



AFRL-RY-WP-TR-2020-0375

DEEP LEARNING FOR ANISOPLANATIC OPTICAL TURBULENCE MITIGATION IN LONG RANGE IMAGING

**Matthew Anthony Hoffmire
University of Dayton**

**DECEMBER 2020
Final Report**

Approved for public release; distribution is unlimited.

See additional restrictions described on inside pages

©Matthew Anthony Hoffmire

STINFO COPY

**AIR FORCE RESEARCH LABORATORY
SENSORS DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7320
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE**

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YY) December 2020		2. REPORT TYPE Thesis		3. DATES COVERED (From - To) 12 November 2020 –12 November 2020	
4. TITLE AND SUBTITLE DEEP LEARNING FOR ANISOPLANATIC OPTICAL TURBULENCE MITIGATION IN LONG RANGE IMAGING				5a. CONTRACT NUMBER FA8650-18-F-1710	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER 62204F/63203F/63270F	
6. AUTHOR(S) Matthew Anthony Hoffmire				5d. PROJECT NUMBER 2003/665A/3720	
				5e. TASK NUMBER N/A	
				5f. WORK UNIT NUMBER Y1UW	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Dayton 300 College Park Dayton, OH 45469				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Sensors Directorate Wright-Patterson Air Force Base, OH 45433-7320 Air Force Materiel Command United States Air Force				10. SPONSORING/MONITORING AGENCY ACRONYM(S) AFRL/RYMT	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-RY-WP-TR-2020-0375	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES PAO case number 88ABW-2020-3547, Clearance Date 12 November 2020. © 2020 Matthew Anthony Hoffmire. Submitted to the School of Engineering of the University of Dayton in partial fulfillment of the requirements for the Degree of Master of Science in Electrical Engineering.. This work was funded in whole or in part by Department of the Air Force. The U.S. Government has for itself and others acting on its behalf an unlimited, paid-up, nonexclusive, irrevocable worldwide license to use, modify, reproduce, release, perform, display, or disclose the work by or on behalf of the U.S. Government. This report is from Task Order 0002 of parent contract FA8650-17-D-1801. Report contains color.					
14. ABSTRACT The acquisition of long range images is often affected by atmospheric optical turbulence. Atmospheric turbulence causes degradation in the images that lowers their utility. The turbulence degradation is caused by changes in temperature and pressure that causes quasi-periodic spatial and temporal blur as well as warping [1,3–5]. The blur and warp is an effect of variations in the index of refraction along the optical path. In the field of astronomy, this has been heavily investigated [4]. For astronomy the field-of-view tends to be narrow which causes the scene to be affected by the atmosphere evenly across the entirety of the image. The uniform effect of this degradation allows one to model the blurring of the entire scene with one linear spatially invariant (LSI) point spread function (PSF) and the warping is only global shifting. This type of imaging scenario with a narrow field-of-view is called isoplanatic [1]. In the case of terrestrial long-range imaging of extended scenes there is a wide field-of-view. This leads to the atmosphere not being uniform over the entire scene, requiring multiple PSFs to be used to successfully model the degradation. This scenario is referred to as anisoplanatic. Turbulence mitigation (TM) in the case of anisoplanatic conditions tends to be a very challenging problem and this is what we address in this thesis.					
15. SUBJECT TERMS anisoplanatic imaging, deep learning, optical turbulence, turbulence mitigation					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OF PAGES 75	19a. NAME OF RESPONSIBLE PERSON (Monitor) Richard Van Hook 19b. TELEPHONE NUMBER (Include Area Code) N/A
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			

DEEP LEARNING FOR ANISOPLANATIC OPTICAL TURBULENCE MITIGATION
IN LONG RANGE IMAGING

Thesis
Submitted to
The School of Engineering of the
UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for
The Degree of
Master of Science in Electrical Engineering

By
Matthew Anthony Hoffmire

Dayton, Ohio

December, 2020



DEEP LEARNING FOR ANISOPLANATIC OPTICAL TURBULENCE MITIGATION
IN LONG RANGE IMAGING

Name: Hoffmire, Matthew Anthony

APPROVED BY:

Russell C. Hardie, Ph.D.
Advisor Committee Chairman
Professor, Electrical and Computer
Engineering

Eric Balster, Ph.D.
Committee Member
Department Chair; Professor,
Electrical and Computer Engineering

Barry K. Karch, Ph.D.
Committee Member
Principal Electronics Engineer, Air
Force Research Labs

Robert J. Wilkens, Ph.D., P.E.
Associate Dean for Research and Innovation
Professor
School of Engineering

Eddy M. Rojas, Ph.D., M.A., P.E.
Dean, School of Engineering

© Copyright by
Matthew Anthony Hoffmire

All rights reserved

2020

ABSTRACT

DEEP LEARNING FOR ANISOPLANATIC OPTICAL TURBULENCE MITIGATION IN LONG RANGE IMAGING

Name: Hoffmire, Matthew Anthony
University of Dayton

Advisor: Dr. Russell C. Hardie

We present a novel deep learning approach for restoring images degraded by atmospheric optical turbulence. We consider the case of terrestrial imaging over long ranges with a wide field of view. This produces an anisoplanatic imaging scenario where the turbulence warping and blurring varies spatially across the image. The proposed turbulence mitigation (TM) method assumes that a sequence of short exposure images are acquired. A block matching algorithm for dewarping observed frames is applied and the resulting images are averaged. A convolutional neural network (CNN) is then employed to perform spatially adaptive restoration. Training the CNN is accomplished using simulated data from a fast simulation tool that mimics turbulence effects and is capable of producing a large amount of degraded imagery from ground truth imagery rapidly. Testing is done using independent data simulated with a different well-validated and pioneering anisoplanatic turbulence simulator. The anisoplanatic simulator is known to be very accurate in modeling turbulence. However, because it is much more computationally demanding, it is used here only for producing the limited amount of testing data needed. Our proposed TM method is evaluated in a number of experiments using quantitative metrics. The quantitative analysis is made possible by virtue of having ground truth imagery that is available with simulated data. A number of restored images are also provided for subjective evaluation. We demonstrate

that the new TM method outperforms all of the benchmark methods in the scenarios tested in this thesis.

For my grandfather John Valosen. Thank you for your impact in my life. May you rest in
peace.

ACKNOWLEDGMENTS

I would like to thank my family for believing in me and always being there to for me. You ensured I received the mental, spiritual, and emotional support I needed. I love you all and God bless!

Another individual that I would like to thank is Dr. Russell Hardie. I have learned a great deal from him as my advisor. Without your support through all the difficulties that arose during my graduate degree, I would not be where I am now. You have gone above and beyond your duties to help me whenever I needed it. Your dedication to your research and to never be satisfied with just adequate work is a lesson that I will take with me throughout my career.

Thank you to Mike Rucci. You were always willing to answer my many questions and explain things in a different way for me to understand. I have gained a deeper understanding of many topics from you. Your mentorship to me has been invaluable during my degree.

To the members at AFRL/RYMT, thank you for bringing on as part of the team and always being willing to help.

Also, thank you to my committee members for making time out of their busy schedules to review my thesis and attend my defense.

Lastly, I want to thank Malia Harvey. Your endless care and encouragement has allowed me to complete my degree. You have been my rock through this whole endeavor.

TABLE OF CONTENTS

ABSTRACT	iii
DEDICATION	v
ACKNOWLEDGMENTS	vi
LIST OF FIGURES	viii
LIST OF TABLES	xii
CHAPTER I. INTRODUCTION	1
1.1 Atmospheric Optical Turbulence	1
1.2 Approaches to Turbulence Mitigation	1
1.3 Turbulence Simulation	3
1.4 Objective of Study	3
1.5 Structure of Thesis	5
CHAPTER II. Turbulence Simulators	6
2.1 Turbulence Characterization	6
2.1.1 Turbulence Statistics	6
2.1.2 Atmospheric OTF and PSF	8
2.2 Anisoplanatic Numerical Wave Propagation Simulation	11
2.3 Fast Warping Simulation	15
2.4 Simulator Comparison	16
CHAPTER III. Turbulence Mitigation	19
3.1 Approaches to Turbulence Mitigation	19
3.2 BM-WF	20
3.3 BM-CNN	21
3.3.1 Overview	21
3.3.2 BM-CNN Network Architecture	21
3.3.3 BM-CNN Training	22
CHAPTER IV. Experimental Results	26
4.1 Quantitative Results	26
4.2 Qualitative Image Results	28
CHAPTER V. Conclusions	54
BIBLIOGRAPHY	56

LIST OF FIGURES

2.1	Long exposure OTF for the parameters in Tables 2.1 and 2.2.	10
2.2	Short exposure OTF for the parameters in Tables 2.1 and 2.2.	10
2.3	Atmospheric optical turbulence simulation overview. An ideal image is processed with a simulator to produce a sequence of degraded frames. The simulator takes into account the optical parameters and the atmospheric parameters.	11
2.4	Anisoplanatic numerical wave propagation geometry. Point sources in the object plane ($z = 0$) are propagated using numerical wave propagation to the pupil plane of the camera to ultimately form PSF in the image plane of the camera.	13
2.5	Four examples of anisoplanatic simulator PSFs for Level 3 turbulence: (a) PSFs as images, (b) PSFs as mesh plots.	14
2.6	Images from the anisoplanatic simulator for Level 3 turbulence: (a) Ideal, (b) degraded, (c) degraded image with superimposed quiver plot showing the PSF centroid based warping motion.	15
2.7	Average SE PSF for Level 3 turbulence: (a) PSF as image, (b) PSF as mesh plot.	16
2.8	Image degradation results using the fast warping simulator on training data Image 16 for 30 input frames. The turbulence levels go from 1 to 4 from top row to bottom row. The noise levels varying in the columns and noise standard deviation of 0, 10, and 30. The numbers below each image correspond to row and column number respectively.	17
3.1	BM-WF turbulence mitigation block diagram [1].	21
3.2	Proposed BM-CNN turbulence mitigation block diagram. Note that we have replaced the WF in the BM-WF method with a CNN to potentially improve performance.	22
3.3	FIFNet CNN network architecture used for the proposed BM-CNN TM method.	23
3.4	BM-CNN Training Block Diagram	25
4.1	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 1, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	34

4.2	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	35
4.3	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	36
4.4	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	37
4.5	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 2, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	38
4.6	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	39
4.7	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	40
4.8	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	41
4.9	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 2, 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	42
4.10	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	43

4.11	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	44
4.12	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	45
4.13	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 3, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	46
4.14	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	47
4.15	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	48
4.16	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	49
4.17	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 4, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	50
4.18	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	51
4.19	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	52

4.20	Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.	53
------	--	----

LIST OF TABLES

2.1	Turbulence parameters used in the simulation results.	8
2.2	Optical parameters used in simulation results.	8
3.1	Turbulence mitigation methods used and their acronyms.	20
3.2	Description of the FIFNet layers [2] used for the proposed BM-CNN TM method. The spatial dimensions of the activations are for training using 61×61 image patches.	24
4.1	Average PSNR (dB) results for 100 validation sequences with turbulence level 1 and 30 input frames.	29
4.2	Average SSIM results for 100 validation sequences with turbulence level 1 and 30 input frames.	29
4.3	Average PSNR (dB) results for 100 validation sequences with turbulence level 2 and 30 input frames.	30
4.4	Average SSIM results for 100 validation sequences with turbulence level 2 and 30 input frames.	30
4.5	Average PSNR (dB) results for 100 validation sequences with turbulence level 3 and 30 input frames.	31
4.6	Average SSIM results for 100 validation sequences with turbulence level 3 and 30 input frames.	31
4.7	Average PSNR (dB) results for 100 validation sequences with turbulence level 4 and 30 input frames.	32
4.8	Average SSIM results for 100 validation sequences with turbulence level 4 and 30 input frames.	32

CHAPTER I

INTRODUCTION

1.1 Atmospheric Optical Turbulence

The acquisition of long range images is often affected by atmospheric optical turbulence. Atmospheric turbulence causes degradation in the images that lowers their utility. The turbulence degradation is caused by changes in temperature and pressure that causes quasi-periodic spatial and temporal blur as well as warping [1,3–5]. The blur and warp is an effect of variations in the index of refraction along the optical path. In the field of astronomy, this has been heavily investigated [4]. For astronomy the field-of-view tends to be narrow which causes the scene to be affected by the atmosphere evenly across the entirety of the image. The uniform effect of this degradation allows one to model the blurring of the entire scene with one linear spatially invariant (LSI) point spread function (PSF) and the warping is only global shifting. This type of imaging scenario with a narrow field-of-view is called isoplanatic [1]. In the case of terrestrial long-range imaging of extended scenes there is a wide field-of-view. This leads to the atmosphere not being uniform over the entire scene, requiring multiple PSFs to be used to successfully model the degradation. This scenario is referred to as anisoplanatic. Turbulence mitigation (TM) in the case of anisoplanatic conditions tends to be a very challenging problem and this is what we address in this thesis.

