Munich to Dubai: How far is it for Semantic Segmentation?

by

Shyam Nandan Rai, Vineeth N Balasubramanian, Anbumani Subramanian, C V Jawahar

in

2020 IEEE Winter Conference on Applications of Computer Vision (WACV)
: 1
-10

Report No: IIIT/TR/2020/-1

Centre for Visual Information Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
March 2020

Munich to Dubai: How far is it for Semantic Segmentation?

Shyam Nandan Rai[1], Vineeth N Balasubramanian[2], Anbumani Subramanian[3], C. V. Jawahar[4]
[1,4]IIIT Hyderabad, [2]IIT Hyderabad, [3]Intel Corporation
[1]shyam.nandan@research.iiit.ac.in, [2]vineethnb@iith.ac.in,
[3]anbumani.subramanian@intel.com, [4]jawahar@iiit.ac.in

Abstract

Cities having hot weather conditions results in geometrical distortion, thereby adversely affecting the performance of semantic segmentation model. In this work, we study the problem of semantic segmentation model in adapting to such hot climate cities. This issue can be circumvented by collecting and annotating images in such weather conditions and training segmentation models on those images. But the task of semantically annotating images for every environment is painstaking and expensive. Hence, we propose a framework that improves the performance of semantic segmentation models without explicitly creating an annotated dataset for such adverse weather variations. Our framework consists of two parts, a restoration network to remove the geometrical distortions caused by hot weather and an adaptive segmentation network that is trained on an additional loss to adapt to the statistics of the ground-truth segmentation map. We train our framework on the Cityscapes dataset, which showed a total IoU gain of 12.707 over standard segmentation models. We also observe that the segmentation results obtained by our framework gave a significant improvement for small classes such as poles, person, and rider, which are essential and valuable for autonomous navigation based applications.

1. Introduction

In computer vision literature, the task of understanding the semantics of the scene is achieved by semantic segmentation. Formally, we define semantic segmentation as a method of classifying each pixel into its object category. Cityscapes [3] is one of the widely used datasets for training semantic segmentation models in an autonomous navigation based setting. The images of Cityscapes have been captured from road scenes from different cities of Germany, which have relatively colder and clear weather. Now, for instance, if we train a semantic segmentation model on the Cityscapes dataset and deploy it in places having extremely hot weather conditions such as Dubai, then the trained model finds hard to keep its optimal performance and give poor segmentation results, as shown in Figure 1(c) (even though the roads

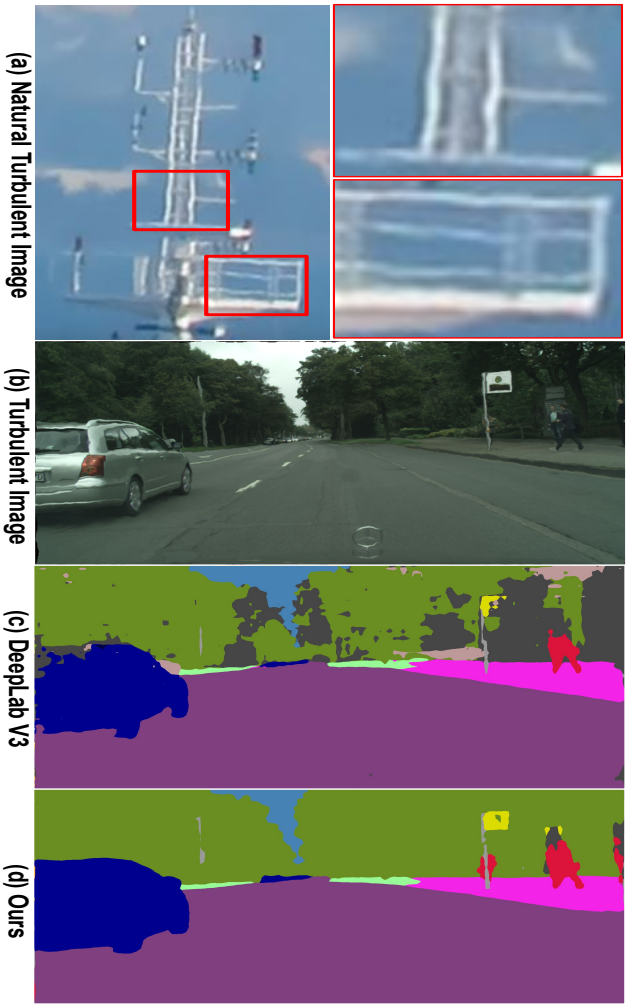

Figure 1: (a) Natural atmospheric turbulent images curated from the internet. The zoomed patches show the geometrical distortion caused by atmospheric turbulence. (b) A simulated atmospheric turbulent image of the Cityscapes dataset. (c)-(d) Performance of DeepLabV3 [2] and our proposed method on the turbulent image.

may look similar). This phenomenon happens due to the geometrical distortion caused by hot weather into the input image shown in Figure 1(a). Hence, it becomes necessary for us to ponder upon the problem of adapting semantic segmentation models in such weather variations due to the change in geographical location. This condition, especially variations caused by hot weather, is also referred to as atmospheric turbulence [44] as it affects the atmospheric parameters such as the refractive index between an object and a camera. In the remaining paper, we interchangeably use atmospheric turbulence and hot weather conditions for convenience. The problem of semantic segmentation model to generalize for hot weather can be bypassed by collected data, especially in such weather conditions and training a model on the collected images. However, collecting and annotating images for such atmospheric conditions is an extremely tedious task, which is time consuming and very expensive.

