Turb-Seg-Res: A Segment-then-Restore Pipeline for Dynamic Videos with Atmospheric Turbulence

Ripon Kumar Saha[1], Dehao Qin[2], Nianyi Li[2], Jinwei Ye[3], Suren Jayasuriya[1]
[1]Arizona State University, [2]Clemson University, [3]George Mason University
{rsaha8, sjayasur}@asu.edu, {dehaoq, nianyil}@clemson.edu, jinweiye@gmu.edu

Abstract

Tackling image degradation due to atmospheric turbulence, particularly in dynamic environments, remains a challenge for long-range imaging systems. Existing techniques have been primarily designed for static scenes or scenes with small motion. This paper presents the first segment-then-restore pipeline for restoring the videos of dynamic scenes in turbulent environments. We leverage mean optical flow with an unsupervised motion segmentation method to separate dynamic and static scene components prior to restoration. After camera shake compensation and segmentation, we introduce foreground/background enhancement leveraging the statistics of turbulence strength and a transformer model trained on a novel noise-based procedural turbulence generator for fast dataset augmentation. Benchmarked against existing restoration methods, our approach restores most of the geometric distortion and enhances the sharpness of videos. We make our code, simulator, and data publicly available to advance the field of video restoration from turbulence: riponcs.github.io/TurbSegRes

1. Introduction

Atmospheric turbulence poses a considerable impediment to high-fidelity long-range imaging, inducing geometric distortions and blur that severely compromise image quality [7, 11, 49]. This issue becomes particularly acute in long-range horizontal and slant path imaging in environments with extreme temperature gradients. Such unpredictable variations in air density and velocity induce characteristic distortions including random pixel displacements, referred to as tilt [6], blur, and contrast reduction. This presents a formidable challenge for current vision algorithms to consistently detect, classify, and delineate objects within such degraded images.

Conventional multi-frame techniques such as "lucky frame" or geometric stabilization [13, 17, 18, 31, 33, 36] perform well in static or slightly dynamic scenes but struggle in highly dynamic environments with significant scene or camera motion. Supervised deep learning-based methods [38, 56] leverage turbulence simulators to generate training data [8, 37]. While these models show potential, they often exhibit temporal artifacts when applied to real-world dynamic video sequences with moving objects.

The limitation of previous methods for dynamic video stems from the compounded effect of object motion and atmospheric turbulence. These factors disrupt optical flow maps which makes image registration difficult for multi-frame techniques. In this paper, we address this challenge by proposing a unique segment-then-restore pipeline. This method utilizes integrated optical flows computed between frames for unsupervised motion segmentation, which then enables the individual processing of static background and dynamic foreground elements. We use a weighted image stacking technique for background enhancement, and a transformer model for both foreground and background sharpening trained on data from a novel tilt-and-blur turbulence simulator. Our approach notably reduces background distortions while maintaining the clarity of foreground details throughout the sequence.

Our specific contributions include:

- An unsupervised segmentation method that utilizes integrated optical flow with optimized number of frames calculated to segment a frame into static background and moving foreground.
- An adaptive Gaussian-weighted image stacking method for background processing which utilizes physics-based turbulence strength statistics for optimal frame selection across a range of turbulence intensities.
- An image restoration transformer model trained on simulated data generated from a novel turbulence video simulator that incorporates both tilt and adaptive blur based on procedural noise to generate plausible turbulence effects.

To validate our approach, we conduct experiments on two well-established datasets, CLEAR [2], OTIS [19]. We also augment the recent URG-T [45] segmentation dataset with

additional real videos captured in the wild that feature extreme turbulence conditions. We compare to state-of-the-art baselines for video restoration in this domain. Finally, we release the code, simulator, and data for our study: riponcs.github.io/TurbSegRes

2. Related Work

Modeling and Simulating Turbulence: The physics of turbulence was first detailed by Kolmogorov [27, 28], and we refer the reader to several reference books [25, 53] including imaging through turbulence [29, 49]. Ihrke et al. [24] and Gutierrez et al. [21] helped advance the understanding and simulation of refractive and atmospheric phenomena, respectively.

In addition to modeling, there has been concerted effort to simulate the effects of turbulence. Potvin et al. [43] presented a parametric model for simulating turbulence effects on imaging systems. Similarly, Repasi and Weiss [47] proposed a computer simulation of image degradations by atmospheric turbulence for horizontal views. Reinhardt et al. [46] developed computationally efficient techniques to simulate optical atmospheric refraction phenomena.

Instead of simulating 3D random turbulence fields, Schwartzman et al. [52] directly created 2D random physics-based distortion vector fields. A series of simulators that utilize Zernike coefficents have been developed [8, 37] culminating in a real-time phase-to-space simulation [9]. In our work, we do not utilize these physics-based simulators because they are targeted for single image creation, but develop a procedural noise-based turbulence video simulator for coherent turbulence effects.

"Lucky Image" Reconstruction Methods: Fried [15] introduced the concept of "lucky imaging" for astronomical imaging through turbulence, and these frames are typically short-exposure to minimize blur [4]. Lucky fusion combines sharp image patches to form the final diffraction limited image using a variety of methods [2, 3, 55, 61]. However, these techniques require a static scene with no motion for lucky imaging to be effective.

Contrasting prior work, Mao et al. [36] pioneered restoration for dynamic sequences leveraging optical-flow guided lucky imaging coupled with blind deconvolution. We avoid lucky image techniques in favor of a segment-then-restore framework for dynamic scene restoration.

Deep Learning for Turbulence Restoration: With the help of the synthetic datasets, several supervised neural networks have recently been proposed to address turbulence restoration. A complex-valued convolutional neural network was proposed [1] for reducing geometric distortions and subsequent refinement of micro-details. AT-Net utilized epistemic uncertainty in their network design for single frame reconstruction [56]. Mao et al. [38] introduced TurbNet, a physics-inspired transformer model for single frame restoration. Zhang et al. [60] also proposed an efficient transformer-based network but for video sequences.