1.2 Approaches to Turbulence Mitigation

There are many different approaches proposed in the literature to address the problem of TM via post processing of degraded images. Many of these methods are based on using short-exposure (SE) images [1, 6]. The reasoning is that SE images can effectively freeze

warping distortions temporally. This produces images with minimal motion blur. However, the geometry of these individual images varies temporally and is most likely incorrect [7,8]. On the other hand, averaging the SE frames produces an image that is equivalent to a long exposure (LE) image. The LE image has the correct geometry but also significant amounts of motion blur [1].

For the reasons mentioned above, many TM algorithms follow the steps of dewarping, fusion, and then blind deconvolution [1,9–16]. Dewarping refers to motion compensation of the SE frames to reduce the effects of motion blur. The motion compensation is typically followed by some sort of averaging to combine the frames together to reduce noise. This fusion also serves to average out the spatially varying blur so that the resulting blur can be well modeled by a simpler LSI PSF [1]. Finally, the deconvolution step attempts to undo the blurring from the LSI PSF. The Block Matching and Wiener Filtering (BM-WF) [1] method uses a parametric PSF model and a fixed Wiener filter for the deconvolution. The parametric model for the PSF takes into account diffraction from the optics, SE blurring, and residual motion blur from imperfect image registration using block matching. The parametric PSF used in BM-WF has only two unknown parameters, making it much easier to optimize than traditional blind deconvolution methods.

Another popular TM method involves what is referred to as bispectral speckle imaging (BSI) [9–11, 17–24]. This BSI method attempts to mitigate turbulence through the estimation of image magnitude and phase separately. The name comes from “high-frequency intensity fluctuations that are referred to as speckles” [10]. The BM-WF outperformed BSI in the studies reported in [1]. For this reason, we have chosen to build on the framework of BM-WF for this thesis.

1.3 Turbulence Simulation

Designing, optimizing, and evaluating TM algorithms is complicated by the difficulty in obtaining ground truth imagery. Recently, there have been a number of turbulence simulators for anisoplanatic imaging presented in the literature [20, 25–27]. Simulation has the advantage of being able to produce a virtually unlimited amount of ground truth data for nearly any scenario. These simulated data can be used for training machine learning methods and for quantitative performance analysis of any TM methods. We believe accurate simulation tools are essential to advance the field of TM.

One well-validated turbulence simulator is the numerical wave propagation anisoplanatic simulator developed by Hardie *et al* and described in [3]. This is the method we shall use for testing the TM algorithms presented in this thesis. We consider this method the “gold standard” for anisoplanatic turbulence data. However, it does have a rather high computational complexity. Thus, we also consider a fast simulation method based on that in Schwartzman [28]. We have extended Schwartzman’s simulator to add atmospheric blurring and we use different tilt statistics as described in Chapter II. For machine learning TM methods that require a large amount of training data, we believe our fast simulator is an effective choice. While we use the fast simulator for training, we shall validate all of our results by generating testing data using the anisoplanatic simulator.

1.4 Objective of Study

The Wiener filter used for deconvolution in the BM-WF method is known to be optimum in a mean squared error sense for wide sense stationary (WSS) Gaussian images and noise, and an LSI blur. However, in practice, we know that real images are not generally WSS

and the statistics are not necessarily Gaussian. Furthermore, while the fusion process does seek to make the residual blur LSI, it may not be perfectly successful in this regard. For these reasons, we believe that there is room for improvement in the BM-WF approach with respect to the Wiener filter. It is our hypothesis that we can replace the Wiener filter with a convolutional neural network (CNN) and use deep learning (DL) to obtain improved performance relative to the BM-WF method. We argue that the non-linear CNN method is better equipped to address the non-stationary nature of most images and also address and spatially varying residual blur that may be present. The availability of the new atmospheric turbulence simulators described above makes it possible to generate a virtually unlimited amount of training data to support successful DL with a CNN in this application. We refer to the new proposed algorithm as the Block Matching Convolutional Neural Network (BM-CNN) method.

Recently CNNs have been used heavily in image processing for a variety of tasks including image restoration and image super resolution (SR). There have been many studies done that show CNNs are successful in restoring images [29–33]. While there are many different types of networks that can do SR and restoration [34–39], for the purposes of this thesis, we employ one recently proposed by Elwarfalli and Hardie [2]. The network is referred to as the Fusion of Interpolated Frames Network (FIFNet). This has been chosen based on good demonstrated SR performance and its lightweight architecture. That is, it has relatively few convolution layers and can be trained from scratch rather quickly. Large networks can take days or even weeks to train from scratch. FIFNet can be trained in a matter of hours. The larger networks also often require transfer learning where pretrained networks are imported and refined with further training. Such transfer learning is not necessary for FIFNet.

To the best of our knowledge, there is only one prior study in the literature that has been done with using CNNs in the field of TM [40]. In that work, they show that a CNN can restore images affected by turbulence. However, their experimental study uses a very simple simulator with images degraded by only 9 different PSFs. Furthermore, they do not employ any image registration in their method. In contrast, here we use a well validated and sophisticated anisoplanatic simulator [3] for validation, and a modified version of the fast warping simulator from [28] for training. Our BM-CNN method also gains improved performance by incorporating block matching registration prior to CNN processing.

We study the performance of our BM-CNN method on four different turbulence levels at two noise levels. We use 800 training sequences and 100 independent test sequences. Our training is done with the fast simulator, but testing is done with both the fast simulator and the “gold standard” anisoplanatic simulator [3]. We shall show there that the BM-CNN consistently outperforms the BM-WF method in the scenarios tested here. We find this to be especially true in light to moderate turbulence and moderate to high noise levels.

1.5 Structure of Thesis

The rest of the thesis is organized as follows. Chapter II provides a description of turbulence characterization statistics. It also explains the two turbulence simulation models used. Chapter III introduces the different TM methods studied in this thesis. Specifically, we introduce our proposed BM-CNN method and compare and contrast it with the BM-WF benchmark method. Details of the CNN architecture and training processed are also provided in Chapter III. Chapter IV presents the experimental results in this thesis. Finally, conclusions are articulated in Chapter V.

CHAPTER II

Turbulence Simulators

2.1 Turbulence Characterization

2.1.1 Turbulence Statistics

Atmospheric optical turbulence impacts the imaging process by introducing random pockets of air with varying indices of refraction along the optical path. Variations in the index of refraction are often modeled with a refractive index structure function. This is given by

$$D_n(\mathbf{r}) = \left\langle [n(\mathbf{x} + \mathbf{r}) - n(\mathbf{x})]^2 \right\rangle = C_n^2 r^{2/3}, \quad (2.1)$$

where $\mathbf{x}$ and $\mathbf{r}$ are 3-D spatial coordinate vectors, $r = |\mathbf{r}|$, $n(\cdot)$ is the index of refraction, and $\langle \cdot \rangle$ represents an ensemble mean operator. The parameter C_n^2 is the refractive index structure parameter and it has units of $\text{m}^{-2/3}$. Typical values of C_n^2 tend to range from 1×10^{-13} to 1×10^{-17} [41]. When this quantity varies along the optical path, it may be expressed as a function, $C_n^2(z)$, and here we define z as the distance from the source.

Another key turbulence statistic that takes the full optical path into account is called the Fried parameter and is denoted r_0 . The Fried parameter is also known as the atmospheric coherence diameter [4,41]. This parameter can be expressed in terms of the refractive index structure function as

$$r_0 = \left[0.423 \left(\frac{2\pi}{\lambda} \right)^2 \int_{z=0}^{z=L} C_n^2(z) \left(\frac{z}{L} \right)^{5/3} dz \right]^{-3/5}, \quad (2.2)$$

where λ is the wavelength and the path length is given by the variable L . It is also important to note that here $z = 0$ is at the source of the image and $z = L$ is at the camera. The Fried parameter is the key parameter that governs the atmospheric optical transfer function, the

point spread function, and point source tilt variance [42]. Note that a small r_0 relative to the camera aperture indicates a high level of atmospheric turbulence.

Warping from turbulence can be characterized by the point source angular tilt variance. The two-axis Zernike tilt (Z-tilt) variance in units of radians squared is given by Tyson [43] in terms of r_0 as

$$T_Z^2 = .3641 \left(\frac{D}{r_0} \right)^{5/3} \left(\frac{\lambda}{D} \right)^2, \quad (2.3)$$

where D is the aperture of the optics. As turbulence strength increases, we see a smaller r_0 and increased tilt variance. The increase in tilt variance corresponds into increased warping in the observed imagery. There is also an increase in blurring as governed by the OTF and PSF in Chapter 2.1.2.

Another relevant statistic is the isoplanatic angle [4]. Point sources separated by less than this angle will have similar PSFs and a mean wave function phase difference at the aperture of less than 1 radian [41, 44]. The isoplanatic angle can also be expressed as a weighted integral of $C_n^2(z)$, yielding [41, 44]

$$\theta_0 = \left[2.91 k^2 L^{5/3} \int_{z=0}^{z=L} C_n^2(z) \left(1 - \frac{z}{L} \right)^{5/3} dz \right]^{-3/5}. \quad (2.4)$$

Generally speaking, as the turbulence level increases, the isoplanatic patch size decreases. This means the warping is less correlated spatially over a given separation distance. This tends to make it harder to perform image registration.

The turbulence parameters studied in this thesis are listed in Table 2.1. The optical system considered is detailed in Table 2.2. All of the experimental results presented in Chapter IV are based on these values. We consider a visible system operating at relatively long range (7 km) at 4 different turbulence levels. We further assume that the sensor is Nyquist sampled relative to the diffraction limited optical cutoff frequency. Note that the

work in [42] does attempt to address the problems of turbulence and undersampling jointly using an approach based on the BM-WF method. We forgo that extra complication in this study.

Table 2.1: Turbulence parameters used in the simulation results.

Parameter	Turbulence Degradation			
	Level 1	Level 2	Level 3	Level 4
$C_n^2 \times 10^{-15} \text{ (m}^{-2/3}\text{)}$	0.0307	0.1227	0.4909	1.9638
Theoretical r_0 (m)	0.3863	0.1681	0.0732	0.0319
Theoretical D/r_0 (unitless)	0.5265	1.2097	2.7789	6.3843
Isoplanatic Angle (pixels)	13.4454	5.8516	2.5472	1.1087
RMS Tilt (pixels)	0.5000	1.0000	2.0000	4.0000

Table 2.2: Optical parameters used in simulation results.

Parameter	Value
Aperture	$D = 0.2034 \text{ (m)}$
Focal length	$l = 1.2000 \text{ (m)}$
F-number	5.8997
Wavelength	$\lambda = 0.5250 \text{ (}\mu\text{m)}$
Object distance	$L = 7000 \text{ (m)}$
Nyquist pixel spacing	$\delta = 1.5487 \text{ (}\mu\text{)m}$
Cut-off frequency	$\rho_c = 322.8571 \text{ (cycles/mm)}$

2.1.2 Atmospheric OTF and PSF

The overall optical transfer function (OTF) for an imaging system in turbulence may be modeled to include an atmospheric and diffraction component. This is given by

$$H(\rho) = H_{atm}(\rho)H_{dif}(\rho), \quad (2.5)$$

where $\rho = \sqrt{u^2 + v^2}$ and u and v are the spatial frequencies in units of cycles per unit distance. The atmospheric OTF is given by [4]

$$H_{atm}(\rho) = \exp \left\{ -3.44 \left(\frac{\lambda l \rho}{r_0} \right)^{5/3} \left[1 - \alpha \left(\frac{\lambda l \rho}{D} \right)^{1/3} \right] \right\}, \quad (2.6)$$

where l is the camera focal length and D is the aperture diameter. The parameter α here serves as a switch. In particular, the long exposure atmospheric OTF is generated by Eq. (2.6) when $\alpha = 0$, and the average short exposure OTF is given by Eq. (2.6) when $\alpha = 1$. The diffraction limited OTF for a circular aperture is given by [45]

$$H_{dif}(\rho) = \begin{cases} \frac{2}{\pi} \left[\cos^{-1} \left(\frac{\rho}{2\rho_c} \right) - \frac{\rho}{2\rho_c} \sqrt{1 - \left(\frac{\rho}{2\rho_c} \right)^2} \right] & \rho \leq \rho_c \\ 0 & \text{otherwise} \end{cases}, \quad (2.7)$$

where $\rho_c = 1/(\lambda f_n)$ is the optical cut-off frequency, and the f-number is $f_n = l/D$. Note that the theoretical atmospheric PSF is found by applying the inverse Fourier transform to Equation (2.5). Examples of optical transfer functions for the optical parameters and turbulence levels in Tables 2.1 and 2.2 are shown in Figs. 2.1 and 2.2. The long-exposure OTFs are in Fig. 2.1 and the short-exposure OTFs are in Fig. 2.2.

