BETWEEN VARIOUS IMAGE-TO-IMAGE TRANSLATION METHODS AND ATMOSPHERIC TURBULENCE REMOVAL METHODS. TEXT IN GREEN INDICATES THE BEST PERFORMING MODEL

	Pix2Pix [35]	DeblurGAN [38]	Gao *et al.* [12]	CycleGAN [66]	Li *et al.* [46]	**DTD-GAN+**
PSNR	17.4403	17.9132	18.7319	18.8531	20.4092	**22.1382**
SSIM	0.4445	0.4917	0.5268	0.5392	0.6508	**0.7316**
MSE	1.4549	1.4406	1.4381	1.4373	0.3527	**0.2876**

Fig. 9. Visual comparison of various restoration methods. We can notice that our best performing method: DTD-GAN+, indicated in blue, qualitatively outperforms the compared baselines and image-to-images translation-based restoration method.

which were either taken in deep turbulent environments or recorded at a distance of several km by long-range video cameras. The steps to make the dataset is as follows. Firstly, we searched the videos on YouTube from the keywords such as "atmospheric turbulent" and "long-range turbulent videos" and their combinations. We then filtered out the search results by manually streaming the videos and identifying those with considerable atmospheric turbulence. Subsequently, we downloaded and extracted the video's frames, which resulted in approximately a thousand images. Next, we manually selected those frames with significant atmospheric distortions and resized them into an image size of 256 × 256. Finally, we had 13 images with significant atmospheric turbulence and 217 with moderate atmospheric turbulence. Figure 8 shows the example images of the real atmospheric dataset.

TABLE III
QUANTITATIVE PERFORMANCE OF OUR PROPOSED BASELINES AND METHOD. GREEN INDICATES THE BEST PERFORMING MODEL

	Baseline1	DT-GAN	DT-GAN+	Baseline2	DTD-GAN(pre)	DTD-GAN	DTD-GAN+
PSNR	21.1302	21.4483	21.8652	21.6801	21.7324	21.8960	**22.1382**
SSIM	0.6936	0.7076	0.7145	0.7156	0.7134	0.7226	**0.7316**
MSE	0.3272	0.3168	0.3010	0.3033	0.3019	0.2964	**0.2876**

Application Datasets: We use the restored images got from various restoration models to show improvement in different computer vision tasks. We evaluate the restored images on three levels of datasets: (a) General: consisting of multiple categories, (b) Categorical: single category having multiple classes, and (c) Single object instance: single class instance within a category. For image classification, a subset of 100,000 ImageNet images is chosen for training and evaluated on 5000 images for the general classification. In categorical classification, the FGCV [65] dataset is used consisting of 6667 training images of aircraft categories belonging to 70 class instances and validated on 3334 images. We randomly chose the spitfire class from FGCV datasets to evaluate a single class instance, comprising 200 training images and 130 validation images. We use the PASCAL3D+ [66] dataset for keypoint detection, pose, and depth estimation with the same distribution of training and validation set of images in all the tasks. The dataset consists of 2D images containing objects belonging to 12 categories with semantic keypoint annotation. Each keypoint is mapped to a 3D CAD model giving the object's pose and depth in 3D space. We used 15,271 training images for the general case, and the trained model is validated on 5991 images. We randomly choose the airplane category of the PASCAL3D+ dataset, consisting of 1244 training images and 566 validation images. We selected all the images from the propeller class of airplane category to evaluate improvement in a single class instance.

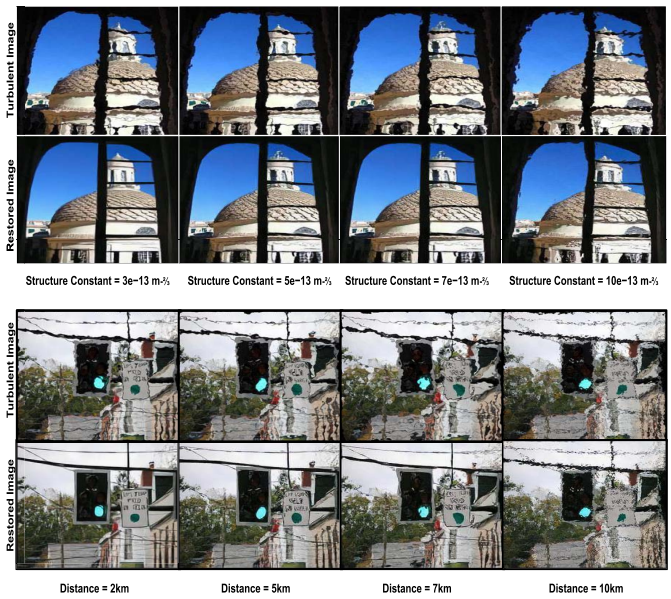

Fig. 10. Row 1 shows the turbulent image and Row 2 shows its corresponding restored image over a range of structure constant. Row 3 display the turbulent image and Row 4 displays its corresponding restored image over a range of object distance. **(Best viewed when zoomed)**.