In addition, there have been unsupervised models proposed in the literature. Li et al. [34] utilize implicit neural representations (INRs) to estimate grid deformation for single image restoration. Jiang et al. [26] extended this INR approach to have a more realistic tilt-and-blur model, but still targeted to static scenes.

Segmentation in Atmospheric Turbulence: Segmentation in atmospheric turbulence is relatively understudied in the literature. Cui and Zhang [12] developed a supervised network for semantic segmentation in turbulence, trained on synthetic datasets. Recently Qin et al. [45] proposed a two-stage unsupervised segmentation for the video affected by atmospheric turbulence, but is computationally expensive. Our segmentation, while conceptually simple, performs near state-of-the-art while having very low latency.

3. Method

Our method is driven by the observation that in dynamic scenes, moving objects and the static background experience distinct motion types: moving objects are influenced by their own motion plus turbulence and camera movements, while the static background is solely affected by turbulence and camera motions. We efficiently segment moving objects from the static background, even amidst turbulence, and apply targeted enhancement strategies to each.

In a nutshell, our full method, visualized in Fig. 1 performs the following steps: (1) video stabilization to reduce camera motion; (2) motion segmentation into dynamic foreground regions and static background; (3) weighted image stacking for the static background; (4) Poisson blending to merge the enhanced background with the dynamic foreground regions; and finally (5) sharpening the entire image using a trained transformer model.

3.1. Stabilization Methodology

Long range imaging with high focal length cameras are extremely sensitive to vibrations which leads to camera motion. Most existing video stabilization methods [48] rely on local feature detection/matching (e.g. SIFT, SURF, and ORB) prone to error due to turbulence-induced distortions. We instead implement a simple, yet effective method of estimating global camera motion via cross-correlation and applying the estimated translation to stabilize the frame.

Let $\mathbf{V} = \{\mathbf{V}_i\}$ be the sequence of video frames, each normalized to $[0, 1]$. We perform the following steps: (1) subtract the mean intensity of the entire video sequence from each frame: $\mathbf{V}'_i = \mathbf{V}_i - \mu$; (2) select the first frame $\mathbf{F}_{\text{ref}} = \mathbf{V}'_1$; (3) crop each frame as $\mathbf{V}^{(i)}_{\text{crop}}$ with a border of 50 pixels (representing the maximum camera shift possible) to avoid edge effects; (4) calculate the cross-correlation

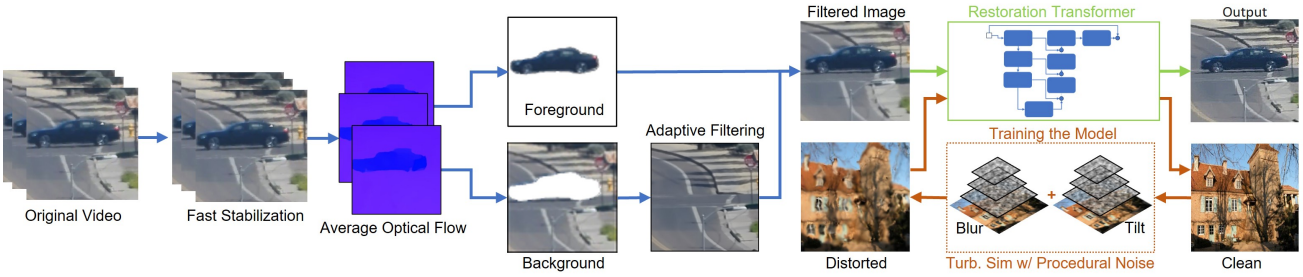


Figure 1. Our pipeline processes dynamic video frames by first stabilizing image sequences using normalized cross-correlation, followed by segmenting moving objects using average optical flow. The background is processed with adaptive filtering and blended with a separately extracted foreground. The background and segmented foreground are seamlessly merged using Poisson pyramid blending. Finally, a transformer architecture, trained on our simulator, refines the combined images.

(xcorr2d) of the cropped frame and the reference frame (implemented as a convolution in practice without flipping the kernel), and extract the coordinate positions of the maximum correlation:

$$\mathbf{H}^{(i)} = \text{xcorr2d}\left(\mathbf{F}_{\text{ref}}, \mathbf{V}_{\text{crop}}^{(i)}\right) \tag{1}$$

$$(\Delta x_i, \Delta y_i) = \arg\max(\mathbf{H}^{(i)}) - \left(\frac{H}{2}, \frac{W}{2}\right) \tag{2}$$

With the extracted $(\Delta x_i, \Delta y_i)$, we translate the selected frame via a matrix multiplication in homogeneous space to perform video stabilization.

This stabilization method, both simple and effective as demonstrated in the supplemental material, gains efficiency through our GPU-accelerated convolution approach, improving upon the CPU-based `scikit-image match_template` function in SciPy. This adaptation achieves a $1000\times$ speedup in the convolution phase, significantly enhancing processing speed for high-resolution images. Tested on videos with camera motion between 0 to 110 pixels and on $1080p$ synthetic video with max 400 pixels vibration, it ensured error-free vibration compensation.

3.2. Motion Segmentation in Turbulent Conditions

We introduce an automated motion segmentation technique that effectively discriminates between turbulence-induced distortions and inherent object motion for dynamic scenes. Our approach accounts for the Gaussian distribution nature [14, 16] of turbulence effects, which often makes capturing object movements challenging [6, 45].