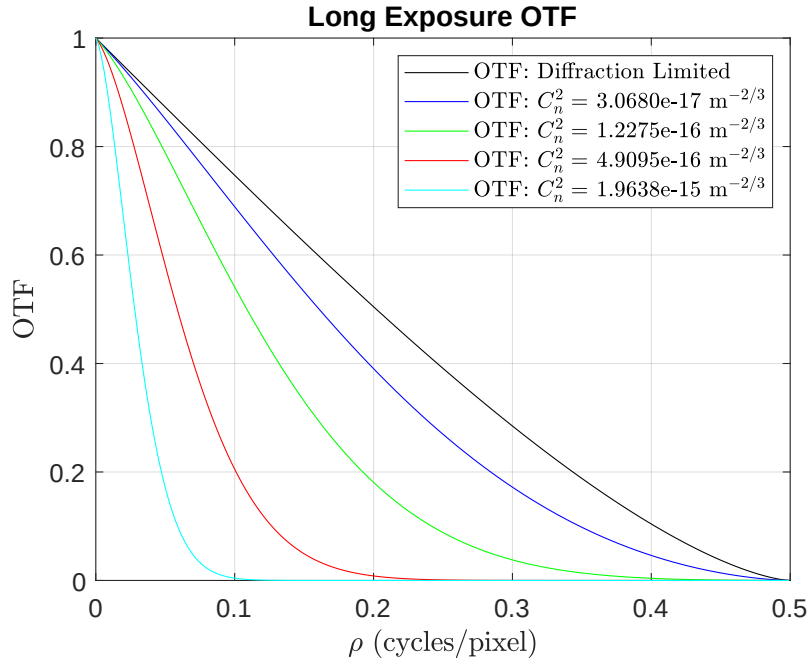


Figure 2.1: Long exposure OTF for the parameters in Tables 2.1 and 2.2.

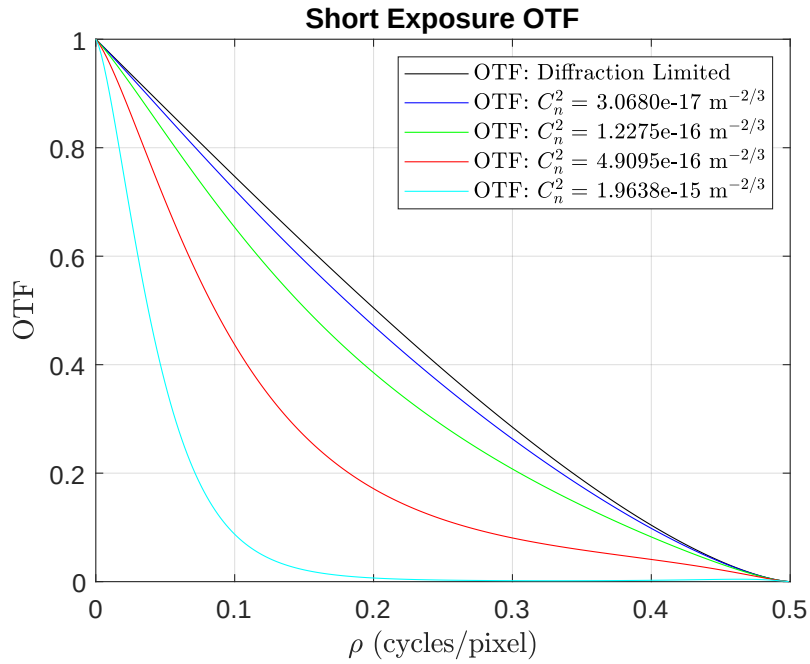


Figure 2.2: Short exposure OTF for the parameters in Tables 2.1 and 2.2.

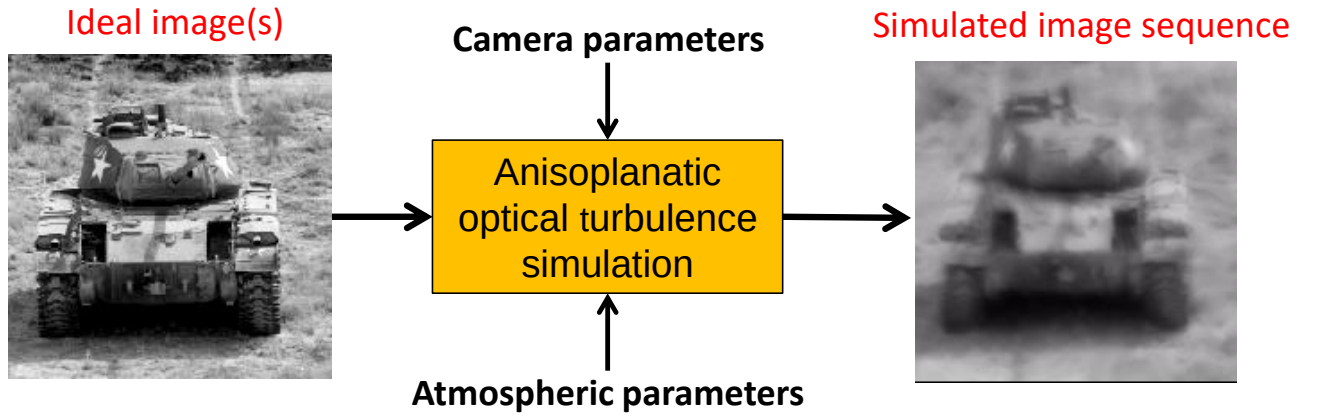


Figure 2.3: Atmospheric optical turbulence simulation overview. An ideal image is processed with a simulator to produce a sequence of degraded frames. The simulator takes into account the optical parameters and the atmospheric parameters.

2.2 Anisoplanatic Numerical Wave Propagation Simulation

The anisoplanatic simulator is a numerical wave propagation simulator that produces realistic anisoplanatic turbulence degradation. Extensive validation has been performed with this simulator [3]. Thus, we treat this as the “gold standard” for our data generation. All of the the anisoplanatic simulation parameters used in this thesis are the same as those reported in [3].

The anisoplanatic simulator begins by taking a number of optical system parameters. These are used to create a simulated camera system. The turbulence simulator also requires knowledge of the atmospheric conditions as captured in the function $C_n^2(z)$ as described in Chapter 2.1.1. With knowledge of the optical system and the turbulence parameters, the simulator creates an array of PSFs in order to degrade ideal images into realistic turbulence degraded ones. A diagram showing this process can be seen in Fig. 2.3.

Figure 2.4 shows the geometry for PSF generation. Note that range shown is just for illustrative purposes and is different from what is used in Table 2.2. The left side of the figure at $z = 0$ is called the object plane. The smallest square on the right represents the pupil plane, and the circle within represents the camera aperture. The red boxes represent phase screens. Each phase screen models the atmospheric effects on a propagating wave for the nearby portion of the optical path. There exist N phase screens between the object and pupil plane. For a PSF to be generated, a point source is propagated from the object plane, through a section of each extended phase screen. This section is referred to as the local phase screen and two examples are shown in the blue and green boxes for two different points in the object plane. This point source continues to go through all N phase screens until it reaches the pupil plane and is focused at the camera image plane [3]. While this can be done for every pixel projected to the object plane, that may be computationally prohibitive. To speed things up, the simulator allows the user to evaluate the PSF on a subsampled grid. This leaves spaces between each resulting PSF on the focal plane. The missing PSF are computed using interpolation. Here we evaluate every 4'th pixel in the horizontal and vertical dimensions, following that in [3]. The PSF skipping speeds up the simulation by a factor of 16 times. Once the PSFs are generated, they are applied in a spatially varying convolution process to produce the final degraded images. The centroids of the PSFs will vary and this creates warping. The PSFs themselves creates blurring. Because the PSFs vary spatially, the simulator produces anisoplanatic warp and blur. Note that realistic spatial correlation is naturally created because point sources will propagate through overlapping regions of the phase screens as shown in Fig. 2.4. The closer the point sources, the more overlap the two propagating waves have within the turbulent medium and the more correlation there will be between the resulting PSFs.

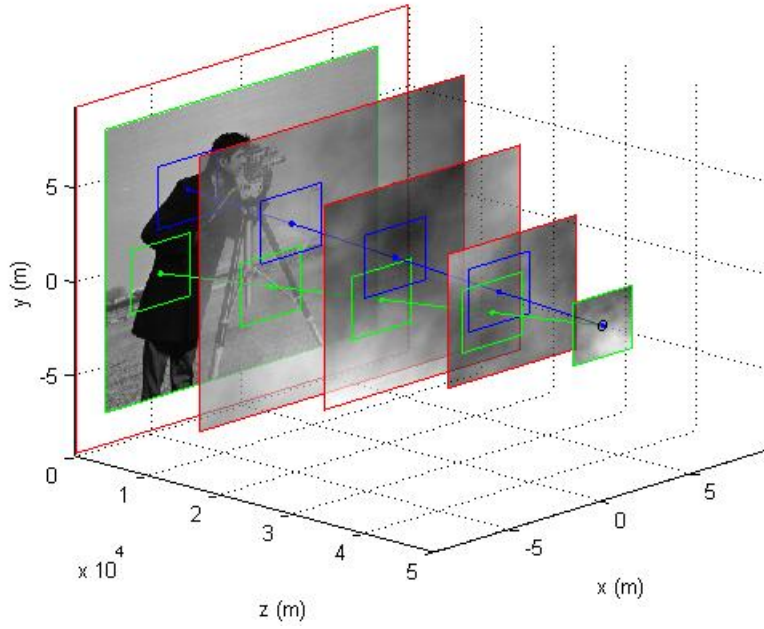
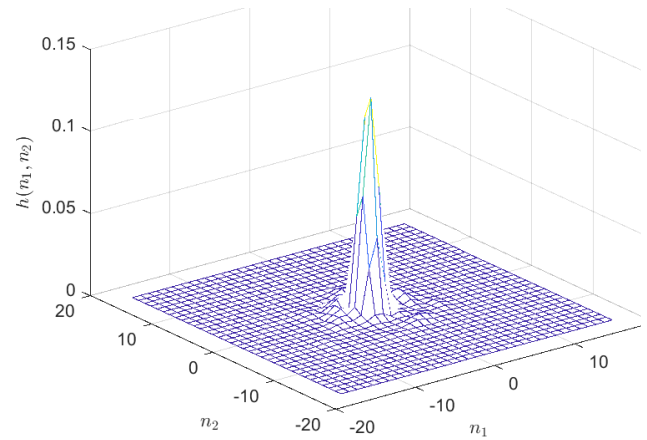
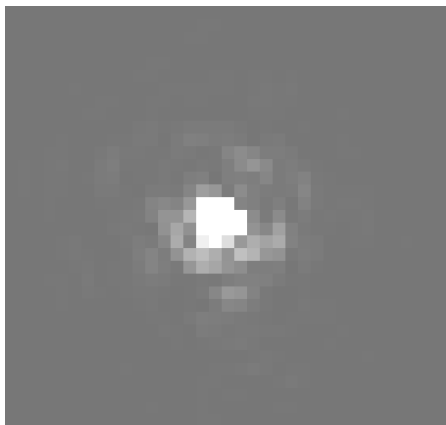
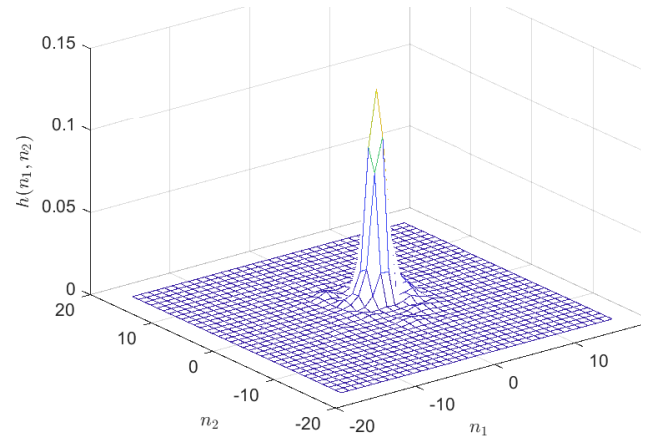
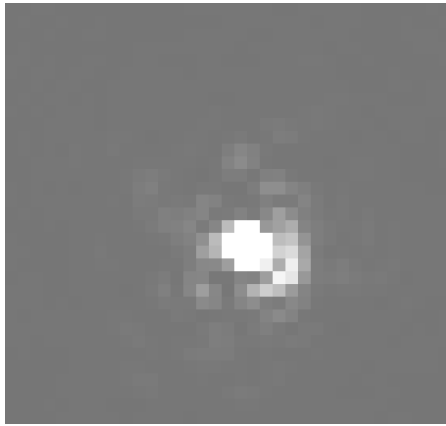
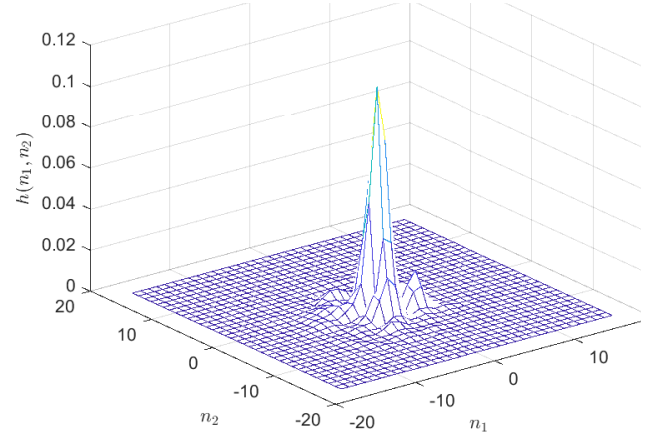
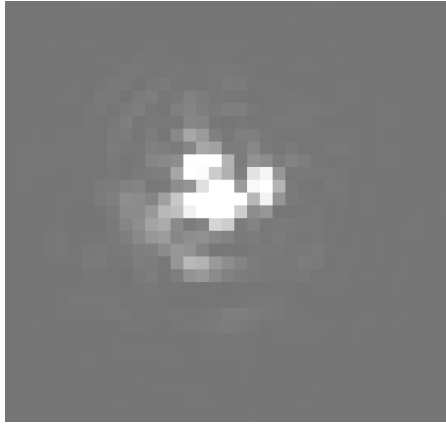