In this paper, we propose a solution to improve semantic segmentation model performance in hot weather without explicitly creating an annotated dataset. The proposed framework consists of two networks: Restoration network and Segmentation network. The restoration network is specifically intended to minimize the geometrical distortion caused by atmospheric turbulence in an image. We could have used existing machine learning methods [24, 37] for restoring images from atmospheric turbulence. But, these methods suffer from two significant limitations: (i) none of the methods works for single image restoration, and (ii) considerable variation in an atmospheric parameter cannot be handled by these methods. To overcome these issues, we train our restoration model on a large scale dataset, having images with varying atmospheric parameters for better generalization. At inference time, the trained restoration network can perform single image restoration. The architecture of our restoration network is adapted from the widely used image-to-image translation network [13]. Additionally, we introduce Channel Attentive Multi-Scale Residual Block (CA-MSRB), which learns local multi-scale features along with the inter-dependencies between residual channels using an attention mechanism.

The restored images obtained from the restoration network are passed to the segmentation network. The segmentation network consists of a DeepLabV3 [2] model, which is trained on multi-class cross-entropy loss between segmentation colormap of the restored image and ground-truth segmentation colormap. To make our semantic segmentation model more adaptive to the turbulent environment, we additionally use CORAL loss [30] between the restored image segmentation colormap and the non-turbulent image segmentation colormap got from pre-trained DeepLabV3 model. By using the additional loss, there is further improvement in segmentation results, and the domain gap between restored and non-turbulent segmentation colormap reduces. Our method shows significant improvement in segmentation results on the Cityscapes dataset, particularly for small classes (Figure 1) like poles, person, and, rider which are essential and valuable classes in autonomous navigation.

Our Contributions:

- We propose an adaptive semantic segmentation framework, which shows significant improvement in segmentation accuracy in hot-weather conditions. This framework bypasses the tedious task of semantic annotation on turbulent images.
- We use CORAL loss [30], as an additional loss to train our semantic segmentation network, which improves the segmentation accuracy and reduces the domain gap. Extensive experiments were conducted on Cityscapes [3] dataset to show improvement in segmentation accuracy, particularly for small classes.
- We proposed a restoration network for removing atmospheric turbulence in the images. Further, we also improve the restoration capabilities of our network on multi-scale, by introducing CA-MSRB block, which achieves state-of-the-art performance over the general image-to-image translation methods.

2. Related Work

Restoration In Atmospheric Turbulence: Removing the phenomena of atmospheric turbulence from images has been studied from the past few decades. Initial methods used adaptive optics [24], which were purely motivated by optics. These methods required precise experimental set-up, which was mainly used for astronomical applications. Lucky imaging [6] was another widely used method that relied on the probabilistic approach to restore images. Multi-frame image restoration approaches [1, 43] by Lucky imaging has also been proposed for enhancing the images and videos degraded from turbulence by correcting the geometrical distortion and reducing the blur present in the images. Frequency-based methods such as Fourier analysis [40] were also used to restore the images.

Recent methods have started using a machine learning approach to recover images from atmospheric turbulence. Zhu *et al*. [44] proposed a restoration method that first suppresses the geometrical distortion of each frame by using B-Spline built on non-rigid registration. After that, an image is generated from the set of registered images by using a temporal regression process. This regression process can also be viewed as the convolution of images with space invariant near-diffraction-limited blur. At last, the final output is produced by applying blind deconvolution on the regression output. However, this approach suffered from a significant limitation from the use of temporal mean to calculate

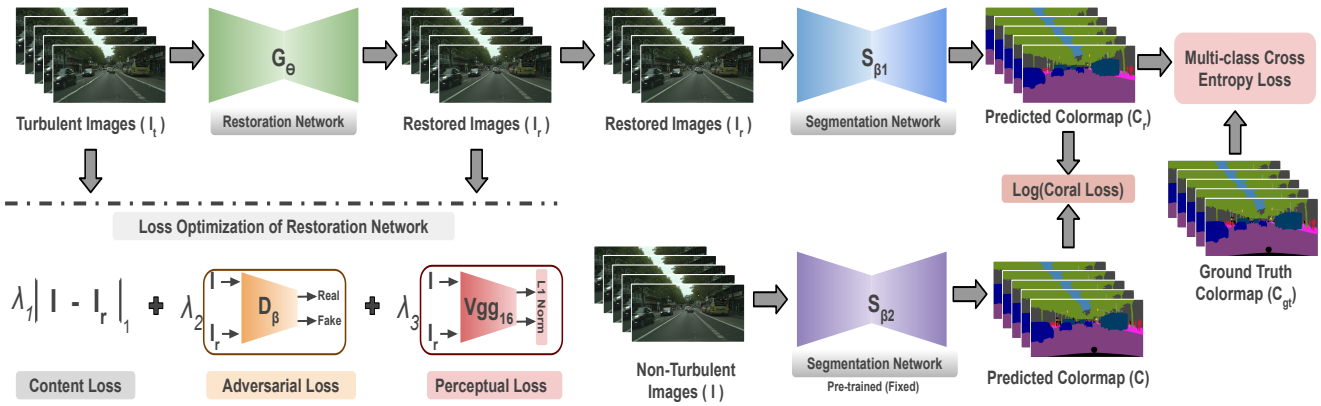

Figure 2: Overview of our framework: Our restoration network G_θ takes the input turbulent images I_t and gives the restored images I_r. To train the parameters of restoration network θ, a linear combination of losses is minimized between the restored and non-turbulent images. The restored images are further fed into the segmentation network $S_{\beta 1}$, which predicts the segmentation colormap C_r. The parameters of segmentation network $\beta 1$ is trained by calculating multi-class cross-entropy loss between C_r and ground-truth segmentation colormap C_{gt}. Additionally, we take the logarithm of CORAL loss between C_r and the predicted segmentation colormap C, which acts as an additional loss to train $\beta 1$. The segmentation colormap C is obtained by passing non-turbulent images I into pre-trained segmentation network $S_{\beta 2}$, with fixed parameters $\beta 2$.

the reference image, which lead to poor image registration. Xie *et al*. [38] proposed a method in which they overcome this limitation. This method first constructs a reference image using low-rank matrix decomposition on a set of input frames. And, for the registration process, they used a variational framework with a spatiotemporal regularization which iteratively optimizes the reference image.