The cornerstone of our method is a multi-frame optical flow analysis based on pre-trained RAFT [54]. Conventional frame-by-frame optical flow struggles in the face of turbulence. Our technique computes the average optical flow (AOF) across a dynamically determined number of neighboring frames, formulated as follows:

$$\text{AOF} = \frac{1}{N}\sum_{i=1}^{N} ||OF(F_{\text{current}}, F_i)||, \tag{3}$$

where F_{current} is the current frame, F_i denotes the i-th neighboring frame, and $||OF(\cdot,\cdot)||$ is the magnitude of the optical flow. The value of N is adaptively selected to enhance the segmentation clarity between static and dynamic regions. This optimization is achieved by maximizing the average distance of the normalized optical flow values from a median value of 0.5, ensuring a clear separation between static (near 0) and dynamic (near 1) elements:

$$N_{\text{opt}} = \arg\max_{N}\left(\frac{1}{M}\sum_{i=1}^{M}|\text{AOF}(N)_i - 0.5|\right), \tag{4}$$

where M is the total number of pixels in the optical flow mask, and $\text{AOF}(N)_i$, representing the optical flow magnitude value for the i-th pixel normalized to the range $[0, 1]$. This allows for an adaptive response to varying motion displacements within turbulent scenes.

Finally, we threshold our calculated AOF to separate the video into static background and dynamic background. While this segmentation technique is simple (leveraging average optical flow), we find it highly effective in practice as shown in Fig. 2, which displays segmentation results across three distinct frames of a video clip. We also conducted ablation on the choice of segmentation in Sec. 5.3 that justify this simpler method is competitive with state-of-the-art unsupervised methods [10, 35, 45, 57] for turbulent video while having lower latency.

3.3. Background Processing

For static backgrounds, we implement tilt correction via weighted averaging, guided by atmospheric turbulence strength as indicated by C_n^2 [51] values derived from video analysis. This method adapts Gaussian weighted averaging

Figure 2. Visual representation of motion segmentation across consecutive frames in a video sequence. Our proposed method effectively separates moving subjects (students walking) visualized by the golden outlines.

relative to detected turbulence, targeting precise stabilization and geometric distortion reduction in the background. C_n^2 can be calculated via image information via the following equation [51, 58]:

$$C_n^2 = \frac{PFOV^2 \times D^{\frac{1}{3}}}{L \times P} \times \frac{\sigma(\mathbf{V})^2}{Grad(\mathbf{V})}, \tag{5}$$

where $\mathbf{V}$ is the sequence of images, $PFOV$ is the pixel field of view, D is the lens aperture diameter, L is the distance to target, and P is the turbulence constant parameter, adjusted based on the scale of turbulence [58]. Lower C_n^2 values, indicative of minimal turbulence, necessitate smaller Gaussian windows as the distortion is negligible. Higher C_n^2 values signal stronger turbulence, requiring larger windows to average more frames for effective distortion correction. Simplified to $C_n^2 \propto \frac{\sigma(\mathbf{V})^2}{Grad(\mathbf{V})}$, this adaptive process culminates in an averaged background with notably reduced geometric distortion. It efficiently addresses a spectrum of turbulence strengths, facilitating optimal correction automatically without needing manual hyperparameter adjustments.

3.4. Video Restoration

After segmentation and background enhancement, our pipeline's final step is restoration, where we merge the dynamic foreground into the background using Poisson blending [41]. We train a transformer-based model to perform restoration for both the foreground and background simultaneously. This is the only supervised part of our machine learning pipeline, and thus requires a dataset of images without distortion paired with the corresponding turbulence-degraded images. Since real data is difficult to acquire such ground truth pairs, in the following subsections we describe our novel simulator based on simplex noise for rapidly generating data for our training. Then we proceed to describe the transformer model that is trained on this data.

3.4.1 Turbulence Video Simulator

There have been existing turbulence-distortion simulators introduced in the prior literature [5, 8, 20, 22, 23, 32, 44, 47, 50, 52]. However, these simulators are targeted mostly for single image generation, and thus suffer from coherency or temporal artifacts when applied frame-by-frame for video. Recent hybrid or physics-based simulators show promise [8, 37], however, we find their latency restrictive when generating varying levels of turbulence for each image for fast data creation[1].

Tilt Simulation: Our approach is tailored to model turbulence for image restoration purposes, focusing on creating plausible turbulence warp/tilt and blur effects rather than simulating turbulence with high physical accuracy. By employing 3D simplex noise, a procedural noise function from computer graphics for creating textures and volumetric effects [40], we generate temporally coherent distortions. This method offers better scalability for high-dimensional noise and low latency and memory requirements, making it ideally suited for fast data creation for training models. The utilization of simplex noise reflects our focus on plausible video distortions over precise physical modeling of turbulence [30]:

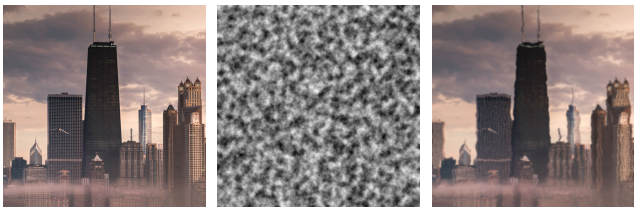
$$s(x, y, t) = \sum_{i=0}^{N-1} 2^i A_i \cdot \text{snoise}(f_i \cdot x, f_i \cdot y, f_i \cdot t) \tag{6}$$

Where A_i is the amplitude that scales the influence of each octave and f_i is the frequency that dictates the granularity of the noise pattern. Utilizing an $N = 8$ octave simplex noise method in Eq. (6), we generate two sets of 3D noise, $\mathcal{N}_x(H, W, T)$ and $\mathcal{N}_y(H, W, T)$ where H is height, W is weight, T is time, to simulate turbulence in the X and Y directions, respectively. This structure ensures that each pixel (x, y) in the input image is dynamically shifted according to these noise sets across frames, effectively mimicking atmospheric turbulence tilt over time. Employing the snoise function[2] ensures spatial and temporal coherence. With frequency ranges (f_i) from 1.5% to 6%, our approach simulates a spectrum of turbulence effects from mild to intense, ensuring qualitative fidelity to advanced models with operational simplicity and efficiency.