Figure 2.4: Anisoplanatic numerical wave propagation geometry. Point sources in the object plane ($z = 0$) are propagated using numerical wave propagation to the pupil plane of the camera to ultimately form PSF in the image plane of the camera.

Examples of some of the PSFs generated with the anisoplanatic simulator are shown in Fig. 2.5. These are for Level 3 turbulence as defined in Table 2.1. On the left side the PSF is shown as an image and on the right the PSF is plotted as a mesh plot. One can see typical PSF variations that gives us the realistic anisoplanatic degradation.

An example of a simulated image using the anisoplanatic simulator is shown in Fig. 2.6 for Level 3 turbulence. The ideal input image is shown in Fig. 2.6(a) and the degraded image is shown in Fig. 2.6(b). The degraded image with a quiver plot superimposed is shown in Fig. 2.6(c). The quiver arrows show the local warping motion produced by PSF centroid offsets.



(a)

(b)

Figure 2.5: Four examples of anisoplanatic simulator PSFs for Level 3 turbulence: (a) PSFs as images, (b) PSFs as mesh plots.

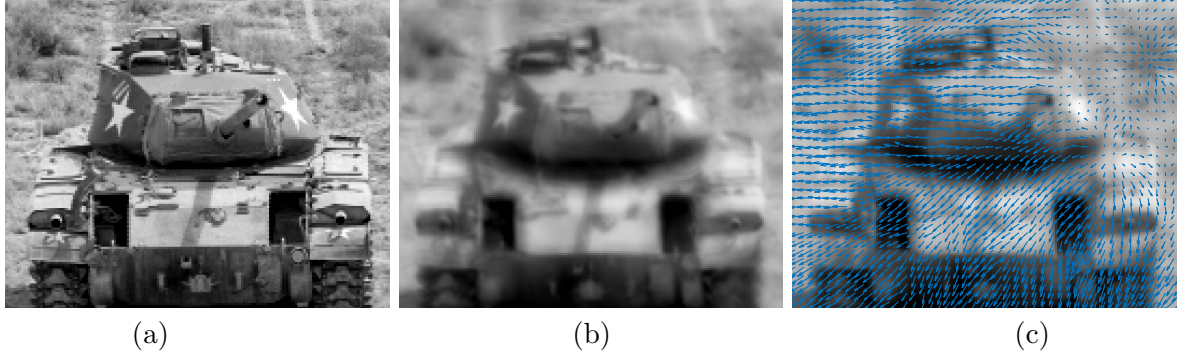


Figure 2.6: Images from the anisoplanatic simulator for Level 3 turbulence: (a) Ideal, (b) degraded, (c) degraded image with superimposed quiver plot showing the PSF centroid based warping motion.

2.3 Fast Warping Simulation

The fast warping simulator used here is based on that presented by Schwartzman *et al* [28]. This simulator works by creating arrays of white Gaussian random noise. The noise is filtered with specially designed LSI filters to transform the white noise into random fields with realistic spatial tilt correlation statistics. These shifts are applied to the image through interpolation leading to what is referred as the distortion field which causes warping [28].

There are two important novel aspects of our fast warping simulator that differ from the one proposed by Schwartzman *et al*. First, in addition to the warping, we also introduce an LSI PSF blurring using the average short-exposure PSF. For one frame, the warping is anisoplanatic, but the blurring is isoplanatic. However, when the simulated frames are averaged, the blurring should be very comparable to the frame average from the anisoplanatic simulator. In Fig. 2.7 an average SE PSF can be seen as both an image, and as mesh plot for Level 3 turbulence. If one were to center and average many PSF like those in Fig. 2.5, the results would closely match the theoretical average SE PSF in Fig. 2.7.

The other difference is that we employ the tilt statistics derived by Santasri Bose-Pillai [46]. These statistics are different from those used in [28] but have been shown to provide excellent agreement with the anisoplanatic simulator statistics [3]. Examples of degraded images from the fast warping simulator can be seen in Fig. 2.8. Images for all four turbulence levels from Table 2.1 are used with three different amounts of noise standard deviation: 0, 10, and 30.

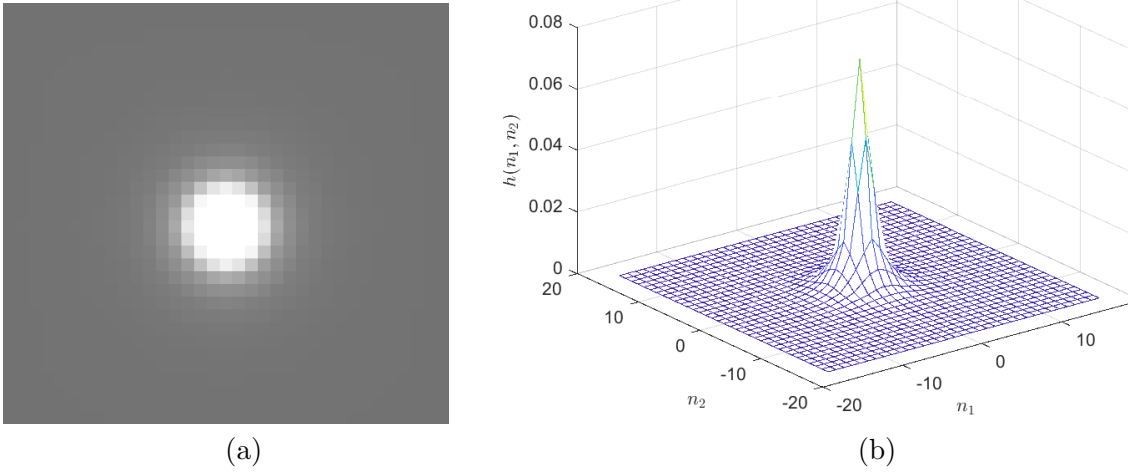


Figure 2.7: Average SE PSF for Level 3 turbulence: (a) PSF as image, (b) PSF as mesh plot.

2.4 Simulator Comparison

While we consider the anisoplanatic simulator to be the “gold standard” for generating accurate optical turbulence degraded frames, it does come with a relatively high computational burden. The warping simulator on the other hand with LSI blurring is much faster. This makes it much more practical for generating the vast quantities of data necessary for DL training. To compare the speeds of the warping simulator and the anisoplanatic simu-

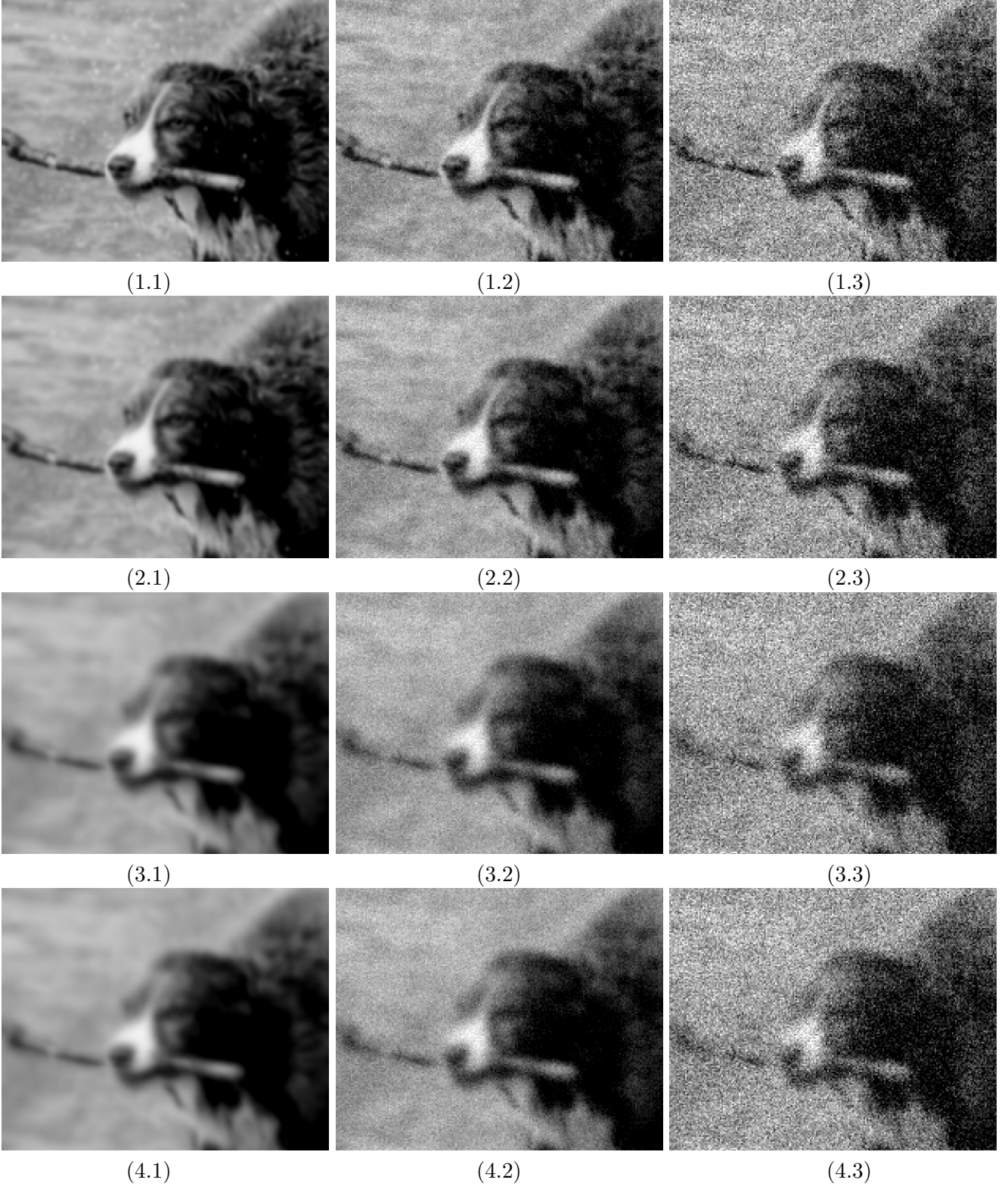


Figure 2.8: Image degradation results using the fast warping simulator on training data Image 16 for 30 input frames. The turbulence levels go from 1 to 4 from top row to bottom row. The noise levels varying in the columns and noise standard deviation of 0, 10, and 30. The numbers below each image correspond to row and column number respectively.

lator, 30 frames of size 256×256 were generated with both simulators and the processing times on the same workstation were recorded. The anisoplanatic simulator took 289 seconds, or nearly 5 minutes to complete this task. The warping simulator did the same in just over 3 seconds. This is approximately two orders of magnitude speed enhancement. This timing study has been conducted using a workstation with Intel(R) Core(TM) i9-9980XE CPU @ 3.00GHz with 128 GB of RAM and two NVIDIA Titan RTX GPUs. Considering that DL training requires hundreds or thousands of images, some of which are much larger than 256×256 , one can clearly see the practical benefit of the fast simulator.