B. Training and Implementation Details

We train our proposed restoration network on 400 thousand pairs of turbulent and original images. Each image is randomly flipped horizontally and resized for data augmentation, followed by cropping. We normalize each training batch with a mean of 0.5 and a standard deviation of 0.5. Real atmospheric turbulent images have a higher image blur than simulated turbulent images. We also blur the training images using Gaussian blur with varying sigma [0.5, 2.5] while testing the restoration model on real atmospheric turbulent images.

The restoration network is jointly trained end-to-end for six epochs with an initial learning rate of $1e-4$ and the next three epochs with a learning rate of $5e-5$ with batch size 16. To optimize the network, we use Adam [68] optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ for computing running averages of gradient and its squares. The values of λ_1 and λ_2 were empirically found to be 0.5 and 0.2, respectively. For each iteration, we update the discriminator and the generator only once. It took 24 hours to train the entire network on a single Nvidia GeForce GTX 1080 Ti GPU.

C. Image Quality Metrics

Image quality is an important measure to evaluate the perceptual and structural information present in an image. To measure the quality of the restored images, we use Peak Signal-to-Noise-Ratio (PSNR), Structural Similarity (SSIM), and Mean Square Error (MSE). These image quality metrics are applied to the restored and non-turbulent images to compare various restoration models quantitatively.

D. Restoration Results

Table III shows the performance of baselines and our proposed methods on various image quality metrics. We can see DTD-GAN performs better DT-GAN as it can remove finer geometric distortions, which is observed in Figure 9. In Table III, we can see DTD-GAN(pre) shows lower performance than DTD-GAN. In DTD-GAN(pre), we apply channel attention to all the features, giving higher weights to noisy or less critical features. Whereas in DTD-GAN, we pass all the features through the 1×1 convolutional layer, which squashes out the relevant features and then gives those features to the channel attention module giving better performance. Adding self-ensemble models denoted by $'+'$ added as a post-fix in our proposed model gives an additional improvement, making DTD-GAN+ the best restoration model.

We compare our final model DTD-GAN+ with Pix2Pix [35], CycleGAN [67], DeblurGAN [38],