However, none of the methods discussed above can be used to restore a single turbulent image as they require multiple turbulent images for restoration. Hence, we overcome this drawback by proposing a deep learning-based restoration model. Our restoration model requires only a single turbulent image to restore itself.

Image-to-Image Translation Via Deep Learning: Recent advancements in deep learning have drastically improved the performance in vision problems, such as classification [11], segmentation [19], and detection [25]. Another powerful property of deep networks is that they can learn to generate high dimensional non-linear data, such as images and audio using generative models [31, 23]. Among all the generative models, Generative Adversarial Network (GAN) [8] is the most successful model, which is widely used in image super-resolution [16], image impanting [39], and image-to-image translation [13].

However, general image-to-image translation model such as PixelRNN [32], Pix2Pix [13] and CycleGAN [42] learns the general mapping from one distribution to another. This restricts the general image-to-image translation model to leverage the specific problem structure, which could be effectively used for removing atmospheric turbulence. We overcome this problem by proposing an image-to-image translation which is specially intended to remove atmospheric turbulence. Recently, Wu *et al*. [36] proposed a method motivated from image stylization [7] to transfer an image from one weather condition to another. They further show improved semantic segmentation results on the styled image. However, their method is not modeled to handle significant geometrical changes in an image. Whereas, our method is specifically intended to work in geometrical distortion caused by extreme atmospheric turbulence.

Channel Attention for CNNs: Channel attention can be viewed as attending selectively on a specific part of the entire information while ignoring the rest of the information. In the context of CNNs, it can be interpreted as assigning selective weights to a feature map of a convolutional layer rather than giving equal weights to all feature maps. J. Hu *et al*. [12] introduced the concept of channel attention in CNNs. Later, this concept further extended into various vision applications, such as super-resolution [41], pose estimation [29], and action classification [4]. Motivated by wide applicability of channel attention in various vision applications, we build a CA-MSRB for use in our restoration network.

3. Our Approach

In this section, we describe the formalization of our proposed restoration model, followed by an improved segmentation model. The formalization of our restoration framework begins with a dataset consisting of turbulent images $I_t = \{I_t^i : i = 1...n\}$ and their corresponding non-turbulent

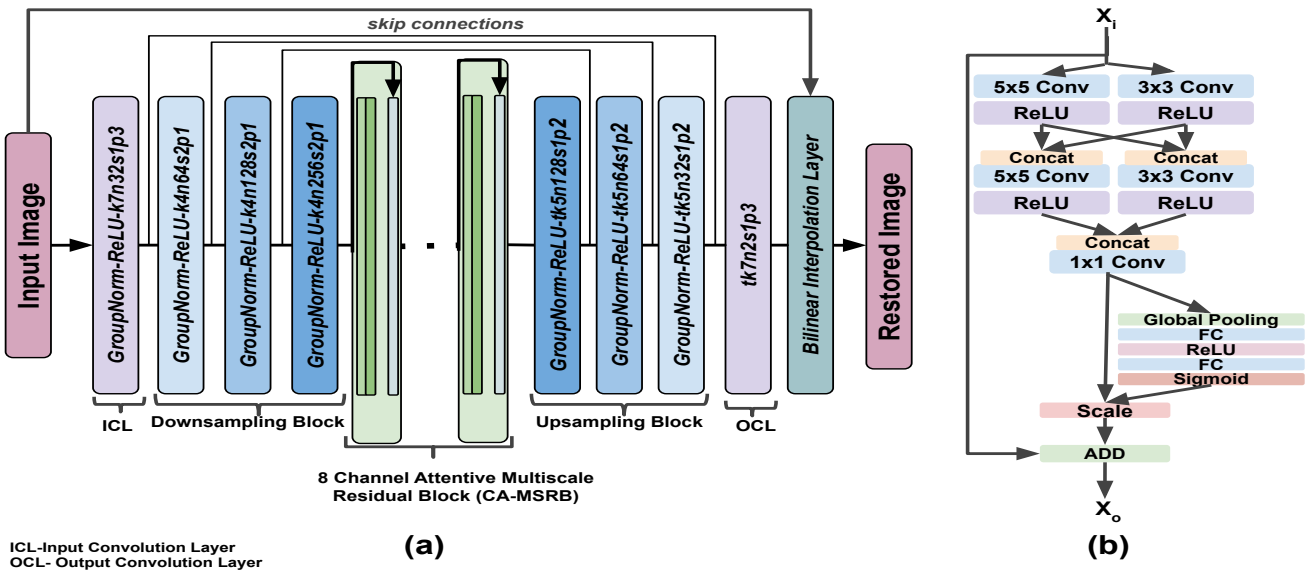

Figure 3: (a) Shows network architecture of our restoration network. The network takes the turbulent image and outputs the corresponding restored image. In the figure, k is the kernel size, n is the number of feature maps, and s is the stride in each convolutional layer with p as padding. (b) Architectural details of proposed Channel Attentive Multi-Scale Residual Block (CA-MSRB) used in the restoration network.