However, speed is not the only thing to consider. The fast simulator is only useful if it generates images that are statistically representative of the real image to be processed. In this thesis we focus on methods that process an average of registered SE frames. Thus, the LSI PSF used by the fast simulator is not problematic, since the average of the anisoplanatic images should be similar. This similarity allows for training of the proposed TM method to take place on the using the fast warping simulator and we rely on using the anisoplanatic simulator for testing. As will be seen in Chapter IV, the results show that this training method is highly effective.

CHAPTER III

Turbulence Mitigation

3.1 Approaches to Turbulence Mitigation

While the main method of the thesis is a deep learning approach, there are many methods that are used to compare against. In Table 3.1 there is a list of methods and the acronyms that they refer to in the rest of the thesis. Each of these methods provides a benchmark for the proposed method to compare against. The SE image is the raw unprocessed image from the simulator plus noise. The LE image is formed as the simple average of the SE frames. In this thesis, all results are generated with 30 frames. The LE images will tend to have greatly reduced noise and improved image geometry. However, it is highly blurred. The noise variance is reduced by a factor equal to the number of frames averaged. The next benchmark method in Table 3.1 is the LE image further processed with a Wiener Filter (WF). The LE-WF method follows the same process as the LE, but adds a deconvolution step using a WF. The theoretical LE PSF is used for the WF and the noise-to-signal (NSR) ratio parameter is optimized on the training data [1].

The remaining method in Table 3.1 employ the Block Matching Algorithm (BMA) to provide dewarping to the SE images. In all cases, the LE image is used as a prototype image to register to, since it has the approximately correct geometry [1]. The first of these BMA based TM algorithms is Block Matching (BM) method. This method uses BMA on the SE images and then averages these to form the image estimate. Processing the BM image with a WF gives rise to the BM-WF method in Table 3.1. This represents the exact same method reported in [1]. Note that here we use integer BMA registration only for computational simplicity. More will be said about the BM-WF method in Chapter 3.2.

Finally, the new proposed method that is the centerpiece of this thesis is the BM-CNN method. More will be said about this in Chapters 3.3 and 3.3.3.

Table 3.1: Turbulence mitigation methods used and their acronyms.

Turbulence Mitigation Method	Acronym
Short Exposure	SE
Long Exposure	LE
Block Matching	BM
Long Exposure Wiener Filter	LE-WF
Block Matching Wiener Filter	BM-WF
Block Matching Convolutional Neural Network	BM-CNN

3.2 BM-WF

The BM-WF method was proposed and described in detail by Hardie *et al* in [1]. A block diagram showing the key steps of this method is provided in Fig. 3.1. The SE frames are acquired and then averaged to form a prototype image with approximately the correct geometry. The BMA algorithm is then used to dewarp the SE images partially. The partially dewarped SE images are averaged. Finally, the average image is processed with a WF. The WF has two parameters, the NSR and a parameter α that represents the amount of tilt correction provided by the BMA registration step. Here we optimize these two parameters using the training data to make a fair comparison with the DL method that has access to training data.

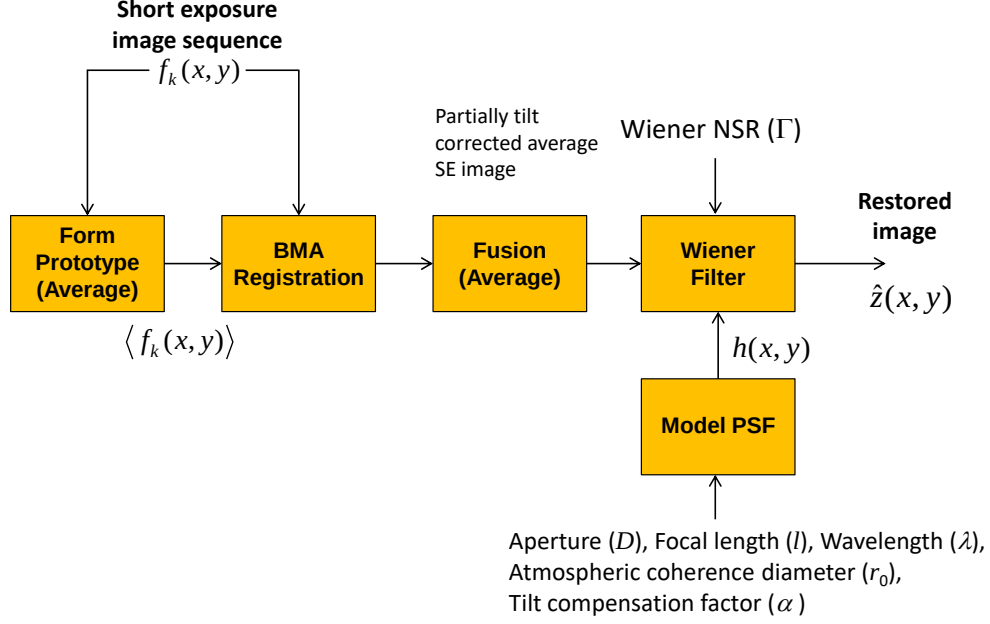


Figure 3.1: BM-WF turbulence mitigation block diagram [1].

3.3 BM-CNN

3.3.1 Overview

The BM-CNN takes a very similar approach, but instead of using a WF, the BMA averaged and fused frames are put into a CNN. Here the CNN learns weights to restore these images. A block diagram detailing the process can be seen in Fig. 3.2.

3.3.2 BM-CNN Network Architecture

As previously stated in Chapter I, the BM-CNN uses the FIFNet network architecture. A useful option that the FIFNet has is that the number of convolution layers and filter sizes can be customized. This allows for the network to be as deep or shallow, as wide or narrow as the user so wishes. For the BM-CNN, 10 convolution (conv) + rectified linear unit (ReLU) activation layers are selected. The first five layers are of filter size 3×3 and the last

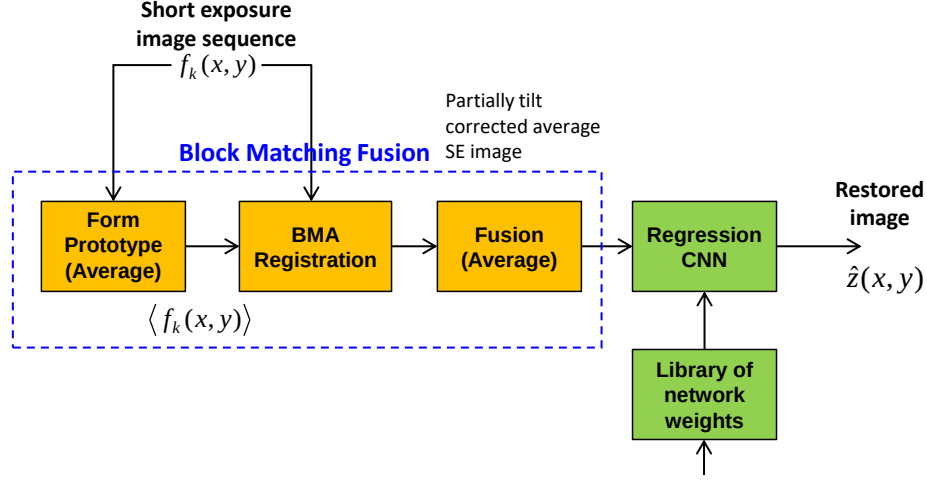


Figure 3.2: Proposed BM-CNN turbulence mitigation block diagram. Note that we have replaced the WF in the BM-WF method with a CNN to potentially improve performance.

five are of size 5×5 for convolution. With the other layers that come from the FIFNet, that makes the total network size of 27 layers, which might be considered a relatively shallow network. In Fig. 3.3 a diagram of the network architecture can be seen. Also, in Table 3.2 the layers of the FIFNet are described.

3.3.3 BM-CNN Training

Training and testing of the network is done here using the DIV2K database [47]. This dataset has become popular for testing image restoration and image super resolution methods. It contains 800 high resolution images for training, and 100 testing. The original images are 24-bit RGB images with 2K resolution. All of the input images have been converted to grayscale with a floating point dynamic range of 0-1. They have also been reduced in size by a scaling factor of 0.25.

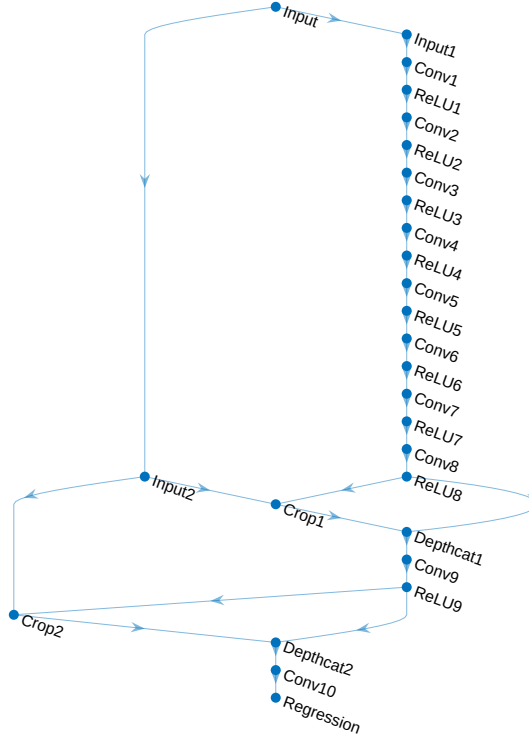


Figure 3.3: FIFNet CNN network architecture used for the proposed BM-CNN TM method.

The basic training process is depicted in Fig. 3.4. The undegraded training data serve as the “truth” images. These images are fed into the fast warping simulator to produced the degraded images. The BMA algorithm is applied to these degraded images and then the registered images are averaged and fed into the CNN. During training iterations images patches of size 61×61 are extracted from random positions in each training image. The corresponding truth patches of size 31×31 are also extracted for regression. The convolutional layers of the CNN do not use zero padding, so the processed image patches get smaller with each layer as shown in Table 3.2. This is why the truth patch is smaller than the input patch.

Table 3.2: Description of the FIFNet layers [2] used for the proposed BM-CNN TM method. The spatial dimensions of the activations are for training using 61×61 image patches.