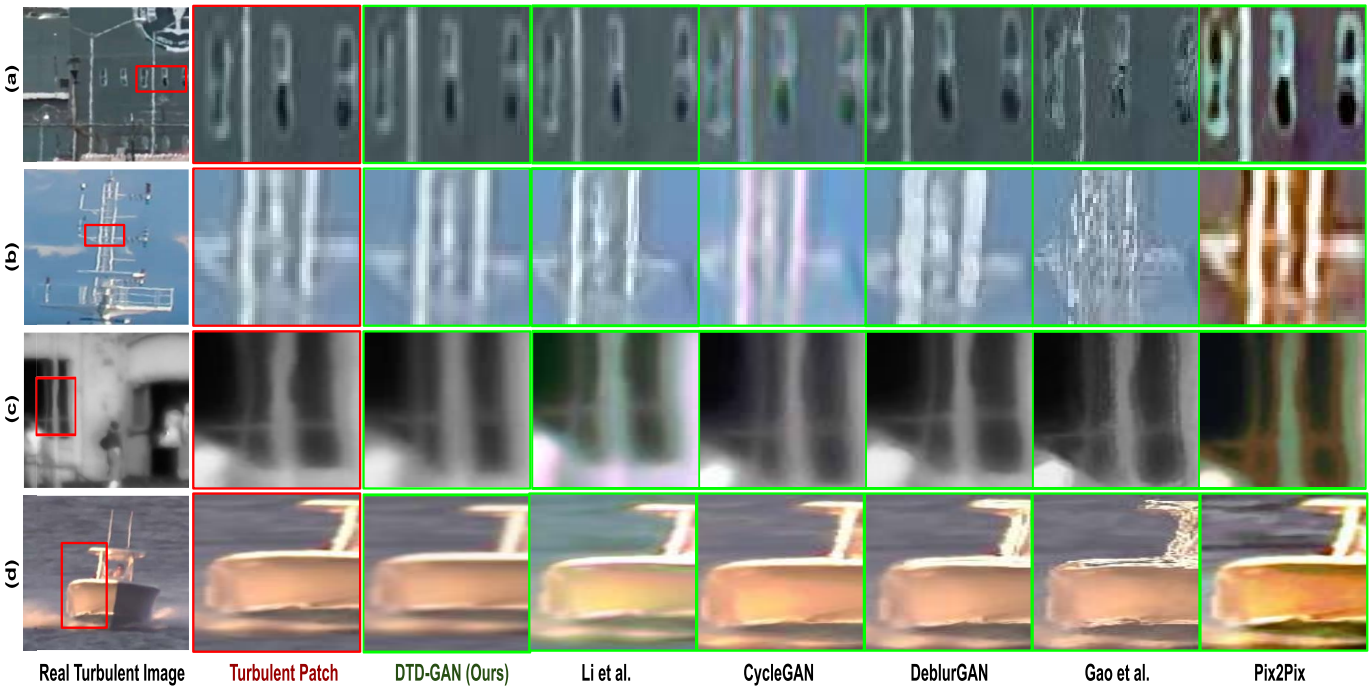

Fig. 11. Shows the restoration of natural turbulent images curated from Youtube. Red and green box contains the turbulent and restored image patch, respectively. Column 1: Natural turbulent image, and Column 2 shows the corresponding turbulent image patch. Column 3: Restored images by DTD-GAN. We can observe that DTD-GAN can minimize the geometrical distortion at the finer level. Column 4: Natural turbulent images restored by Li *et al.* [46]. The restored images suffer from geometrical distortion at the finer lever, which is significant in (b,c) and suffers from image artifacts. Column 5-8: Restored images by CycleGAN, DeblurGAN, Gao *et al.* [12] and Pix2Pix. All the restored images significantly suffer from image artifacts and struggle to remove the geometrical distortions. **(Best viewed when zoomed)**.

Gao *et al.* [12], and Li *et al.* [46]. Our final model DTD-GAN+ shows significant quantitative improvement over the methods mentioned above, as shown in Table II. Figure 9 shows the qualitative restoration results. We can observe from the figure that the restored images from Pix2Pix, CycleGAN, DeblurGAN, and Gao *et al.* [12] have significant geometric distortion and suffer from image artifacts. Li *et al.* [46] slightly elevates the problem of image artifacts, but it is unable to minimize finer geometrical distortions. Our proposed model DTD-GAN+ not only removes the geometric distortion but also alleviates the problem of image artifacts. Figure 11 shows the restoration of images having natural atmospheric turbulence. We find that DTD-GAN performs the best qualitatively, showing the generalization capability of our proposed model on a real dataset. Our model also minimizes image artifacts and finer atmospheric turbulence, prominent in other baselined methods. We use DTD-GAN by removing the self-ensemble strategy [62] denoted by '+' because it introduces blurriness in the final output due to the averaging of the outputs. However, this method gives higher quantitative results on various image metrics.

E. Ablative Study of Baselines

In this subsection, we perform an ablative investigation on Baseline1(BS1) and Baseline2(BS2) that consists of the proposed residual block (Figure 4(c)) and dense residual block (Figure 5(b)), respectively. Each baseline consists of a two-stage network: WarpNet and ColorNet. We create multiple baseline variations: *BS1 w/o WN*: Baseline1 without WarpNet, *BS1 w/o CN*: Baseline1 without ColorNet, *BS2 w/o WN*: Baseline2 without WarpNet, and *BS2 w/o CN*: Baseline2

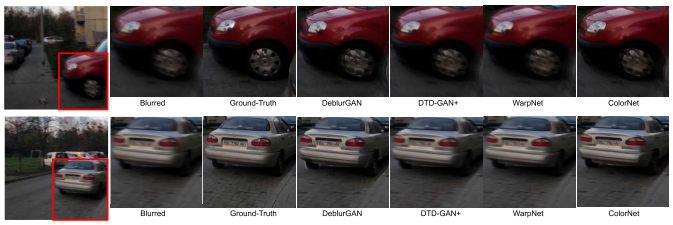

Fig. 12. Our method is specially designed for atmospheric turbulence. Figure presents the image-deblurring results obtained from various restoration networks. The results validate that the proposed network DTD-GAN+ and its corresponding sub-networks, WarpNet and ColorNet, perform poorly on image deblurring, as expected. **(Best viewed when zoomed)**.

without ColorNet. Table IV shows the depletion in the performance of baselines by removing WarpNet or ColorNet. The reason being ColorNet by itself struggles to remove the geometric distortion caused by atmospheric turbulence. Whereas WarpNet finds it challenging to remove the image blur. We observe that the performance of WarpNet is better than ColorNet as the former restores the structural information of an image. Furthermore, we notice that the two-stage model consisting of both ColorNet and WarpNet performs better than a single-stage model that contains either of the two. Next, we train the baselines on our proposed loss function. In Table V, we can observe that adding $l_{WarpNet}$ to the $l_{generator}$ improved the restoration performance of the baselines on various image quality metrics.

F. Effects of Channel-Attention and Sub-Pixel Mechanism

This subsection presents improved restoration due to the addition of channel attention and sub-pixel mechanism into our Baseline1(BS1) and Baseline2(BS2). We run multiple

TABLE IV

RESTORATION PERFORMANCE OF BASELINE1(BS1) AND BASELINE2(BS2) EITHER WITHOUT WARPNET(WN) OR COLORNET(CN). WE OBSERVE THAT THE BASELINES CONSISTING OF THE JOINT MODEL OF BOTH WN AND CN PERFORM THE BEST, INDICATED IN BLUE

	BS1 w/o WN	**BS1 w/o CN**	**BS1**	**BS2 w/o WN**	**BS2 w/o CN**	**BS2**
WarpNet	✗	✓	✓	✗	✓	✓
ColorNet	✓	✗	✓	✓	✗	✓
PSNR	20.943	21.075	21.130	21.441	21.579	21.680
SSIM	0.6761	0.6854	0.6936	0.7051	0.7127	0.7156
MSE	0.3321	0.3292	0.3272	0.3167	0.3122	0.3033