Blur Simulation: Adopting a similar approach to tilts, we utilize a set of 3D Perlin noise [42] to impose atmospheric blurring on tilt-adjusted frames. Generating Perlin noise with parameters such as width, height, depth, base frequency, and a level count of 11 allows dynamic modulation of blur intensity. This ensures spatial and temporal coherence with tilt magnitudes, enhancing realism. Adaptive Gaussian blurring adjusts the blur (σ) based on the 3D Perlin noise map and tilt map values, effectively simulating the variability of atmospheric scattering. This results in frames

[1] At the time of publication, a real-time physics-based simulator code implementation [9] was made available that could also potentially be used for training our transformer

[2] We utilize the Github package https://github.com/caseman/noise



(a) Original (b) Noise (c) Turbulence

Figure 3. Comparison of the original image, noise-added image, and the image with simulated turbulence.

exhibiting authentically-varied blur levels, contributing to the simulation's overall authenticity.

Simulation Results and Comparisons: Our tilt-and-blur video simulator, capable of processing 200×200 resolution videos at interactive rates (> 14 fps) on a consumer desktop, sets a new standard for efficiency and flexibility in turbulence simulation. Unique in its ability to produce coherent time-dependent distortions, it generates frames in just 100ms without precomputation, enabling the rapid creation of 100K short clips at up to 1024×1024 resolution for transformer model training. This performance surpasses general-purpose simulators and demonstrates substantial advantages over methods like the PS2 simulator [8], particularly in generating training data with realistic temporal dynamics and without extensive pre-computation. A sample result is illustrated in Fig. 3, with qualitative comparisons to a physics-based simulator shown in supplementary materials.

3.4.2 Training the Transformer Model

Our pipeline is compatible with any supervised deblurring model that can be trained from scratch on our procedurally generated, simulated turbulence data. For the final version of our pipeline, we leverage the image restoration transformer model, Restormer [59], due to its specific advantages in handling the complexities of turbulence-induced distortions. The Restormer model is particularly adept at dealing with the unpredictable and diverse nature of turbulence, thanks to its efficient attention mechanism that can focus on different regions of an HD image with varying distortion levels. Additionally, its capability to preserve and restore high-frequency details is crucial in maintaining the integrity of fine features in HD images, which are often adversely affected by turbulence.

4. Datasets and Implementation

4.1. Datasets

CLEAR Dataset [2]: The CLEAR dataset is a common benchmark for atmospheric turbulence mitigation [2]. The data includes (1) eight sequences capturing a variety of objects 3.5m away with turbulence induced by a set of 8 gas stoves, and (2) three scenes captured outdoors at various long distances with actual atmospheric turbulence. The sequences were captured with a Canon EOS-1D Mark IV camera with 105mm lens for (1) and 400mm lens for (2).

OTIS Dataset [19]: The OTIS dataset includes image sequences specifically designed for atmospheric turbulence mitigation studies. It features (1) controlled sequences with synthetic turbulence affecting various objects, generated through heat sources, and (2) natural scenes captured under real-world atmospheric turbulence. These images were captured with high-quality cameras, offering diverse scenarios to test turbulence mitigation techniques.

Augmented URG-T Dataset [45]: The URG-T dataset [45] consists of 20 videos recorded in outdoor environments at long-range with corresponding ground truth masks for motion segmentation [45]. Scenes include urban outdoor scenes including moving vehicles, airplanes, and pedestrians. Each clip features up to 56 frames at 1920×1080 resolution, shot with a Nikon Coolpix P1000, utilizing a 539mm focal length (effectively 3000mm due to sensor crop).

The URG-T dataset [45] lacks scenes with static backgrounds, essential for testing our restoration pipeline. To compensate, we added a collection of static videos capturing various atmospheric turbulence levels, using a Nikon Coolpix P1000 with $125\times$ zoom. We filmed additional scenes in a desert under summer conditions, with subjects placed 100 meters to 1 kilometer away, at $1080p$ resolution.

4.2. Training Details

We run our method on an NVIDIA A100 GPU, using the AdamW optimizer with a 0.0003 initial learning rate for up to 60,000 iterations. Batch and patch sizes dynamically adjust, ranging from 28 to 4 and 128 to 384 pixels, respectively, following a predefined schedule. Training is completed approximately in one day.

4.3. Comparison to State-of-the-Art

We compare our method to the following state-of-the-art methods for turbulence restoration:

1. AT-Net [56]: Deep learning framework that employs epistemic uncertainty analysis for effective restoration of images affected by atmospheric turbulence.
2. TCI 2020 [36]: Physics-based models that integrates space-time non-local averaging for enhanced turbulence mitigation in both static and dynamic sequences.
3. TurbNet [38]: Transformer-based model designed for atmospheric turbulence imaging, adept at extracting dynamic distortion maps and restoring turbulence-free images.

4.4. Latency

Our final pipeline's latency per frame for each stage is listed in Tab. 1 for 100%, 50%, and 25% of $1080p$ resolution. The main bottleneck is the calculation of the AOF and corresponding segmentation, due to pre-trained RAFT [54] and the forward pass through the sharpening transformer. For competing methods, the latency per frame for $1080p$ resolution on a NVIDIA RTX 3090 GPU is: TCI - 4200s; AT-Net - 80s; Turbnet - 5s, while our pipeline latency per frame is 5.71s on a NVIDIA A100 GPU.

Table 1. Per frame latency for operations at different resolutions

Operations	Size (pixels)		
	1920x1080	960x540	480x270
Read/Convert	0.08s	0.08s	0.08s
Vibration Calc.	0.03s	0.014s	0.006s
Stabilize	0.12s	0.05s	0.03s
Segmentation	1.06s	1.06s	1.06s
Gaussian Mean	0.34s	0.09s	0.02s
Combine BG/FG	0.38s	0.11s	0.026s
Sharpening	3.7s	1.0s	0.25s
Total	5.71s	2.404s	1.472s

4.5. Quantitative Metrics

For evaluating performance on the CLEAR dataset with available turbulence-free images, we employ standard metrics such as PSNR and SSIM, alongside IoU metrics to assess our pipeline's segmentation efficacy against URG-T's benchmarks [45].