Layers	Type	Spatial kernel	Activations
1	Input	-	$61 \times 61 \times 1$
2	Splitting	-	$61 \times 61 \times 1$
3, 4	2D Conv+ReLU	3×3	$59 \times 59 \times 64$
5, 6	2D Conv+ReLU	3×3	$57 \times 57 \times 64$
7, 8	2D Conv+ReLU	3×3	$55 \times 55 \times 64$
9, 10	2D Conv+ReLU	3×3	$53 \times 53 \times 64$
11, 12	2D Conv+ReLU	3×3	$51 \times 51 \times 64$
13, 14	2D Conv+ReLU	5×5	$47 \times 47 \times 64$
15, 16	2D Conv+ReLU	5×5	$43 \times 43 \times 64$
17, 18	2D Conv+ReLU	5×5	$39 \times 39 \times 64$
19, 20	2D Crop+ReLU	-	$39 \times 39 \times 1$
21	Concatenate	-	$39 \times 39 \times 65$
22, 23	2D Conv+ReLU	5×5	$35 \times 35 \times 65$
24	2D Crop	-	$35 \times 35 \times 1$
25	Concatenate	-	$35 \times 35 \times 66$
26	2D Conv	5×5	$31 \times 31 \times 1$
27	Regression	-	-

CNN updates during training are done using stochastic gradient descent with momentum optimization. Sixty four different random patches are extracted from each training image. We use a mini-batch size of 64 so that each iteration corresponds to one training image. All training is done with a total of 100 epochs. We set the initial learning rate to 0.1 and reduced this by a factor of 0.1 after every 10 epochs. The workstation used for training is equipped with an Intel(R) Core(TM) i9-9980XE CPU @ 3.00GHz with 128 GB of RAM. It is also equipped with two NVIDIA Titan RTX GPUs. The total training time for this network is approximately 100 minutes. The performance of this network is evaluated in depth in Chapter IV.

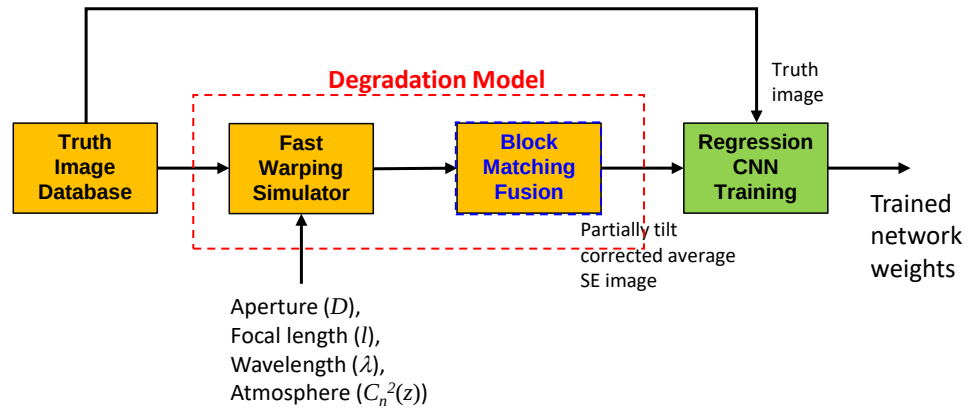


Figure 3.4: BM-CNN Training Block Diagram

CHAPTER IV

Experimental Results

The experimental results have all been conducted using the turbulence degradation parameters from Table 2.1 and the optical parameters from Table 2.2. The four different turbulence levels have all been trained using the fast warping simulator for noise standard deviation of 10 and 30 with 30 frames per image. The testing data follows the same data generation, but there are two different types of testing data: fast warping simulator and the anisoplanatic simulator. These will be referred to as Warp and Aniso respectively when seeing the results in the tables. As previously stated, while the anisoplanatic is more accurate, it takes a long time to generate the data, so it is only used for testing purposes.

4.1 Quantitative Results

To quantitatively compare results of the all TM methods, the metrics of peak signal-to-noise ratio (PSNR) [48] and structural similarity index measure of image quality (SSIM) [49] are used. For PSNR the higher the number the better. For SSIM the numbers range from 0 to 1. An SSIM of 1 means the two images are identical, so getting numbers close to 1 is desired. The quantitative error analysis is computed over the 100 validation images with ignore border of 5 pixels all around. The purpose of the ignore border is to account for possible border effects from the spatial processing. The border effects are due to running out of available pixels that surround a the pixel that is being processed at the borders. The ignore border is done for all methods to ensure that the tests are fair, but to also ensure no method is being penalized against something that is a known issue.

Tables 4.1 and 4.2 show the PSNR and SSIM for turbulence level 1. When looking at all the different TM methods for the PSNR results, it can be seen the that the BM-CNN

performs best on both Warp testing, and Aniso testing. This is the case for a noise standard deviation of 10 and 30. The top two performing methods are BM-WF and BM-CNN. When comparing the two on the “gold standard” Aniso testing data, the BM-CNN does 1.3369 dB better than BM-WF on noise 10 and 1.5818 dB on noise 30. For SSIM, again the BM-CNN is the best performing method. When comparing the BM-CNN to the BM-WF on Aniso testing, the former performs better by 0.0582 on noise 10 and 0.1028 on noise 30.

Tables 4.3 and 4.4 show the PSNR and SSIM for turbulence level 2. Again the BM-CNN is the best performing in all cases. The BM-CNN outperforms the BM-WF in PSNR by 0.9643 dB for noise 10 and 1.2356 dB for noise 30. For SSIM, the BM-CNN is better by 0.0562 for noise 10 and 0.0914 for noise 30.

Tables 4.5 and 4.6 show the PSNR and SSIM for turbulence level 3. When comparing the BM-CNN on the Aniso testing data with PSNR, the BM-CNN does 0.3408 dB better than BM-WF on noise 10 and 0.7392 on noise 30. When comparing the BM-CNN to the BM-WF on Aniso testing with SSIM, the former performs better by 0.0408 on noise 10 and 0.0782 on noise 30.

Following the trend of the other three turbulence levels, the BM-CNN comes out on top of BM-WF in turbulence level 4. In terms of PSNR, the BM-CNN is 0.1752 dB better for noise 10 and 0.2892 dB for noise 30. For SSIM, the BM-CNN is 0.0257 better with noise of 10 and 0.0333 better with noise 30.

For all four turbulence levels, the BM-CNN performs the best for PSNR and SSIM. Another thing to note is that the BM-CNN always has a bigger improvement over the BM-WF when the noise standard deviation is 30. From these tests, the BM-CNN is the best out of all the TM methods and it appears to do better with higher noise. We attribute this the

ability of the CNN to perform in a spatially varying manner and adapt to structural changes in the image. For example, in flat areas of the scene, the CNN can do more smoothing. On the other hand the fixed WF does not have this capability.

4.2 Qualitative Image Results

While it is useful to show that the BM-CNN performs better using metrics, it is also important that discernible difference can be seen with the human eye as well. Figures 4.1-4.4 show the truth and all TM method results for turbulence level 1. This is done with noise standard deviation of 10, for 30 frames, and for four different testing data images. When looking at these images, SE, LE, and LE-WF are easily seen to be worse than the BM-WF and BM-CNN. When comparing both of the BM methods, the content of the pictures look similar, but the BM-CNN is significantly less noisy than that of the BM-WF. This can most clearly be seen in Fig. 4.4 in the background of the sign.

Figures 4.5-4.8 show images for turbulence level 2 with noise 10 for 30 frames. Similar conclusions can be drawn from those made about the turbulence level 1 noise 10 images. Overall the images are slightly more blurry due to the increase in the $C_n^2(z)$ but are not much different. A real difference can be seen when the noise is increased to 30 for this same turbulence level. These images can be seen in Figures 4.9-4.12. The images look much more degraded and the benefit the BM-CNN over the BM-WF is much easier to see.

In Figures 4.13-4.16 the same process is repeated for turbulence level 3, with 30 frames and noise of 10. Here the it is much easier to see the shortcomings of the SE, LE, and LE-WF methods. Both BM methods do a much better job at TM. The BM-CNN does have better quality, but it looks as if it is due to it being better at denoising the image.

Table 4.1: Average PSNR (dB) results for 100 validation sequences with turbulence level 1 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	23.3489	23.2323	17.6345	17.6083
LE	26.1827	25.9293	25.3834	25.1803
BM	26.7063	26.4955	25.4702	25.3453
LE-WF	29.7577	29.7500	27.2442	27.1366
BM-WF	30.3766	30.3271	27.2241	27.2097
BM-CNN	31.8466	31.6640	28.8948	28.7915

Table 4.2: Average SSIM results for 100 validation sequences with turbulence level 1 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	0.5435	0.5576	0.2482	0.2627
LE	0.8060	0.8055	0.7005	0.7064
BM	0.8230	0.8250	0.6856	0.6969
LE-WF	0.8572	0.8652	0.7607	0.7701
BM-WF	0.8630	0.8717	0.7444	0.7585
BM-CNN	0.9264	0.9299	0.8556	0.8613

Table 4.3: Average PSNR (dB) results for 100 validation sequences with turbulence level 2 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	21.6393	21.4881	17.0607	17.0140
LE	24.1923	23.8974	23.6663	23.4118
BM	25.6641	25.5496	24.4865	24.4400
LE-WF	27.1175	26.9127	25.5973	25.3992
BM-WF	29.2104	29.3532	26.3433	26.4179
BM-CNN	30.2577	30.3175	27.5861	27.6535

Table 4.4: Average SSIM results for 100 validation sequences with turbulence level 2 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	0.4589	0.4702	0.2071	0.2197
LE	0.7086	0.7042	0.6088	0.6108
BM	0.7817	0.7887	0.6350	0.6515
LE-WF	0.7960	0.8009	0.7065	0.7125
BM-WF	0.8395	0.8542	0.7181	0.7373
BM-CNN	0.9006	0.9104	0.8158	0.8287

Table 4.5: Average PSNR (dB) results for 100 validation sequences with turbulence level 3 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	19.7836	19.5549	16.2892	16.1915
LE	21.7032	21.3931	21.4016	21.1164
BM	23.2272	23.1571	22.3702	22.3104
LE-WF	23.8516	23.4875	23.2291	22.9434
BM-WF	26.2424	26.5105	24.1303	24.2210
BM-CNN	26.5992	26.8513	24.8633	24.9602

Table 4.6: Average SSIM results for 100 validation sequences with turbulence level 3 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	0.3647	0.3697	0.1622	0.1706
LE	0.5597	0.5489	0.4733	0.4687
BM	0.6534	0.6644	0.5160	0.5296
LE-WF	0.6712	0.6673	0.6077	0.6067
BM-WF	0.7482	0.7772	0.6277	0.6493
BM-CNN	0.7936	0.8180	0.7095	0.7275

Table 4.7: Average PSNR (dB) results for 100 validation sequences with turbulence level 4 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	18.2356	17.9519	15.5183	15.3691
LE	19.5227	19.2212	19.3411	19.0539
BM	20.2898	20.1493	19.8932	19.7544
LE-WF	21.2001	20.7712	20.9670	20.6208
BM-WF	22.4137	22.3624	21.4067	21.3097
BM-CNN	22.5889	22.5473	21.6959	21.6129

Table 4.8: Average SSIM results for 100 validation sequences with turbulence level 4 and 30 input frames.

Mitigation	Simulated Data: Training / Testing			
	Noise Std = 10		Noise Std = 30	
	Warp/Warp	Warp/Aniso	Warp/Warp	Warp/Aniso
SE	0.2963	0.2973	0.1324	0.1376
LE	0.4618	0.4458	0.3870	0.3771
BM	0.4903	0.4839	0.3992	0.3966
LE-WF	0.5417	0.5283	0.5186	0.5078
BM-WF	0.5788	0.5911	0.5228	0.5251
BM-CNN	0.6033	0.6168	0.5540	0.5584

Lastly, the images are created for turbulence level 4, again doing 30 frames and noise 10 and can be seen in Figs. 4.17-4.20. At this level of turbulence the BM methods are the only ones that appear to provide much benefit for TM. Again the BM-CNN image looks less noisy than the BM-WF. This can be seen particularly around the chest of the penguin in Fig. 4.17. Looking at Fig. 4.19 the edges of the arches look a bit more defined in the BM-CNN as well.