TABLE V

PERFORMANCE IMPROVEMENT IN BASELINES BY TRAINING ON NOVEL LOSS FUNCTION DENOTED BY: $l_{generator}\,(l_{gen}) + l_{WarpNet}\,(l_{WN})$

	Baseline1		**Baseline2**	
	l_{gen}	$l_{gen} + l_{WN}$	l_{gen}	$l_{gen} + l_{WN}$
PSNR	20.901	21.130	21.418	21.680
SSIM	0.6803	0.6936	0.7071	0.7156
MSE	0.3481	0.3272	0.3098	0.3033

variations of baselines to study the effects of channel attention, and sub-pixel mechanism gives as BS1 w/ SPM: Baseline1 with sub-pixel mechanism, BS1 w/ CA: Baseline1 with channel attention, BS2 w/ SPM: Baseline2 with sub-pixel mechanism, and BS2 w/ CA: Baseline2 with channel attention. Table VI shows that DT-GAN and DTD-GAN, which is the joint contribution of the sub-pixel mechanism and channel attention which, gives the highest performance improvement in our baselines. The performance improvement is due to channel attention and sub-pixel mechanisms that improve inter-channel dependency and remove atmospheric turbulence at the finer level. We also observed that the improvement in BS1 is more significant than BS2 because dense residual blocks already carry forward the lower-level information, which reduces the combined effect of channel-attention and sub-pixel mechanism.

G. Analysis of Varying Distance and Structure Constant Restoration

The two major parameters that control atmospheric turbulence are the object distance from the camera and the refractive fluctuation of an environment characterized by structure constant C_n^2. So, we analyze our proposed restoration model's restoration capabilities at various object distances and structure constant C_n^2. In Figure 10(Row 2), we observe that our model can restore images at various strength of C_n^2 ranging from $3e - 13m^{-2/3}$ to $1e - 12m^{-2/3}$ at 2 km object distance. A high value of C_n^2 is $8e-13m^{-2/3}$ which is usual in summer daytime, can be easily restored by our model as shown in Figure 10(Row 2). Figure 10(Row 4) shows that our restoration model can restore images at a distance ranging from $2-10$ Km keeping the C_n^2 to be $3e - 13m^{-2/3}$. Figure 10(Row 4) shows that our model struggles to restore turbulent images at a distance of 10 Km as the image degradation due to atmospheric turbulence is very high.

H. Limitations of Proposed Architecture

In this subsection, we show that DTD-GAN+ is specifically intended for removing atmospheric turbulence. We present an empirical study that shows DTD-GAN+ performs better on tasks to minimize atmospheric turbulence than DeblurGAN. Whereas, DeblurGAN gives better results for image deblurring tasks than DTD-GAN+ and its sub-network: WarpNet and ColorNet.

Atmospheric Turbulence Minimization: Now, we demonstrate the performance of DTD-GAN+ and DeblurGAN on minimizing atmospheric turbulence. We use Wasserstein GAN with gradient penalty as the GAN variant in DeblurGAN. We train all the networks on the ImageNet turbulent dataset consisting of 400, 000 turbulent and non-turbulent image pairs. All networks use an image patch size of 256×256 as input to the network with a batch size of 16. The learning rate is set to $1e - 4$ for the initial seven epochs and then linearly decayed to zero for the subsequent seven epochs. Figure 9 shows the qualitative results obtained from the proposed method DTD-GAN+ and DeblurGAN. We can observe from the Figure 9 that images restored from DeblurGAN contain large geometrical distortions. In contrast, the DTD-GAN+ restoration results given in the Figure 9 shows that DTD-GAN+ has effectively minimized the atmospheric turbulence present in the images. Additionally, Table VIIb reflects the performance of DTD-GAN+ on various image quality metrics, which is better than DeblurGAN.

Image De-blurring: Now, we will compare the performance of DTD-GAN+ and DeblurGAN on image deblurring tasks. Additionally, we also show the image deblurring results by the sub-networks of DTD-GAN+ that are WarpNet and ColorNet. For training and evaluating the networks, we use the GoPro dataset [69] that consists of 2103 pairs of blurred and sharp image pairs having $720p$ image quality. We train each network for 150 epochs with a batch size of 8 and a learning rate of $1e - 4$.

Table VIIa and Figure 12 show the quantitative and qualitative image deblurring performance of DeblurGAN, DTD-GAN+, WarpNet, and ColorNet. We observe that ColorNet performs better than WarpNet in the deblurring task because ColorNet is responsible for minimizing image blur while minimizing atmospheric turbulence. However, WarpNet performs poorly among all the methods because it is explicitly designed to mitigate the geometrical distortions. WarpNet uses a sub-pixel mechanism to calculate precise motion that captures the geometrical distortions caused by atmospheric turbulence, and this does not complement the task of minimizing image blur. It results in sub-optimal performance of DTD-GAN+. Additionally, we also notice the performance of ColorNet is close to DeblurGAN as both networks are designed to minimize image blur.