Line Deviation Metric: For real-world scenarios lacking ground truth, we introduce a novel metric focusing on a system's capacity to preserve the integrity of straight lines during image restoration under turbulence. This metric, rooted in the observation that accurate restoration should correct distortions affecting straight structural elements (e.g., building edges), leverages Canny edge detection and the Probabilistic Hough Line Transform [39] to quantify angular deviations from true vertical/horizontal orientations. A reduced mean deviation score per image indicates superior performance in maintaining geometric fidelity in turbulent conditions, with our method's effectiveness quantified through rolling mean and standard deviation across sequences.

5. Experimental Results

5.1. Qualitative Comparison

Visual Results on Real Turbulence Video: In Fig. 4, we showcase zoomed in regions from single frames of two video sequences captured as part of the URG-T dataset. AT-Net and Turbnet are single-frame deep learning approaches that, while effective in some respects, introduce a significant number of artifacts into the processed images (see exaggerated sharpening effects on the edges of objects). TCI[3], is a multi-frame approach that excels at removing distortion but struggles with moving objects, e.g. the missing moving car in the first region or distortion in the second car.

Our method, in contrast, maintains the integrity of several object edges, presenting them solidly with minimal distortion, and importantly, it also accurately renders moving objects like cars, preserving their visibility and detail. *We recommend viewing the supplemental video to closely evaluate our restoration performance across scenes.*

OTIS Reconstruction Results: In Fig. 5, we showcase two scenes from the OTIS dataset (each video containing 300 frames): (1) a star chart target imaged under turbulence, and (2) a house entrance with a fence partially occluding it. The input images exhibit significant distortion, yet our method manages to produce images with the sharpest edges and without artifacts. For instance, in the Door image, the staircases appear crooked in results from other methods, and the foliage is indiscernible behind the fence. In contrast, our method leverages all 300 frames to correct the alignment of the staircase, and a closer examination shows we uniquely render the fence visible across the images with a continuous straight line.

5.2. Quantitative & Qualitative Comparison

Temporal Consistency Comparison: Our Turb-Seg-Res method demonstrates remarkable stability in the time domain, effectively mitigating the common issue of pixel fluctuation or the "image dancing" effect typically observed in turbulent conditions. This stability is evident when visually comparing video sequences, as our approach maintains consistent pixel behavior across frames as shown in Fig. 6.

Table 2. In the CLEAR dataset evaluation, images smaller than 400×400 pixels are excluded due to the fixed size requirements of AT-Net (256×256) and Turbnet (400×400). Larger images were processed using a patch-based approach. When we included all images of clear dataset, we got an average PSNR of 28.09 and SSIM of 0.935.

Turb.	**PSNR (dB)**			**SSIM**		
	AT-Net	**Turbnet**	**Ours**	**AT-Net**	**Turbnet**	**Ours**
Low	22.91	18.86	**26.48**	0.737	0.572	**0.853**
Medium	23.37	18.99	**26.50**	0.770	0.589	**0.867**
High	22.62	19.15	**24.82**	0.759	0.588	**0.828**

CLEAR Dataset Comparison: We ran quantitative evaluation of our method on the CLEAR dataset in Tab. 2.

[3]Note that TCI reconstructs in grayscale per the original paper [36]. We also developed a color extension for the TCI method by running on color channels independently, which we show in the supplemental material.

Figure 4. Comparative visualization of turbulence mitigation methods: AT-Net and Turbnet introduce artifacts, failing to clearly define edges of the road, while TCI maintains edge integrity but blurs moving objects. Our method shows minimal distortion with clear depiction of both static and dynamic elements.

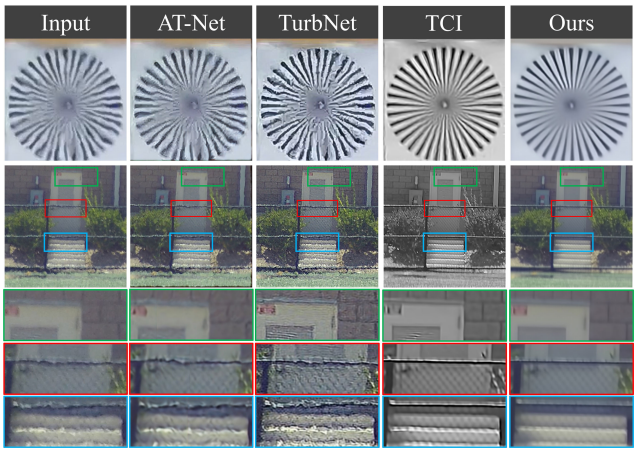


Figure 5. Analysis of the OTIS dataset reveals distinct outcomes. The upper row, featuring 'Pattern16' (300 frames, 135×135 pixels), demonstrates notable clarity in our method compared to AT-Net and TurbNet, which exhibit significant distortion in finer details. The lower row, showcasing 'Fixed Background (Door)' (300 frames, 520×520 pixels), highlights our approach's superiority in maintaining geometric integrity, especially in fence structures and the brickwork of stairs and doors. While TCI performs well on the fence, it distorts the circular patterns. Our method excels in preserving straight lines and enhancing edge definition in circular patterns, offering the clearest and least distorted results.