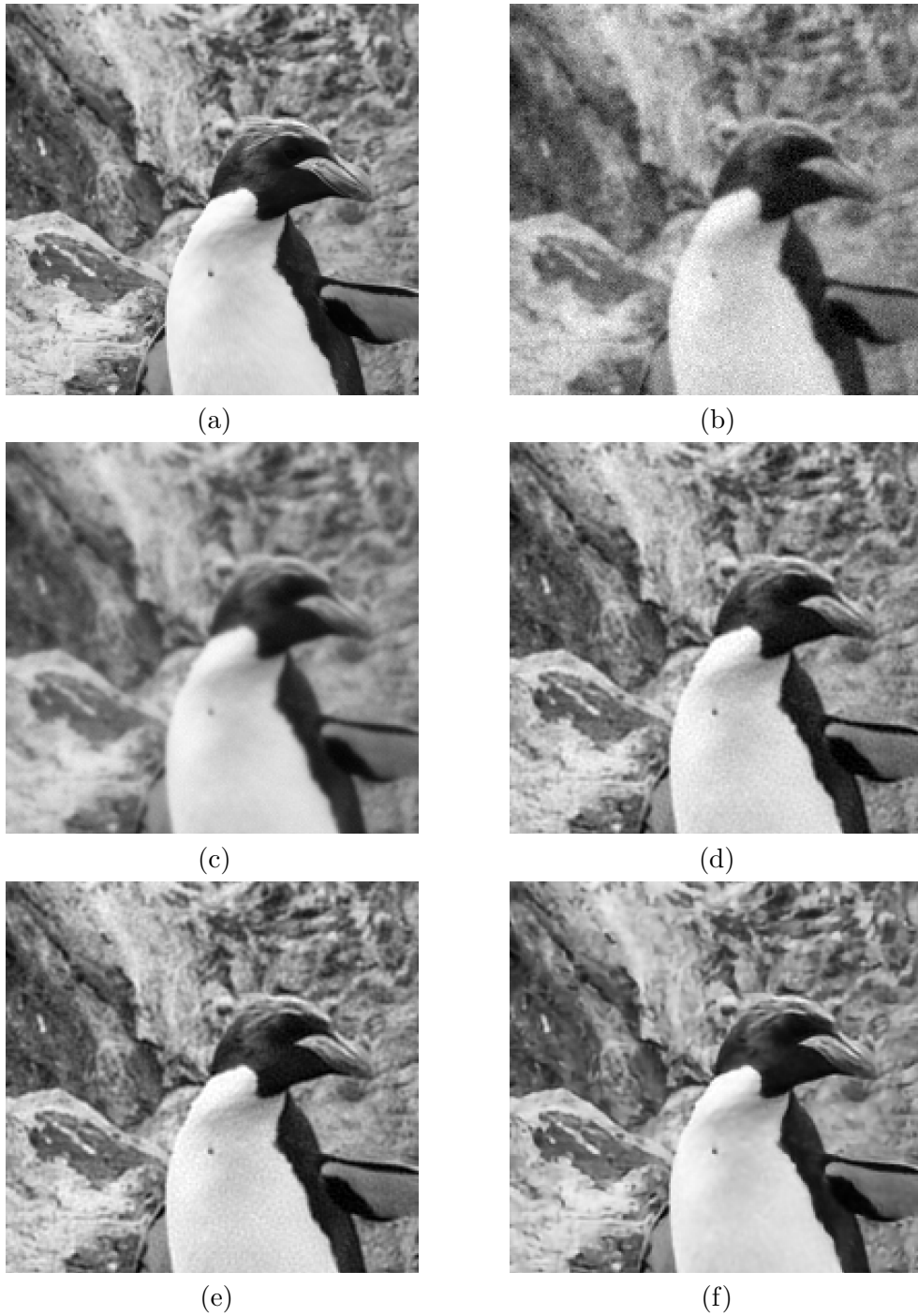


Figure 4.1: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 1, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.2: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.3: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.4: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 1 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

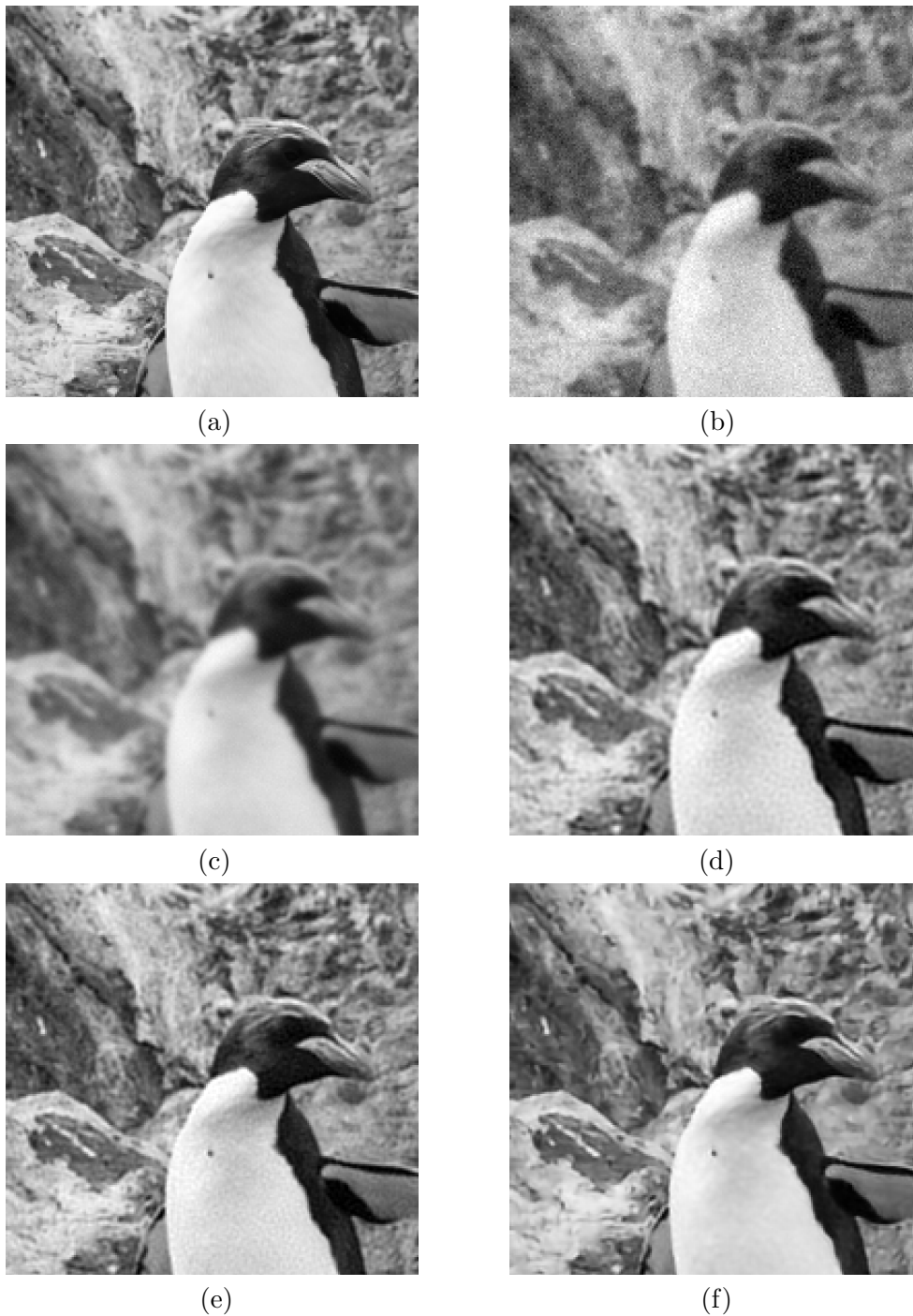


Figure 4.5: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 2, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

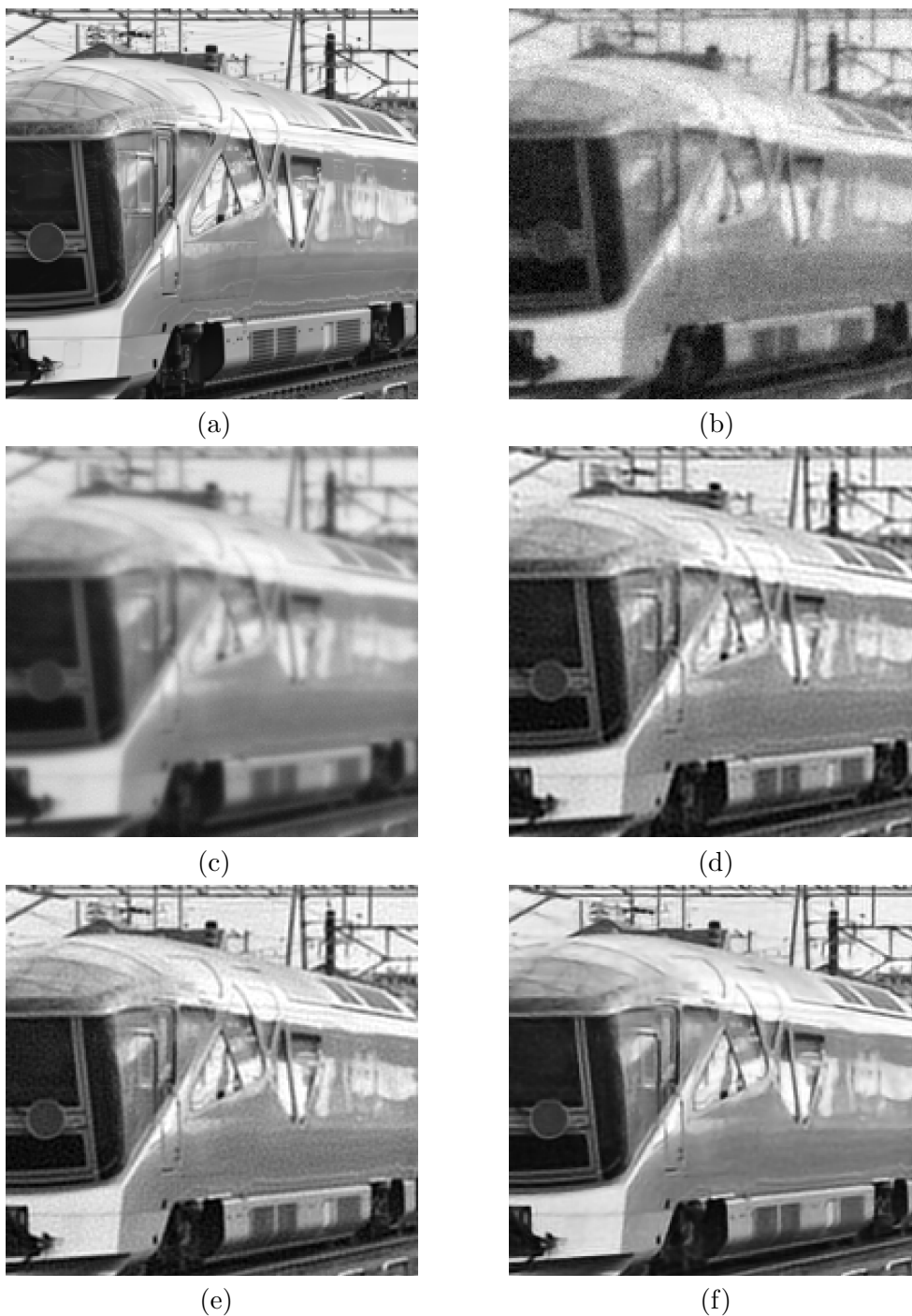
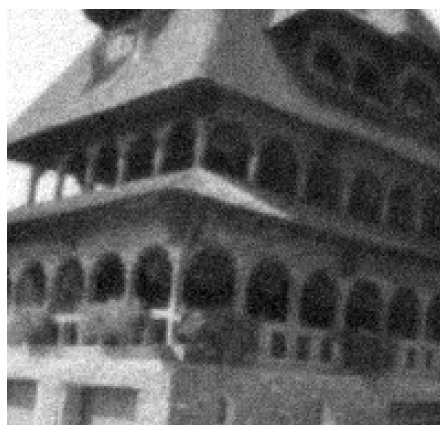


Figure 4.6: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.7: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.8: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