TABLE VI

IMPROVEMENT IN THE PERFORMANCE OF BASELINE1(BS1) AND BASELINE2(BS2) BY ADDING CHANNEL ATTENTION(CA) AND SUB-PIXEL MECHANISMS(SPM). BOLD ENTRIES INDICATE THAT INTEGRATING CA AND SPM INTO THE BASELINES GIVES THE BEST PERFORMANCE

	BS1	**BS1 w/ SPM**	**BS1 w/ CA**	**DT-GAN**	**BS2**	**BS2 w/ SPM**	**BS2 w/ CA**	**DTD-GAN**
Channel Attention	✗	✗	✓	✓	✗	✗	✓	✓
Sub-Pixel Mechanism	✗	✓	✗	✓	✗	✓	✗	✓
PSNR	21.1302	21.2933	21.4052	**21.4483**	21.6801	21.7307	21.8400	**21.8960**
SSIM	0.6936	0.7014	0.7057	**0.7076**	0.7156	0.7206	0.7217	**0.7226**
MSE	0.3272	0.3217	0.3173	**0.3168**	0.3033	0.2997	0.2989	**0.2964**

TABLE VII

(A) QUANTITATIVE RESULTS COMPARISON FOR IMAGE DEBLURRING TASK. WE SHOW THE DEBLURRING PERFORMANCE OF OUR PROPOSED FRAMEWORK: DTD-GAN+ AND ITS SUB-NETWORKS: WARPNET AND COLORNET. WE OBSERVE THAT DEBLURGAN PERFORMS WELL ON IMAGE DEBLURRING TASKS COMPARED TO OUR PROPOSED DTD-GAN+. (B) QUANTITATIVE RESULTS COMPARISON ON THE TASK OF MINIMIZING ATMOSPHERIC TURBULENCE. WE CAN SEE THAT DTD-GAN+ OUTPERFORMS DEBLURGAN. TEXT IN BOLD REPRESENTS THE BEST-PERFORMING METHOD

	DeblurGAN	DTD-GAN+	WarpNet	ColorNet
PSNR	**27.302**	25.194	24.807	26.971
SSIM	**0.9058**	0.8615	0.8596	0.8915
MSE	**0.1513**	0.1977	0.2067	0.1614

(a)

	DeblurGAN	DTD-GAN+
PSNR	17.9132	**22.1382**
SSIM	0.4917	**0.7316**
MSE	1.4406	**0.2876**

(b)

TABLE VIII

QUANTITATIVE RESULTS ON ORIGINAL, TURBULENT AND RESTORED IMAGES IN DOWNSTREAM COMPUTER VISION TASKS WHICH ARE VALIDATED ON DATASETS CONSISTING OF DIFFERENT NUMBER OF CLASSES. **TOP**: CLASSIFICATION ACCURACY OF VARIOUS IMAGE SETS ON TRAINED RESNET-18. **MIDDLE**: VIEW POINT ESTIMATION ERROR CALCULATED BY CONVEX OPTIMIZATION FRAMEWORK CAPTURING THE DEVIATION BETWEEN THE PREDICTED AND GROUNDTRUTH POSE IN VARIOUS DATASETS. **BOTTOM**: KEYPOINTS ESTIMATION ERROR OBTAINED BY USING THE HEATMAPS OF STACKED HOURGLASS NETWORK ON DIFFERENT IMAGE SETS

	Classification Accuracy								
	ResNet-18								
	Original (%)	Turbulent (%)	Pix2Pix [35] (%)	DeblurGAN (%)	Gao *et al.* [12] (%)	CycleGAN [66] (%)	Li *et al.* [46] (%)	DT-GAN+ (%)	DTD-GAN+ (%)
General	52.68	35.02	23.46	27.13	31.44	35.12	36.88	**45.66**	**45.81**
Categorical	67.53	27.48	26.74	26.77	27.51	28.03	32.52	**42.78**	**42.94**
Instance	91.53	77.69	76.15	78.05	79.08	79.23	80.77	**87.69**	**88.46**
	View Point Estimation Error								
	Convex Optimization Framework								
	Original (*degrees*)	Turbulent (*degrees*)	Pix2Pix [35] (*degrees*)	DeblurGAN (*degrees*)	Gao *et al.* [12] (*degrees*)	CycleGAN [66] (*degrees*)	Li *et al.* [46] (*degrees*)	DT-GAN+ (*degrees*)	DTD-GAN+ (*degrees*)
General	11.231	25.351	29.032	29.037	28.943	28.931	25.457	**23.201**	**22.392**
Categorical	9.671	16.365	22.233	22.101	21.619	21.011	16.715	**14.519**	**14.223**
Instance	11.445	27.124	31.391	30.997	29.847	29.732	25.567	**17.814**	**16.962**
	Keypoints Estimation Error								
	Stacked Hourglass Network								
	Original (*relative error*)	Turbulent (*relative error*)	Pix2Pix [35] (*relative error*)	DeblurGAN (*relative error*)	Gao *et al.* [12] (*relative error*)	CycleGAN [66] (*relative error*)	Li et al. [46] (*relative error*)	DT-GAN+ (*relative error*)	DTD-GAN+ (*relative error*)
General	0.000	341.249	366.657	361.113	355.781	352.945	338.374	**321.157**	**319.894**
Categorical	0.000	282.992	321.212	320.691	313.257	309.814	261.586	**240.454**	**239.284**
Instance	0.000	311.657	342.543	337.589	325.008	322.674	308.890	**264.787**	**256.561**