images $I = \{I^i : i = 1...n\}$, where $I^i \in \mathbb{R}^{N \times M}$ and n is the total number of samples in the dataset. Then, I_t^i is passed through the restoration network G_θ having learnable parameters θ. The output of G_θ is the restored image I_r^i of corresponding turbulent image I_t^i. The loss between I_r^i and I_i is used to train the restoration network parameters. After the restoration of turbulent images, we pass the restored images to the segmentation framework. The segmentation framework consists of two input heads. The first input head takes the restored image I_r^i, which is passed through trainable semantic segmentation network $S_{\beta 1}$ with trainable parameters $\beta 1$. Another input head takes the corresponding non-turbulent image I^i of I_r^i, which is passed through a pretrained network $S_{\beta 2}$ with fixed parameters $\beta 2$. Thereafter, we minimize the second-order statistics between the predicted segmentation map C_r^i of $S_{\beta 1}$ and C_i of $S_{\beta 2}$ which makes the C_r^i to adapt the domain statistics of C_i. Figure 2 shows an overview of our approach. In subsequent subsections, we describe our restoration and segmentation networks emphasizing the architectural details and losses used in training the network.

3.1. Restoration Network

The architecture of our restoration network is motivated by, Ledig *et al*. [16] and Li *et al*. [18] with some significant architectural changes for better adaptation to our problem, which is discussed below in detail along with the loss functions.

Network Architecture: Our restoration network architecture consists of an input convolutional layer, upsampling block, Channel Attentive Multi-Scale Residual Block (CA-MSRB), downsampling block, and output convolutional layer. The input convolutional layer projects input image to feature space whose output after that passed through the downsampling block. The downsampling blocks are comprised of 3 Group Normalization layer [35], 3 ReLU layer, and 3 convolutional layers. We perform Group Normalization rather than other normalization techniques as it gives lower training loss on smaller batch sizes. The configuration of each convolutional layer has a filter size of 4×4 with padding 1 and stride 2. After each convolutional layer in the downsampling block, the number of output features doubles. The downsampled features obtained from the downsampled block are passed through 8 CA-MSRB blocks.

The architecture of CA-MSRB is inspired by Hu *et al*. [12] and Li *et al*. [17]. The CA-MSRB consists of three parts: multi-scale atmospheric distortion learning, channel attention, and local residual learning. The multi-scale atmospheric distortion learning comprises of two bypass networks with convolutional kernel sizes of 5×5 and 3×3. The information between bypass networks is shared to facilitate the learning of atmospheric distortion at multiple scales. The output of the bypass networks is fused by a 1×1 convolutional layer. After fusion, we apply channel attention on the output of a 1×1 convolutional layer to capture channel-wise inter-dependencies. After that, the

learned weights are used to scale channel features learned by the 1×1 convolutional layer. At last, we perform local residual learning by adding input of CA-MSRB block to the channel attention output, so that the atmospheric distortion learned from the previous layers is effectively is passed deeper into the network. Figure 3(b) shows the architectural details of a CA-MSRB.

The output features obtained after all CA-MSRB blocks are then upsampled by upsampling blocks. The architecture of the upsampling block is similar to downsampling blocks, instead of the convolutional layer, it uses transpose convolution to upsample the features. The number of output features decreases from 256 to 32 across the upsampling block. Finally, we use the output convolutional layer to get the final warping field, which contains the flow movement of pixels displaced from its original position due to atmospheric turbulence. The field is warped bi-linearly applied to the input image to get the restored image. We also add skip-connection [26] to our restoration network to recover information lost during downsampling.

Restoration Loss Function: Our restoration network can be trained only on $L1$ loss (content loss), however, it results in overly smooth output images. To overcome this problem, we train our restoration network by minimizing a loss function consisting of a linear combination of content loss, perceptual loss [14], and adversarial loss [8]. Perceptual loss is used to add perceptually relevant characteristics into the output image. This loss is calculated by taking the $L1$ distance between the restored and non-turbulent images feature representation of Conv4_3 of VGG16 [28]. Lastly, we add adversarial loss to our loss function, so that, the output image lies in the natural image manifold. Our restoration network G_θ is used as the generator for our adversarial training. The architecture of the discriminator is adopted from DCGAN [25]. Also, we use spectral normalization [22] which stabilizes the training of discriminator networks. We use least-square loss function [21] in training the network which results in high-quality output image generation. The overall loss function of our restoration or generator network is:

$$l_{gen} = \lambda_1 ||I - G_\theta(I_t)||_1 + \lambda_2 [D_\beta(G_\theta(I_t)) - 1]^2 + \lambda_3 ||\psi_{4,3}(I) - \psi_{4,3}(G_\theta(I_t))||_1 \quad (1)$$

where, $\psi_{4,3}$ is the feature map of VGG16 at Conv4_3 layer output and λ_1, λ_2, and λ_2 are hyper-parameters that empirically estimated during training the network.The loss for discriminator is formulated as:

$$l_{disc} = [D_\beta(G_\theta(I_t))^2 + (D_\beta(I) - 1)^2] \quad (2)$$

also, we apply spectral normalization on each layer of discriminator, so that $||D_\beta||_{Lip} \leq 1$.

Method	PSNR	SSIM	MS-SSIM	MSE
CycleGAN [42]	22.3450	0.6597	0.9010	203.6862
Pix2Pix [13]	25.1881	0.7934	0.9563	146.6870
UNet [26]	25.9149	0.8042	0.9611	134.6829
Li *et al*. [18]	26.1525	0.8095	0.9631	131.7664
Ours	**26.6137**	**0.8120**	**0.9655**	**125.7047**

Table 1: Quantitative comparison of our restoration model with state-of-the-art image-to-image translation models. All the models were trained on the Cityscapes dataset. We can observe that our model outperforms the other general image-to-image translation models over all the image quality metrics.