As one can see, our method achieved superior PSNR and SSIM values compared to competing methods, showing the benefit of our background processing and sharpening transform. Note that we do not need to perform segmentation as

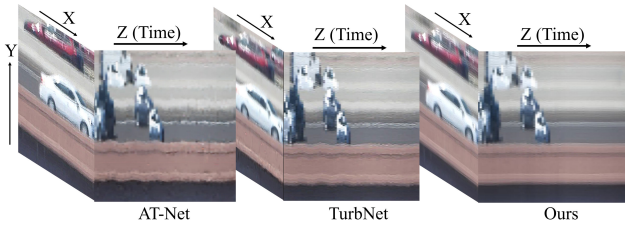


Figure 6. Demonstration of the Turb-Seg-Res method's stability in the time domain, highlighting its effectiveness in reducing pixel fluctuations in turbulent conditions.

there is no dynamic scene motion in the CLEAR dataset.

Line Deviation Metric Comparison: Fig. 7 reveals that for a single image sequence, our method attains a significantly lower average line deviation score than Turbnet and AT-Net. This quantitative metric, complemented by qualitative analysis in Fig. 4, confirms the superiority of our approach in preserving straight line integrity across frames. Specifically, we report a deviation score of Our: $1.89(\pm 0.39)$, outperforming AT-Net: $3.33(\pm 0.33)$ and Turbnet: $5.79(\pm 0.20)$. This underscores the enhanced geometric accuracy of our image restoration process.

5.3. Ablation Study

Importance of Segmentation: In Fig. 8, we highlight the relative importance of performing segmentation + background enhancement prior to restoration with our trained transformer. The pipeline with segmentation results in sharper background reconstructions on the static STOP sign while still maintaining fidelity for the moving car. Choos-

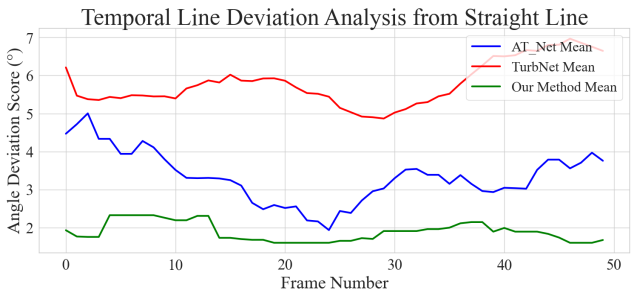


Figure 7. Probabilistic Hough Line Transform-based line deviation metric comparison. Our method exhibits minimal image distortion, particularly in maintaining straight lines, leading to a significantly lower line deviation score compared to competitors.

ing the optimal segmentation algorithm for our pipeline raises a crucial question. Despite the simplicity of our AOF method, an ablation study demonstrates its competitiveness with top unsupervised segmentation models in turbulence. In Tab. 3, we compare it against RGA [45], TMO [10], Deformable Sprites [57], and DS-Net [35], where it ranks second in performance but boasts significantly lower latency than the leading RGA method [45]. Note that creating precise ground truth masks is very challenging for human annotators due to the inherent fuzziness of edges in turbulence-distorted images, leading to potential nuisance factors during annotation.

Table 3. Mean IOU Scores for Different Segmentation methods on the URG-T dataset [45].

Video Name	RGA [45]	TMO [10]	DSNet [57]	DSprites [35]	Ours
Mean IoU	0.696	0.468	0.277	0.334	0.627

Importance of Stabilization: When applied across the entire CLEAR dataset [2], stabilization markedly improved our network's performance. The PSNR value rose from 26.80 to 28.09, and SSIM increased from 0.888 to 0.935, demonstrating stabilization's significant role in enhancing overall image quality.

Additional Results: The supplement provides ablation studies on optical flow choice, visualization of line deviation metrics, a comparison with the P2S simulator [8], qualitative video comparisons, analysis of restoration network choices, and the impact of stabilization and segmentation.

6. Conclusion

This paper presents the first segment-then-restore pipeline for dynamic videos with atmospheric turbulence to obtain enhanced visual reconstruction including the mitigation of geometric distortion as well as recovering high frequency sharpness. In addition to showing the value of segmentation for this problem, our paper introduces a novel tilt-and-blur simulator based on procedural noise (Simplex and

Figure 8. Comparative analysis of image restoration methods in atmospheric turbulence conditions. Panel (A) displays the outcome of a single-frame approach without segmentation, resulting in noticeable artifacts. Panel (B) exhibits the efficacy of our segmentation-based restoration pipeline, significantly enhancing the fidelity of the stop sign with minimal distortions.

Perlin) which enables large scale video dataset creation for training. Compared to state-of-the-art dynamic reconstruction methods, our pipeline is able to synthesize higher quality reconstructions with relatively low latency ($\approx$ 1-2 minutes per video) for $1080p$ resolution video. We release open-source code, simulator, and data available at this link: riponcs.github.io/TurbSegRes.

We acknowledge potential negative societal impacts, such as privacy concerns arising from enhanced long-range surveillance. We did conduct our research with an ASU IRB STUDY00018456 human subject research exemption.

Future work will explore enhancing segmentation accuracy with more annotated data, closing the gap between noise-based and physics-based simulation training, and developing real-time restoration for applications in robotics and embedded systems.

Acknowledgements

We wish to thank ASU Research Computing for providing GPU resources to support this research. This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via 2022-21102100003 and NSF IIS-2232298/2232299/2232300. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, NSF or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

[1] Nantheera Anantrasirichai. Atmospheric turbulence removal with complex-valued convolutional neural network. Pattern Recognition Letters, 171:69–75, 2023. 2

[2] Nantheera Anantrasirichai, Alin Achim, Nick G Kingsbury, and David R Bull. Atmospheric turbulence mitigation using complex wavelet-based fusion. IEEE Transactions on Image Processing, 22(6):2398–2408, 2013. 1, 2, 5, 8

[3] Mathieu Aubailly, Mikhail A Vorontsov, Gary W Carhart, and Michael T Valley. Automated video enhancement from a stream of atmospherically-distorted images: the lucky-region fusion approach. Atmospheric Optics: Models, Measurements, and Target-in-the-Loop Propagation III, 7463:104–113, 2009. 2