V. APPLICATION

In this section, we show improvement in downstream tasks: image classification, keypoint detection, pose, and depth estimation tasks using restored images. The restoration model's performance is evaluated by training a task-specific model with non-turbulent images and then validating the trained model performance on turbulent, restored, and original image sets. For the image classification task, we train a ResNet-18 [25] model. Table VIII shows more than 10% increase in classification accuracy for the restored images by DTD-GAN+ over

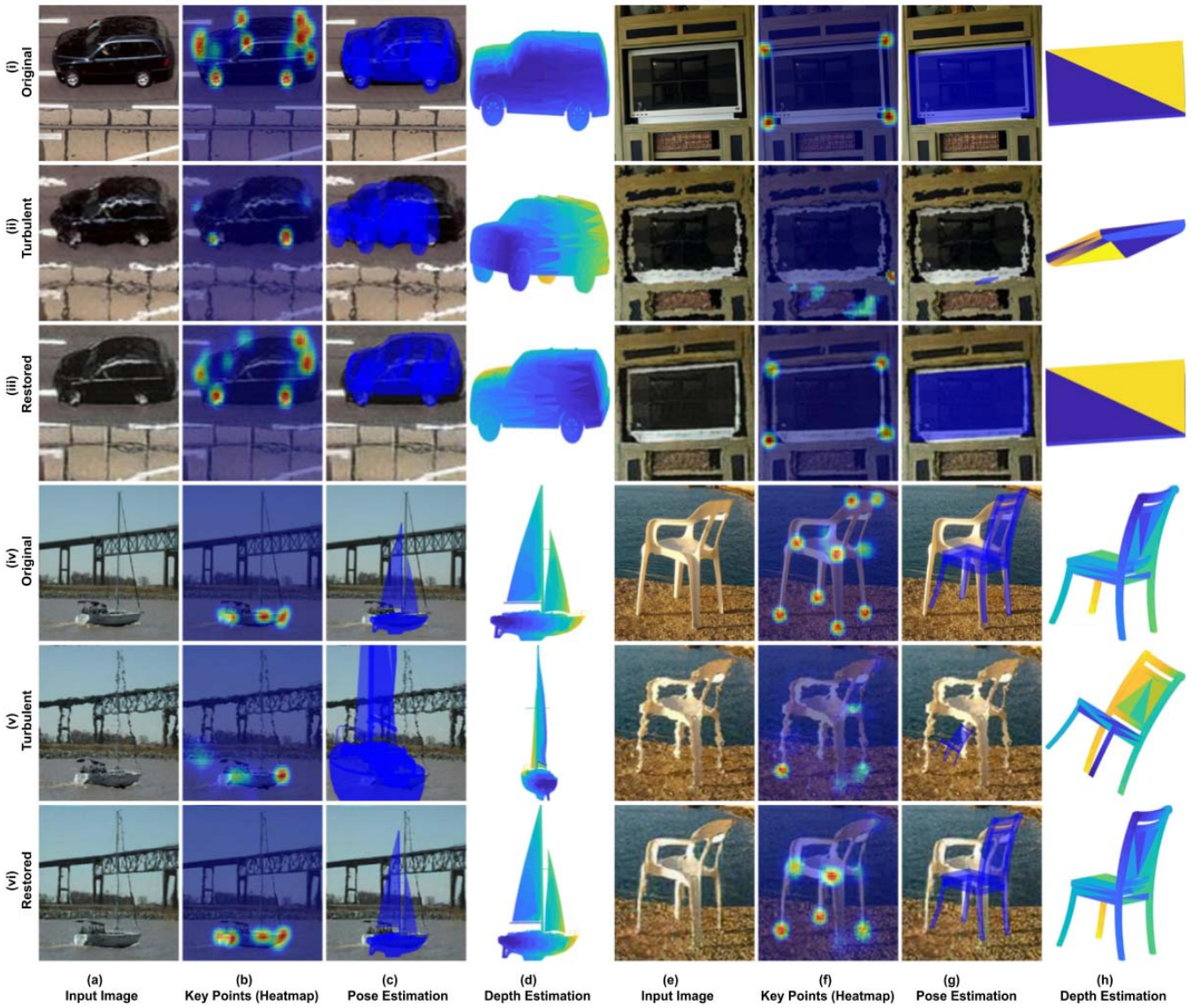

Fig. 13. **Column wise:** (a) and (e) are the input images to the network. (b) and (f) are the output heatmaps of the hourglass network overlapped on the input image. (c) and (g) are the CAD models projected on the input image. (d) and (h) are the depth visualization of CAD models. **Row wise:** (i) and (iv) are the results of original images. (ii) and (v) are the results of the turbulent images. (iii) and (vi) are the results of the restored images by DTD-GAN+. **(Best viewed when zoomed)**.

turbulent images across all the validation datasets. We also find that image artifacts in the restored images from Pix2Pix results in lower classification accuracy than turbulent images.