3.2. Segmentation Network

The restored images I_r^i obtained as output from the restoration network are passed into $S_{\beta 1}$. The output of $S_{\beta 1}$ network is $C_r^i = S_{\beta 1}(I_r^i)$, which is the predicted colormap of restored image. The parameters $\beta 1$ of $S_{\beta 1}$ is trained by using multi-class cross-entropy loss function. Ideally, our predicted colormap of restored images C_r^i from $S_{\beta 1}$ should be equal to $C^i = S_{\beta 2}(I^i)$, the predicted colormap of non-turbulent image on pre-trained network. Hence, we use CORAL [30] as an additional loss function to further minimize the gap between C_r^i and C^i. CORAL loss is widely used in domain adaptation to match the second-order statistics of source and target distribution. In this problem, C_r^i can be considered as a sample from the source domain and C^i sample from the target domain. We take natural logarithm of the output of CORAL loss for better stability. The overall segmentation loss function can be formulated as:

$$L_s = l_{cross-entopy}(C_r^i, C_{gt}^i) + \gamma l_{coral}(C_r^i, C_i) \quad (3)$$

where, C_{gt}^i is the ground-truth segmentation colormap of input image I^i and γ is the hyper-parameter of L_s. For all our semantic segmentation experimentation, we use DeepLabV3 [2] as our semantic segmentation network to get predicted colormap of an image.

4. Experimentation

4.1. Experimental Settings

Dataset: Turbulent images can be simulated by using computer graphics [10]. But, these methods use high computational power for rendering. So, we use a physics-based method [27], which efficiently renders turbulent images by following a few simple 2D operations. We use pixel-level Cityscapes dataset to create our turbulent dataset, which was used in the experiments of our proposed framework. The synthesized dataset consists of 2975 training image pairs and 500 validation image pairs of turbulent and non-turbulent image pairs. We follow the evaluation method

Method	Dataset	road	swalk	build.	wall	fence	pole	tlight	sign	veg.	terrain	sky	person	rider	car	truck	bus	train	mbike	bicycle	mIoU
DeepLabV3 [2]	Non-Turbulent	97.34	82.94	91.24	65.36	69.53	41.58	56.01	66.03	89.47	70.63	91.57	74.94	58.68	91.20	81.71	86.89	88.19	66.89	68.99	75.746
DeepLabV3 [2]	Turbulent	94.22	66.13	71.98	25.77	18.27	23.14	30.39	37.51	71.15	46.99	86.87	44.06	15.43	64.26	32.93	47.57	19.79	15.65	29.41	44.291
DeepLabV3 [2]	Restored	95.07	69.5	81.65	26.92	26.59	32.04	39.21	42.42	85.07	53.47	87.94	58.03	36.45	85.47	54.61	64.03	24.21	33.79	48.65	55.006
MMD-DLV3	Restored	95.24	69.12	83.43	29.56	28.29	31.24	38.04	44.23	85.56	53.57	87.42	56.27	38.86	86.03	54.65	**64.78**	23.96	32.61	50.11	55.419
Coral-DLV3	Restored	95.25	69.30	82.92	29.17	26.04	31.61	**39.80**	**44.63**	85.79	53.79	87.99	57.76	38.28	85.95	**55.96**	63.79	26.45	33.82	50.75	55.739
Joint Coral-DLV3	Restored	**95.76**	**71.81**	**83.86**	**30.72**	**30.55**	**33.28**	39.29	43.59	**86.06**	**55.19**	**88.61**	**60.73**	**39.21**	**87.41**	54.54	63.09	**30.02**	**36.41**	**52.79**	**57.011**
IoU Gain		*1.54*	*5.68*	*11.88*	*4.95*	*12.28*	*10.15*	*8.90*	*6.08*	*14.92*	*8.20*	*1.74*	*16.67*	*23.78*	*23.15*	*21.61*	*15.53*	*10.23*	*20.76*	*23.39*	*12.707*

Table 2: Quantitative comparison of various semantic segmentation methods on the Cityscapes dataset. We compare the performance of DeepLabV3 on non-turbulent, turbulent, and restored (output images from our restoration method) dataset of Cityscapes. We see significant gain in IoU by using the restored images over turbulent images. The performance is further improved by using the Joint Coral-DLV3 model, which is a DeeplabV3 network jointly trained on CORAL loss and multi-class cross-entropy. We compare our Joint Coral-DLV3 model with MMD-DLV3 and Coral-DLV3, which are the DeepLabV3 model trained on MMD and CORAL as a loss function, respectively. Finally, we show the IoU gain by the Joint Coral-DLV3 method on restored images over the DeepLabV3 model trained on turbulent images.