[4] David Bensimon, A Englander, R Karoubi, and Meira Weiss. Measurement of the probability of getting a lucky short-exposure image through turbulence. JOSA, 71(9):1138–1139, 1981. 2

[5] Jeremy P Bos and Michael C Roggemann. Technique for simulating anisoplanatic image formation over long horizontal paths. Optical Engineering, 51(10):101704–101704, 2012. 4

[6] Stanley H Chan. Tilt-then-blur or blur-then-tilt? clarifying the atmospheric turbulence model. IEEE Signal Processing Letters, 29:1833–1837, 2022. 1, 3

[7] Eli Chen, Oren Haik, and Yitzhak Yitzhaky. Detecting and tracking moving objects in long-distance imaging through turbulent medium. Applied Optics, 53(6):1181–1190, 2014. 1

[8] Nicholas Chimitt and Stanley H Chan. Simulating anisoplanatic turbulence by sampling intermodal and spatially correlated zernike coefficients. Optical Engineering, 59(8): 083101, 2020. 1, 2, 4, 5, 8

[9] Nicholas Chimitt, Xingguang Zhang, Zhiyuan Mao, and Stanley H. Chan. Real-time dense field phase-to-space simulation of imaging through atmospheric turbulence. IEEE Transactions on Computational Imaging, 8:1159–1169, 2022. 2, 4

[10] Suhwan Cho, Minhyeok Lee, Seunghoon Lee, Chaewon Park, Donghyeong Kim, and Sangyoun Lee. Treating motion as option to reduce motion dependency in unsupervised video object segmentation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 5140–5149, 2023. 3, 8

[11] EC Crittenden Jr, AW Cooper, GW Rodeback, SH Kalmbach, and RL Armstead. Effects of turbulence on imaging through the atmosphere. In Optical Properties of the Atmosphere, pages 130–134. SPIE, 1978. 1

[12] Linyan Cui and Yan Zhang. Accurate semantic segmentation in turbulence media. IEEE Access, 7:166749–166761, 2019. 2

[13] Hamidreza Fazlali, Shahram Shirani, Michael Bradford, and Thia Kirubarajan. Atmospheric turbulence removal in long-range imaging using a data-driven-based approach. International Journal of Computer Vision, 130(4):1031–1049, 2022. 1

[14] David L Fried. Statistics of a geometric representation of wavefront distortion. JOSA, 55(11):1427–1435, 1965. 3

[15] David L Fried. Probability of getting a lucky short-exposure image through turbulence. JOSA, 68(12):1651–1658, 1978. 2

[16] David L Fried. Anisoplanatism in adaptive optics. JOSA, 72 (1):52–61, 1982. 3

[17] Md Hasan Furhad, Murat Tahtali, and Andrew Lambert. Restoring atmospheric-turbulence-degraded images. Applied Optics, 55(19):5082–5090, 2016. 1

[18] Ronen Gal, Nahum Kiryati, and Nir Sochen. Progress in the restoration of image sequences degraded by atmospheric turbulence. Pattern Recognition Letters, 48:8–14, 2014. 1

[19] Jérôme Gilles and Nicholas B Ferrante. Open turbulent image set (otis). Pattern Recognition Letters, 86:38–41, 2017. 1, 5

[20] R Gray. Zernikecalc: a matlab function to work with zernike polynomials over circular and non-circular pupils, 2019. 4

[21] Diego Gutierrez, Francisco J Seron, Adolfo Munoz, and Oscar Anson. Simulation of atmospheric phenomena. Computers & Graphics, 30(6):994–1010, 2006. 2

[22] Russell C Hardie, Jonathan D Power, Daniel A LeMaster, Douglas R Droege, Szymon Gladysz, and Santasri Bose-Pillai. Simulation of anisoplanatic imaging through optical turbulence using numerical wave propagation with new validation analysis. Optical Engineering, 56(7):071502–071502, 2017. 4

[23] Bobby R Hunt, Amber L Iler, Christopher A Bailey, and Michael A Rucci. Synthesis of atmospheric turbulence point spread functions by sparse and redundant representations. Optical Engineering, 57(2):024101, 2018. 4

[24] Ivo Ihrke, Gernot Ziegler, Art Tevs, Christian Theobalt, Marcus Magnor, and Hans-Peter Seidel. Eikonal rendering: Efficient light transport in refractive objects. ACM Transactions on Graphics (TOG), 26(3):59–es, 2007. 2

[25] A. Ishimaru. Wave Propagation and Scattering in Random Media. Wiley, 1999. 2

[26] Weiyun Jiang, Vivek Boominathan, and Ashok Veeraraghavan. Nert: Implicit neural representations for unsupervised atmospheric turbulence mitigation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4235–4242, 2023. 2

[27] Andrey Nikolaevich Kolmogorov. Dissipation of Energy in Locally Isotropic Turbulence. Akademiia Nauk SSSR Doklady, 32:16, 1941. 2

[28] Andrei Nikolaevich Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers. Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences, 434 (1890):9–13, 1991. 2

[29] N.S. Kopeika. A System Engineering Approach to Imaging. SPIE Optical Engineering Press, 1998. 2

[30] Ares Lagae, Sylvain Lefebvre, Rob Cook, Tony DeRose, George Drettakis, David S Ebert, John P Lewis, Ken Perlin, and Matthias Zwicker. A survey of procedural noise functions. Computer Graphics Forum, 29(8):2579–2600, 2010. 4

[31] Chun Pong Lau, Yu Hin Lai, and Lok Ming Lui. Restoration of atmospheric turbulence-distorted images via rpca and quasiconformal maps. Inverse Problems, 35(7):074002, 2019. 1

[32] Kevin R Leonard, Jonathan Howe, and David E Oxford. Simulation of atmospheric turbulence effects and mitigation algorithms on stand-off automatic facial recognition. In Optics and Photonics for Counterterrorism, Crime Fighting, and Defence VIII, pages 182–198. SPIE, 2012. 4