We trained Stacked Hourglass Network [70] for correctly estimating semantic key points in an image degraded due to atmospheric turbulence. We perform the evaluation process by taking the square absolute difference of output heatmaps containing probable keypoints area generated between non-turbulent and turbulent or restored images relative to the heatmaps predicted by non-turbulent images. In Figure 13 (column b and column f), we observe that the restored images by DTD-GAN+ can detect almost all the semantic key points that are undetected by the network for turbulent images. Quantitatively, there is more than a 5% decrease in the relative keypoint error for the restored images by DTD-GAN+ compared to turbulent images shown in Table VIII. We can observe that restored images from DeblurGAN, Gao *et al.* [12], CycleGAN, and Pix2Pix have higher keypoint estimation errors than turbulent images, as the geometrical distortions significantly persist in the restored images.

We use the methodology of Pavlakos *et al.* [71] to get the pose and depth of an object in a 2D image. In detail, this approach uses semantic keypoints locations on the 3D CAD model along with the key points of a 2D image predicted by an hourglass network and then employing these correspondences in a convex optimization framework that estimates the shape and pose of an object in the 2D image. For measuring the view estimation error between the poses, we use the geodesic distance $\delta(.)$ between an estimated pose, P1, and ground truth pose, P2, which is as follows:

$$\delta(P_1, P_2) = \frac{||\log(P_1^T P_2)||_F}{\sqrt{2}} \tag{5}$$

where F denotes the Frobenius norm. We show the qualitative results in Table VIII indicates that there is more than an 11% decrease in median viewpoint estimation error on restored images over turbulent images across all the datasets. However, the restored images by Pix2Pix and CycleGAN suffer from image artifacts that make identifying semantic keypoints difficult, resulting in higher viewpoint estimation error than turbulent images. In Figure 13 (column c-d and column g-h), we can visualize that the restored images pose and depth estimation by DTD-GAN+ are close to non-turbulent images. Finally, We conclude that our proposed restoration model DTD-GAN+ significantly improves computer vision task performances compared to other restoration models from all the above experiments.

VI. Conclusion

In this paper, we propose a deep adversarial network to minimize atmospheric turbulence. In comparison to the traditional restoration methods, our model neither utilizes any prior knowledge about the turbulence field nor does it combine multiple images to generate the restored image at test time. Our proposed restoration architecture achieves state-of-the-art performance by outperforming the general image-to-image translation models. We simulate a large dataset to train our model and show its generalization capabilities by inferring the trained model on simulated images over a range of atmospheric turbulence parameters and on images having natural atmospheric turbulence. Extensive experiments are also conducted for various tasks such as image classification, semantic keypoints detection, pose, and depth estimation, which helped improve various computer vision tasks. Additionally, the experiments are conducted on datasets from a thousand classes to a single instance to show that the restoration model can be generalized for any dataset size. Our research opens new directions in the relatively new and narrow area of restoring atmospheric turbulent images. It would be interesting to examine and extend the applicability of current methods to atmospheric turbulent videos.

References

[1] B. L. Ellerbroek, "First-order performance evaluation of adaptive-optics systems for atmospheric-turbulence compensation in extended-field-of-view astronomical telescopes," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 11, no. 2, p. 783, 1994.

[2] M. C. Roggemann, B. M. Welsh, and B. R. Hunt, *Imaging Through Turbulence*. Boca Raton, FL, USA: CRC Press, 2018.

[3] C. P. Lau, Y. H. Lai, and L. M. Lui, "Restoration of atmospheric turbulence-distorted images via RPCA and quasiconformal maps," 2017, *arXiv:1704.03140*.

[4] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling, "Efficient filter flow for space-variant multiframe blind deconvolution," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 607–614.

[5] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.

[6] K. Hornik, M. Stinchcombe, and H. White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Netw.*, vol. 3, no. 5, pp. 551–560, Jan. 1990.

[7] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, no. 2, pp. 246–257, Jun. 1991.

[8] X. Zhang, R. Jiang, T. Wang, P. Huang, and L. Zhao, "Attention-based interpolation network for video deblurring," *Neurocomputing*, vol. 453, pp. 865–875, Sep. 2021.

[9] S. Anwar and N. Barnes, "Real image denoising with feature attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3155–3164.

[10] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, Jan. 2005.

[11] W. H. Chak, C. P. Lau, and L. M. Lui, "Subsampled turbulence removal network," 2018, *arXiv:1807.04418*.

[12] J. Gao, N. Anantrasirichai, and D. Bull, "Atmospheric turbulence removal using convolutional neural network," 2019, *arXiv:1912.11350*.

[13] A. Schwartzman, M. Alterman, R. Zamir, and Y. Y. Schechner, "Turbulence-induced 2D correlated image distortion," in *Proc. IEEE Int. Conf. Comput. Photogr. (ICCP)*, May 2017, pp. 1–13.