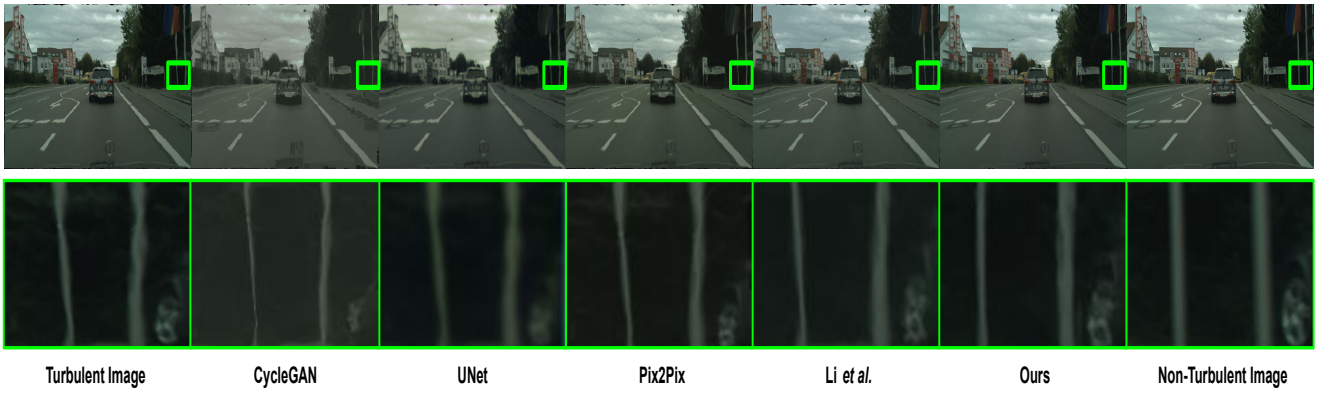

Figure 4: Qualitative comparison of our restoration network with other general image-to-image translation methods.

for semantic segmentation used in the Cityscapes dataset, where 19 semantic labels were used for evaluation without the void labels. We share all the parameter information for generating turbulent images in the supplementary material.

Implementation Details: We train our restoration network on the Cityscapes training image pairs of turbulent and non-turbulent images. The restoration network was trained for 15 epochs with a learning rate of $1e-4$ and batch size 8. For another 10 epochs, the restoration network was trained for a learning rate of $1e-5$. We use Adam [15] as the network optimizer with $beta_1$ = 0.5 and $beta_2$ = 0.999 for computing, running average of gradients and its squares. The value of λ_1, λ_2, and λ_3 in equation 1 are empirically found to be 100, 1 and 5, respectively. To train all the segmentation models, we largely employ the training protocol followed in DeepLabV3 [2]. The value of γ in equation 3 is empirically found to be 0.002.

Evaluation Metrics: We measure the structural and perceptual quality between images by Peak Signal-to-Noise-Ratio (PSNR), Structural Similarity (SSIM) [33], and Mean Square Error (MSE). Additionally, we use Multi-Scale Structural Similarity (MS-SSIM) [34] to measure the structural similarity between images at various resolutions. For semantic segmentation tasks, we use Intersection over Union (IoU) [3], which is a common evaluation metric among semantic segmentation methods.

4.2. Experimental Results

We compare our image restoration results with CycleGAN [42], Pix2Pix [13], UNet [26] and Li *et al*. [18] on Cityscapes test dataset. Table 1 and Figure 4 shows the qualitative and quantitative comparison of our model output with the other model. From Table 1, we can see that our model outperforms the other methods on all the image quality metrics. From the qualitative results observed in Figure 4, we notice that our model removes geometrical distortion caused by atmospheric turbulence as well as recovers the perceptual information. We can also see CycleGAN and Pix2Pix struggle to remove the geometrical distortion caused by turbulence. However, UNet and Li *et al*. eliminate the geometrical distortions to some extent but suffers from overly-smooth textures and color artifacts.