[33] Dalong Li, Russell M Mersereau, and Steven Simske. Atmospheric turbulence-degraded image restoration using principal components analysis. IEEE Geoscience and Remote Sensing Letters, 4(3):340–344, 2007. 1

[34] Nianyi Li, Simron Thapa, Cameron Whyte, Albert W. Reed, Suren Jayasuriya, and Jinwei Ye. Unsupervised non-rigid image distortion removal via grid deformation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 2522–2532, 2021. 2

[35] Jing Liu, Jiaxiang Wang, Weikang Wang, and Yuting Su. Dsnet: Dynamic spatiotemporal network for video salient object detection. Digital Signal Processing, 130:103700, 2022. 3, 8

[36] Zhiyuan Mao, Nicholas Chimitt, and Stanley H Chan. Image reconstruction of static and dynamic scenes through anisoplanatic turbulence. IEEE Transactions on Computational Imaging, 6:1415–1428, 2020. 1, 2, 5, 6

[37] Zhiyuan Mao, Nicholas Chimitt, and Stanley H Chan. Accelerating atmospheric turbulence simulation via learned phase-to-space transform. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 14759–14768, 2021. 1, 2, 4

[38] Zhiyuan Mao, Ajay Jaiswal, Zhangyang Wang, and Stanley H Chan. Single frame atmospheric turbulence mitigation: A benchmark study and a new physics-inspired transformer model. In European Conference on Computer Vision, pages 430–446. Springer, 2022. 1, 2, 5

[39] Jiri Matas, Charles Galambos, and Josef Kittler. Robust detection of lines using the progressive probabilistic hough transform. Comput. Vis. Image Underst., 78:119–137, 2000. 6

[40] Mark Olano, John C Hart, Wolfgang Heidrich, Bill Mark, and Ken Perlin. Real-time shading languages. SIGGRAPH Course Notes, 2002. 4

[41] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In Seminal Graphics Papers: Pushing the Boundaries, Volume 2, pages 577–582. 2023. 4

[42] Ken Perlin. An image synthesizer. In Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, page 287–296, New York, NY, USA, 1985. Association for Computing Machinery. 4

[43] G Potvin, JL Forand, and D Dion. A parametric model for simulating turbulence effects on imaging systems. DRDC Valcartier TR 2006, 787, 2007. 2

[44] Guy Potvin, J Luc Forand, and Denis Dion. A simple physical model for simulating turbulent imaging. In Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXII, pages 323–335. SPIE, 2011. 4

[45] Dehao Qin, Ripon Saha, Suren Jayasuriya, Jinwei Ye, and Nianyi Li. Unsupervised region-growing network for object segmentation in atmospheric turbulence. arXiv:2311.03572, 2023. 1, 2, 3, 5, 6, 8

[46] Colin N Reinhardt, Stephen M Hammel, and Dimitris Tsintikidis. Efficient physics-based predictive 3d image modeling and simulation of optical atmospheric refraction phenomena. In Laser Communication and Propagation through the Atmosphere and Oceans V, page 99790S. International Society for Optics and Photonics, 2016. 2

[47] Endre Repasi and Robert Weiss. Computer simulation of image degradations by atmospheric turbulence for horizontal views. In Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXII, page 80140U. International Society for Optics and Photonics, 2011. 2, 4

[48] Marcos Roberto e Souza, Helena de Almeida Maia, and Helio Pedrini. Survey on digital video stabilization: Concepts, methods, and challenges. ACM Comput. Surv., 55(3), 2022. 2

[49] M.C. Roggemann, B.M. Welsh, and B.R. Hunt. Imaging Through Turbulence. Taylor & Francis, 1996. 1, 2

[50] Michael C Roggemann. Optical performance of fully and partially compensated adaptive optics systems using least-squares and minimum variance phase reconstructors. Computers & Electrical Engineering, 18(6):451–466, 1992. 4

[51] Ripon Kumar Saha, Esen Salcin, Jihoo Kim, Joseph Smith, and Suren Jayasuriya. Turbulence strength c_n^2 estimation from video using physics-based deep learning. Optics Express, 30(22):40854–40870, 2022. 3, 4

[52] Armin Schwartzman, Marina Alterman, Rotem Zamir, and Yoav Y Schechner. Turbulence-induced 2d correlated image distortion. In 2017 IEEE International Conference on Computational Photography (ICCP), pages 1–13. IEEE, 2017. 2, 4

[53] Valerian Ilich Tatarski. Wave propagation in a turbulent medium. Courier Dover Publications, 2016. 2

[54] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In European conference on computer vision, pages 402–419. Springer, 2020. 3, 6

[55] Yuan Xie, Wensheng Zhang, Dacheng Tao, Wenrui Hu, Yanyun Qu, and Hanzi Wang. Removing turbulence effect via hybrid total variation and deformation-guided kernel regression. IEEE Transactions on Image Processing, 25(10): 4943–4958, 2016. 2

[56] Rajeev Yasarla and Vishal M Patel. Learning to restore images degraded by atmospheric turbulence using uncertainty. In 2021 IEEE International Conference on Image Processing (ICIP), pages 1694–1698. IEEE, 2021. 1, 2, 5

[57] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2657–2666, 2022. 3, 8

[58] Steve Zamek and Yitzhak Yitzhaky. Turbulence strength estimation from an arbitrary set of atmospherically degraded images. JOSA A, 23(12):3106–3113, 2006. 4

[59] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang.

Restormer: Efficient transformer for high-resolution image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5728–5739, 2022. 5

[60] Xingguang Zhang, Zhiyuan Mao, Nicholas Chimitt, and Stanley H Chan. Imaging through the atmosphere using turbulence mitigation transformer. IEEE Transactions on Computational Imaging, 2024. 2

[61] Xiang Zhu and Peyman Milanfar. Removing atmospheric turbulence via space-invariant deconvolution. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(1):157–170, 2012. 2