[14] R. Ragazzoni, E. Marchetti, and G. Valente, "Adaptive-optics corrections available for the whole sky," *Nature*, vol. 403, no. 6765, pp. 54–56, Jan. 2000.

[15] M. Campbell, "Atmospheric turbulence and its influence on adaptive optics," Tech. Rep., 2009.

[16] D. L. Fried, "Probability of getting a lucky short-exposure image through turbulence," *J. Opt. Soc. Amer.*, vol. 68, no. 12, p. 1651, 1978.

[17] M. Aubailly, M. A. Vorontsov, G. W. Carhart, and M. T. Valley, "Automated video enhancement from a stream of atmospherically-distorted images: The lucky-region fusion approach," *Proc. SPIE*, vol. 7463, Aug. 2009, Art. no. 74630C.

[18] X. Zhu and P. Milanfar, "Image reconstruction from videos distorted by atmospheric turbulence," *Proc. SPIE*, vol. 7543, Jan. 2010, Art. no. 75430S.

[19] R. Hufnagel, "Restoration of atmospherically degraded images: Woods hole summer study," *Proc. Nat. Acad. Sci. USA*, 1966.

[20] Y. Yitzhaky, "Restoration of atmospherically blurred images according to weather-predicted atmospheric modulation transfer functions," *Opt. Eng.*, vol. 36, no. 11, p. 3064, Nov. 1997.

[21] X. Zhu and P. Milanfar, "Removing atmospheric turbulence via space-invariant deconvolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 157–170, Jan. 2013.

[22] Y. Xie, W. Zhang, D. Tao, W. Hu, Y. Qu, and H. Wang, "Distortion-driven turbulence effect removal using variational model," 2014, *arXiv:1401.4221*.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[28] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[29] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014.

[30] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*.

[31] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[32] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, "DRAW: A recurrent neural network for image generation," 2015, *arXiv:1502.04623*.

[33] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," 2016, *arXiv:1601.06759*.

[34] A. van den Oord *et al.*, "Conditional image generation with PixelCNN decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016.

[35] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," 2017, *arXiv:1611.07004*.

[36] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5485–5493.

[37] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4681–4690.
[38] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "DeblurGAN: Blind motion deblurring using conditional adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8183–8192.
[39] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3155–3164.
[40] Q. Yang *et al.*, "Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1348–1357, Jun. 2018.
[41] H. Gu, Y. Xian, I. C. Unarta, and Y. Yao, "Generative adversarial networks for robust cryo-EM image denoising," 2020, *arXiv:2008.07307*.
[42] Y. Ganin, D. Kononenko, D. Sungatullina, and V. Lempitsky, "DeepWarp: Photorealistic image resynthesis for gaze manipulation," in *Proc. ECCV*, 2016, pp. 311–326.
[43] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6148–6157.
[44] B. D. de Vos, F. F. Berendsen, M. A. Viergever, M. Staring, and I. Išgum, "End-to-end unsupervised deformable image registration with a convolutional neural network" in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Cham, Switzerland: Springer, 2017.
[45] M. Jaderberg *et al.*, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015.
[46] Z. Li, Z. Murez, D. Kriegman, R. Ramamoorthi, and M. Chandraker, "Learning to see through turbulent water," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 512–520.
[47] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip reading sentences in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3444–3453.
[48] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ICML*, 2015, pp. 2048–2057.
[49] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.
[50] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "BAM: Bottleneck attention module," 2018, *arXiv:1807.06514*.
[51] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, 2018, pp. 3–19.
[52] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[53] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 286–301.
[54] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11065–11074.
[55] K. Metwaly, X. Li, T. Guo, and V. Monga, "NonLocal channel attention for NonHomogeneous image dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 452–453.
[56] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," 2016, *arXiv:1607.08022*.
[57] Y. Wu and K. He, "Group normalization," 2018, *arXiv:1803.08494*.
[58] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.
[59] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
[60] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
[61] W. Shi *et al.*, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.
[62] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1865–1873.
[63] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2794–2802.
[64] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. ECCV*, 2016, pp. 694–711.
[65] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*.
[66] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond Pascal: A benchmark for 3D object detection in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 75–82.
[67] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," 2017, *arXiv:1703.10593*.
[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[69] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3883–3891.
[70] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. ECCV*, 2016, pp. 483–499.
[71] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis, "6-DoF object pose from semantic keypoints," in *Proc. ICRA*, 2017, pp. 2011–2018.

Shyam Nandan Rai received the bachelor's degree from IIIT Sri City in 2017. He is currently pursuing the master's degree in computer science with IIIT Hyderabad. His research interests include computer vision and deep learning for solving problems related to image restoration in turbulent weather conditions.

C. V. Jawahar (Member, IEEE) is currently the Amazon Chair Professor with the IIIT Hyderabad, India, where he leads a group focusing on AI, computer vision, and machine learning. His research interests include a set of problems that overlap with vision, language, and learning. He is a fellow of INAE and IAPR.