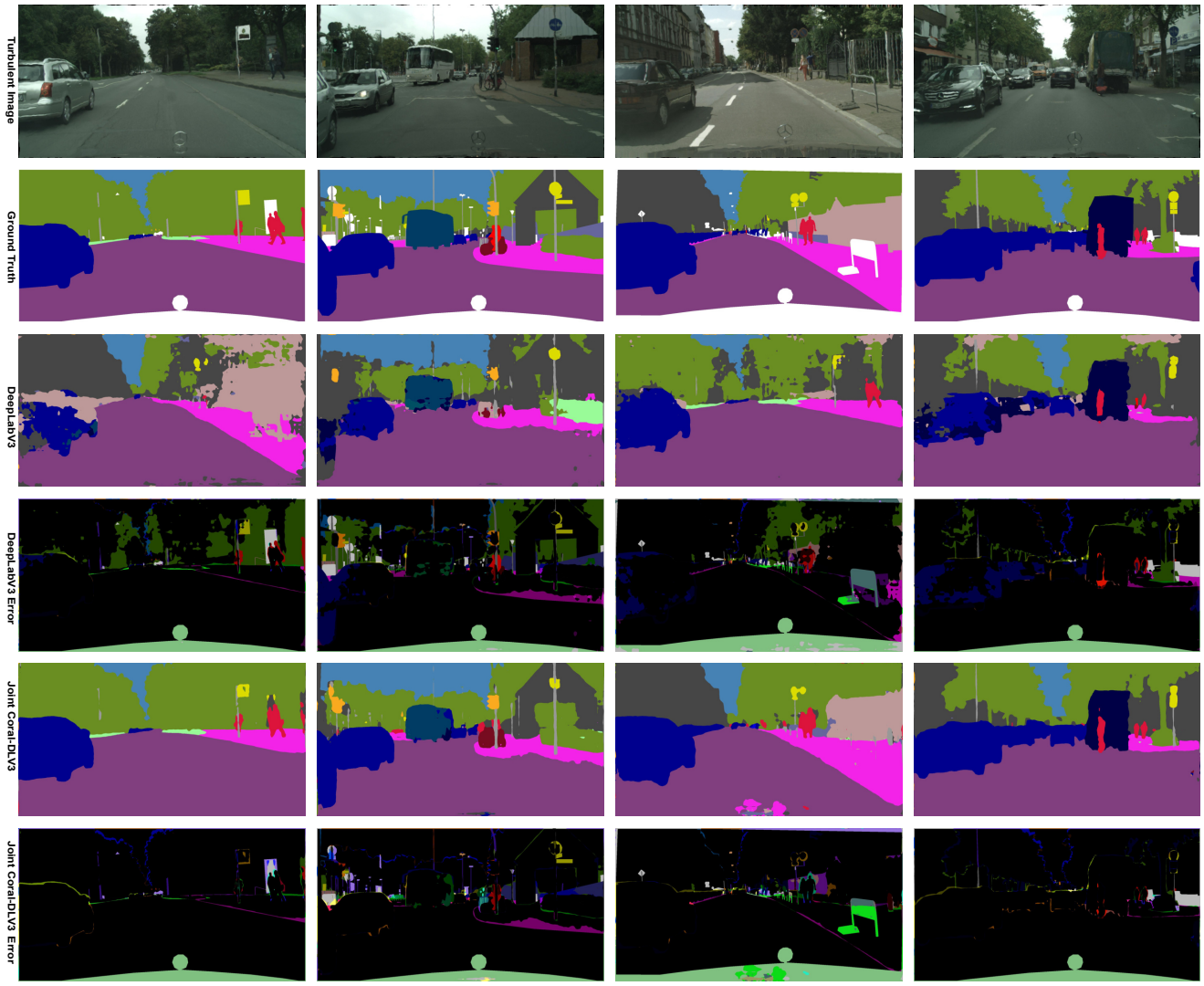

Figure 5: Shows the qualitative comparison of semantic segmentation results between DeepLabV3 and Joint Coral-DLV3. **Row 1:** The input turbulent image passed into the segmentation network. **Row 2:** Ground-truth segmentation colormap of the corresponding input image. **Row 3&4:** Predicted segmentation colormap by DeepLabV3 and its corresponding segmentation error from ground-truth. **Row 5&6:** Predicted segmentation colormap by Joint Coral-DLV3 and its corresponding segmentation error from ground-truth.

Table 2 shows the semantic segmentation results on non-turbulent, turbulent, and restored images of the Cityscapes validation dataset. MMD-DLV3 is the DeepLabV3 model trained on Maximum Mean Discrepancy (MMD) [20, 9] as a loss function. In the experiments, we used linear time MMD with a linear kernel. Coral-DLV3 and Joint Coral-DLV3 are the DeepLabV3 model trained on only CORAL and jointly by CORAL and multi-class cross-entropy as the loss function, respectively. From Table 2, we notice that the segmentation performance of the MMD-DLV3 was marginally inferior to Coral-DLV3, which shows that CORAL loss is better able to adapt the statistics of the ground-truth segmentation map.

We can observe our Joint Coral-DLV3 model on restored images making a total IoU gain of 12.707 over the results of the DeepLabV3 which is trained and validated on turbulent images. We also observe that the IoU gain is significant in classes such as bicycle, person, car, rider, and fence, which are more important and valuable classes than other classes like, sky for self-driving cars. Joint Coral-DLV3 gives a marginal improvement over individually training the network on CORAL loss or multi-class cross-entropy, which can be seen in Table 2. Figure 5 shows the semantic segmentation result of our model and DeepLabV3. We can

Multi-scale residual block	✗	✗	✓	✗	✓	✓	✗	✓
Spectral Normalization	✗	✓	✗	✗	✓	✗	✓	✓
Channel Attention	✗	✗	✗	✓	✗	✓	✓	✓
PSNR	20.1793	20.5455	21.7452	21.2314	22.0432	21.9523	21.8170	**22.1411**
SSIM	0.5765	0.5935	0.6319	0.6153	0.6495	0.6487	0.6473	**0.6517**
MS-SSIM	0.8057	0.8317	0.8759	0.8592	0.8889	0.8861	0.8836	**0.8910**
MSE	258.592	247.758	223.105	239.311	217.121	219.653	220.897	**214.686**

Table 3: Ablation investigation of multi-scale residual block, spectral normalization, and channel attention on our proposed restoration network. We find that combining all three components gave the best performance on all the image quality metrics. In the ablative investigation, we train all the models for 10 epochs.

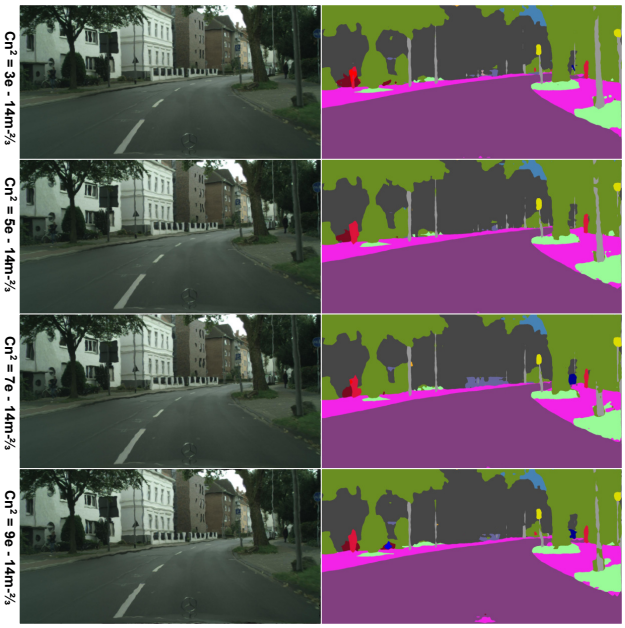

Figure 6: Semantic segmentation output of turbulent images on varying structure constant C_n^2. We can observe that there is a small degradation in the segmentation performance as the value of C_n^2 increases. **(Best viewed when zoomed.)**

notice that in the segmentation colormap that small object classes such as poles and pedestrians are segmented out by using our model on restored images.

4.3. Ablation Study

We perform our first ablation study on our restoration model by demonstrating the effectiveness of our residual block CA-MSRB, which is split into MSRB and channel attention block for ablative study. Additionally, we also show the effectiveness of spectral normalization in the discriminator. We train all our models on the Cityscapes dataset for 10 epochs. Table 3 shows the ablation investigation on the effects of using spectral normalization, MSRB, and channel attention block. We find that using MSRB of residual block improves the quality of output, which shows the advantage of multi-scale learning in MSRB. Then, we use CA-MSRB by combining MSRB and channel attention block showing further improvement. Lastly, we include the spectral normalization technique into our discriminator network along with CA-MSRB into the restoration model resulting in the best combination for the restoration model. We also show the effect of varying structure constant (Cn^2) [5] on semantic segmentation. Cn^2 is an important parameter for atmospheric turbulence, which measures the atmospheric refractive index and directly depends on atmospheric temperature. To get a wide variety of turbulent images, we change the Cn^2 while stimulating turbulent images using [27] method. Figure 6 shows the semantic segmentation results by Joint Coral-DLV3. We observe that as the value of Cn^2 increases, it becomes difficult to segment smaller classes such as poles.

5. Conclusion

In this paper, we have proposed a semantic segmentation framework which adapts to hot weather conditions and gives improved segmentation results over standard semantic segmentation network. Our framework circumvents the painstaking tasks of semantic annotation on turbulent images. The proposed framework works in two stages. In the first stage, we propose a restoration network specifically intended to remove geometrical distortion from turbulent images. Additionally, to improve the performance of our restoration network, we propose CA-MSRB block, which learns local residual at multi-scale along with interdependencies between the residual channels. In the next stage, the restored images were passed through an adaptive semantic segmentation model to give segmentation colormap. The segmentation results showed by our framework gave a significant improvement in small classes such as poles, person, and rider, which are more important and valuable classes for autonomous applications. Our work opens the possibility of improving semantic segmentation in other weather conditions such as rain, snow, and fog, which could be the next possible extension of this work.

References

[1] 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. In *Atmospheric Optics: Models, Measurements, and Target-in-the-Loop Propagation III*, volume 7463, page 74630C. International Society for Optics and Photonics, 2009.

[2] L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[4] A. Diba, M. Fayyaz, V. Sharma, M. Mahdi Arzani, R. Yousefzadeh, J. Gall, and L. Van Gool. Spatio-temporal channel correlation networks for action classification. In *ECCV*, 2018.

[5] D. L. Fried. Limiting resolution looking down through the atmosphere. *JOSA*, 56(10):1380–1384, 1966.

[6] D. L. Fried. Probability of getting a lucky short-exposure image through turbulence. *JOSA*, 68(12):1651–1658, 1978.

[7] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014.

[9] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *JMLR*, 2012.

[10] D. Gutierrez, F. J. Seron, A. Munoz, and O. Anson. Simulation of atmospheric phenomena. *Computers & Graphics*, 30(6):994–1010, 2006.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[14] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017.

[17] J. Li, F. Fang, K. Mei, and G. Zhang. Multi-scale residual network for image super-resolution. In *ECCV*, 2018.

[18] Z. Li, Z. Murez, D. Kriegman, R. Ramamoorthi, and M. Chandraker. Learning to see through turbulent water. In *WACV*, 2018.

[19] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[20] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. In *ICML*, 2015.

[21] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley. Least squares generative adversarial networks. In *ICCV*, 2017.

[22] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

[23] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[24] R. Ragazzoni, E. Marchetti, and G. Valente. Adaptive-optics corrections available for the whole sky. *Nature*, 403(6765):54, 2000.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[26] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[27] A. Schwartzman, M. Alterman, R. Zamir, and Y. Y. Schechner. Turbulence-induced 2d correlated image distortion. In *ICCP*, 2017.

[28] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[29] K. Su, D. Yu, Z. Xu, X. Geng, and C. Wang. Multi-person pose estimation with enhanced channel-wise and spatial information. In *CVPR*, 2019.

[30] B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, 2016.

[31] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al. Conditional image generation with pixelcnn decoders. In *NIPS*, 2016.

[32] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. *CoRR*, abs/1601.06759, 2016.

[33] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004.

[34] 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*, volume 2, pages 1398–1402. IEEE, 2003.

[35] Y. Wu and K. He. Group normalization. In *ECCV*, 2018.

[36] Z. Wu, X. Wang, J. E. Gonzalez, T. Goldstein, and L. S. Davis. ACE: adapting to changing environments for semantic segmentation. *CoRR*, abs/1904.06268, 2019.

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

[38] Y. Xie, W. Zhang, D. Tao, W. Hu, Y. Qu, and H. Wang. Distortion-driven turbulence effect removal using variational model. *CoRR*, abs/1401.4221, 2014.

[39] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with deep generative models. In *CVPR*, 2017.

[40] Y. Yitzhaky, I. Dror, and N. S. Kopeika. Restoration of atmospherically blurred images according to weather-predicted atmospheric modulation transfer function. *Optical Engineering*, 36, 1997.

[41] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018.

[42] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017.

[43] X. Zhu and P. Milanfar. Image reconstruction from videos distorted by atmospheric turbulence. In *Visual Information Processing and Communication*, volume 7543, page 75430S. International Society for Optics and Photonics, 2010.

[44] X. Zhu and P. Milanfar. Removing atmospheric turbulence via space-invariant deconvolution. *IEEE TPAMI*, 2012.