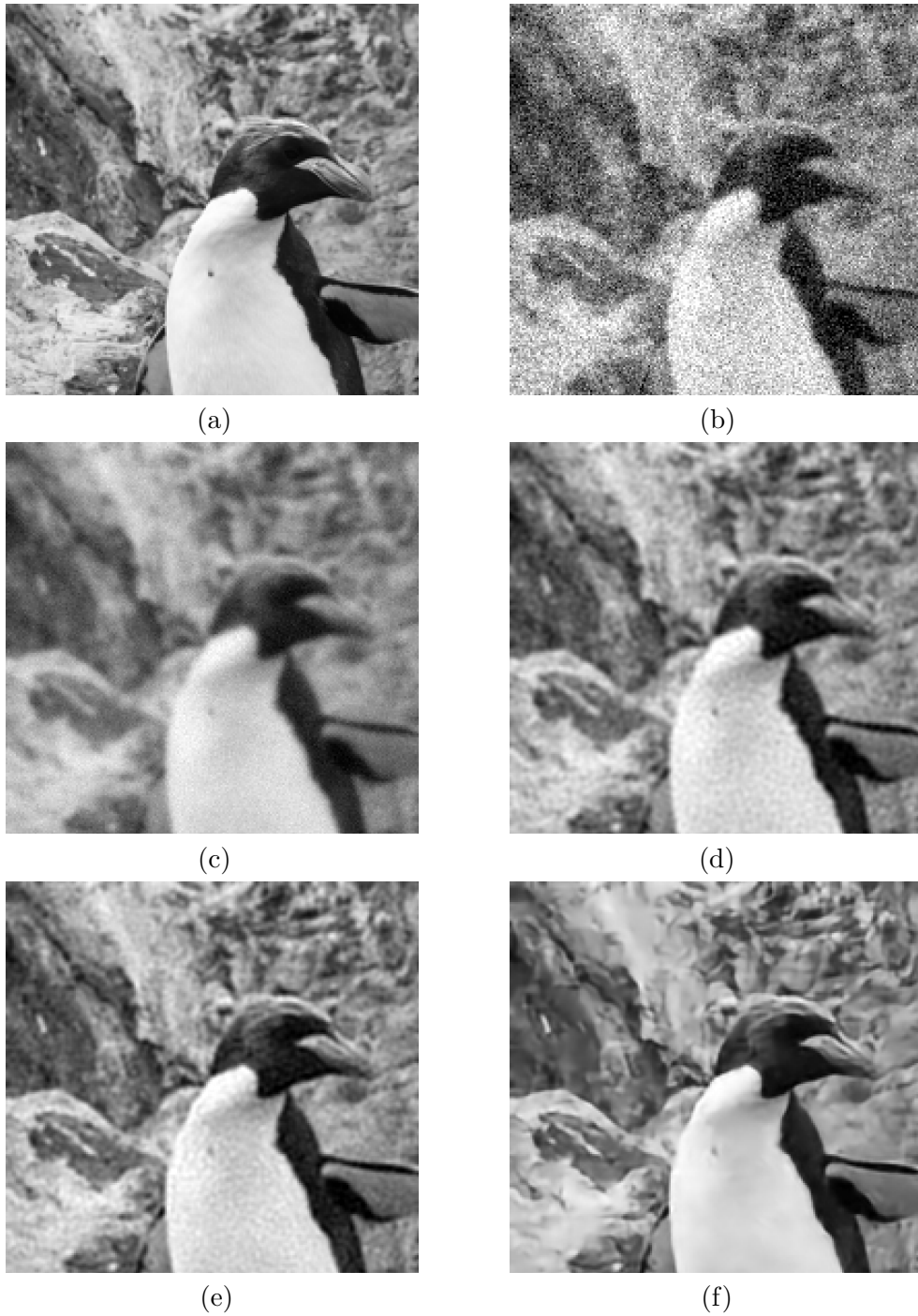
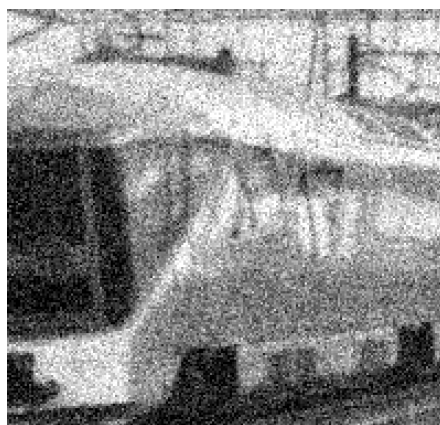


Figure 4.9: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 2, 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.10: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.11: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.12: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 2 and 30 input frames, and noise standard deviation of 30. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

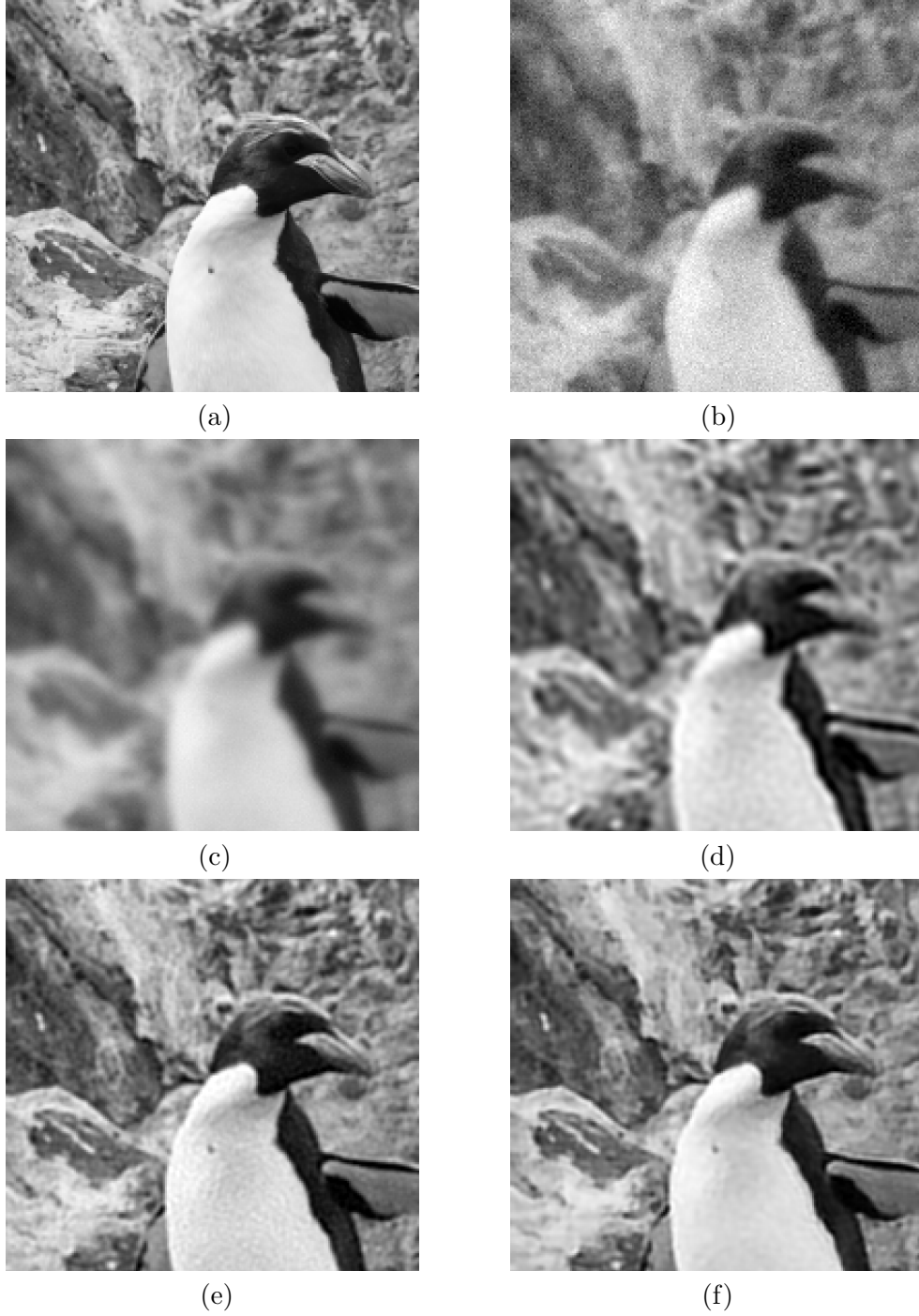


Figure 4.13: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 3, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

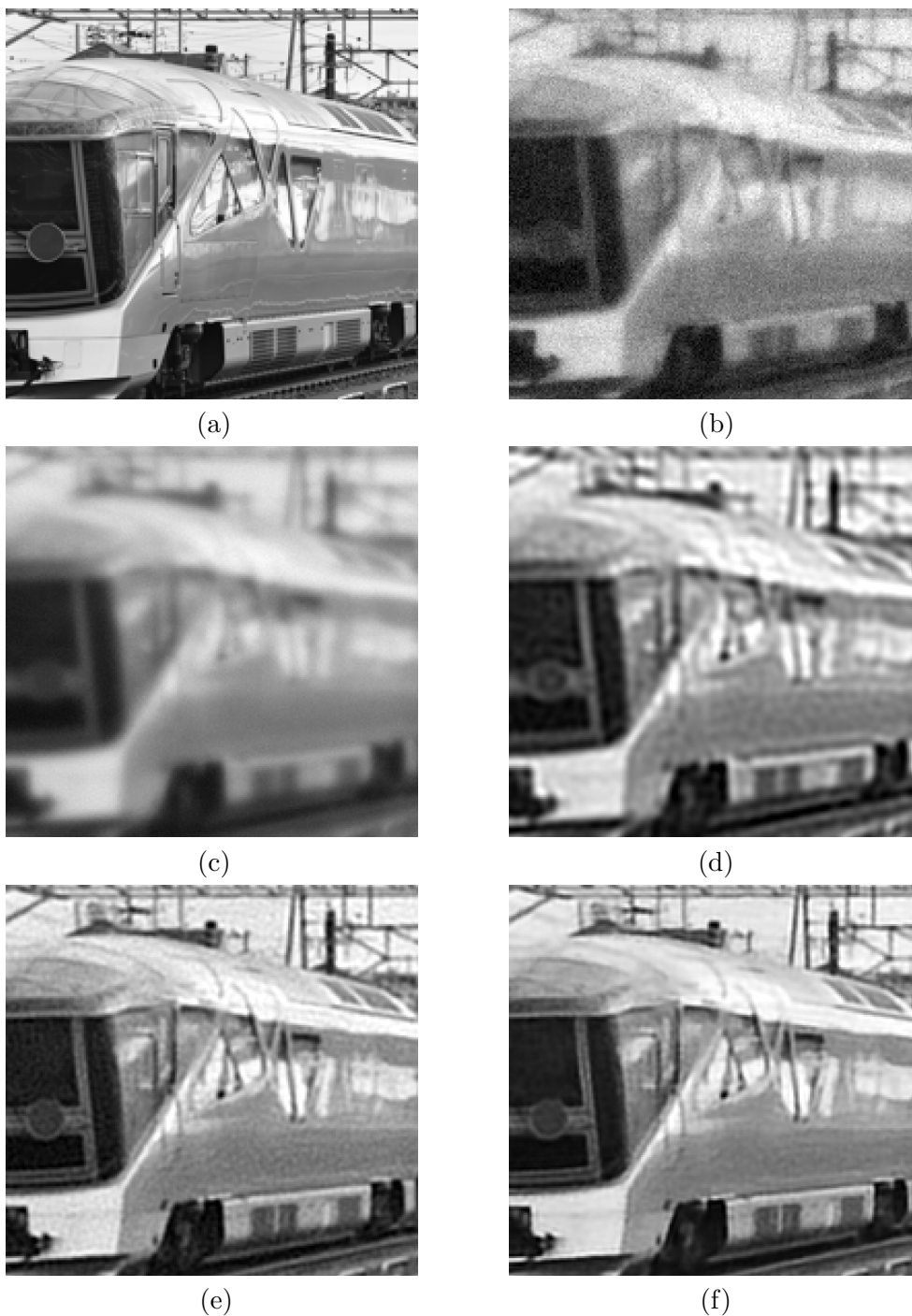


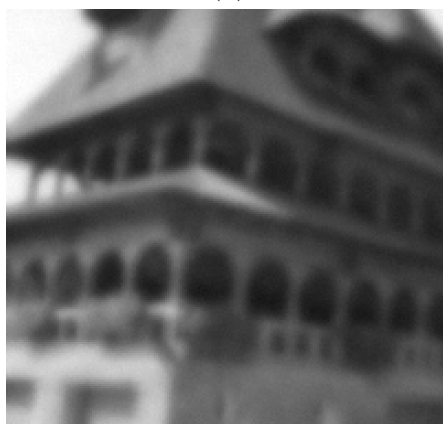
Figure 4.14: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.15: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.16: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 3 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

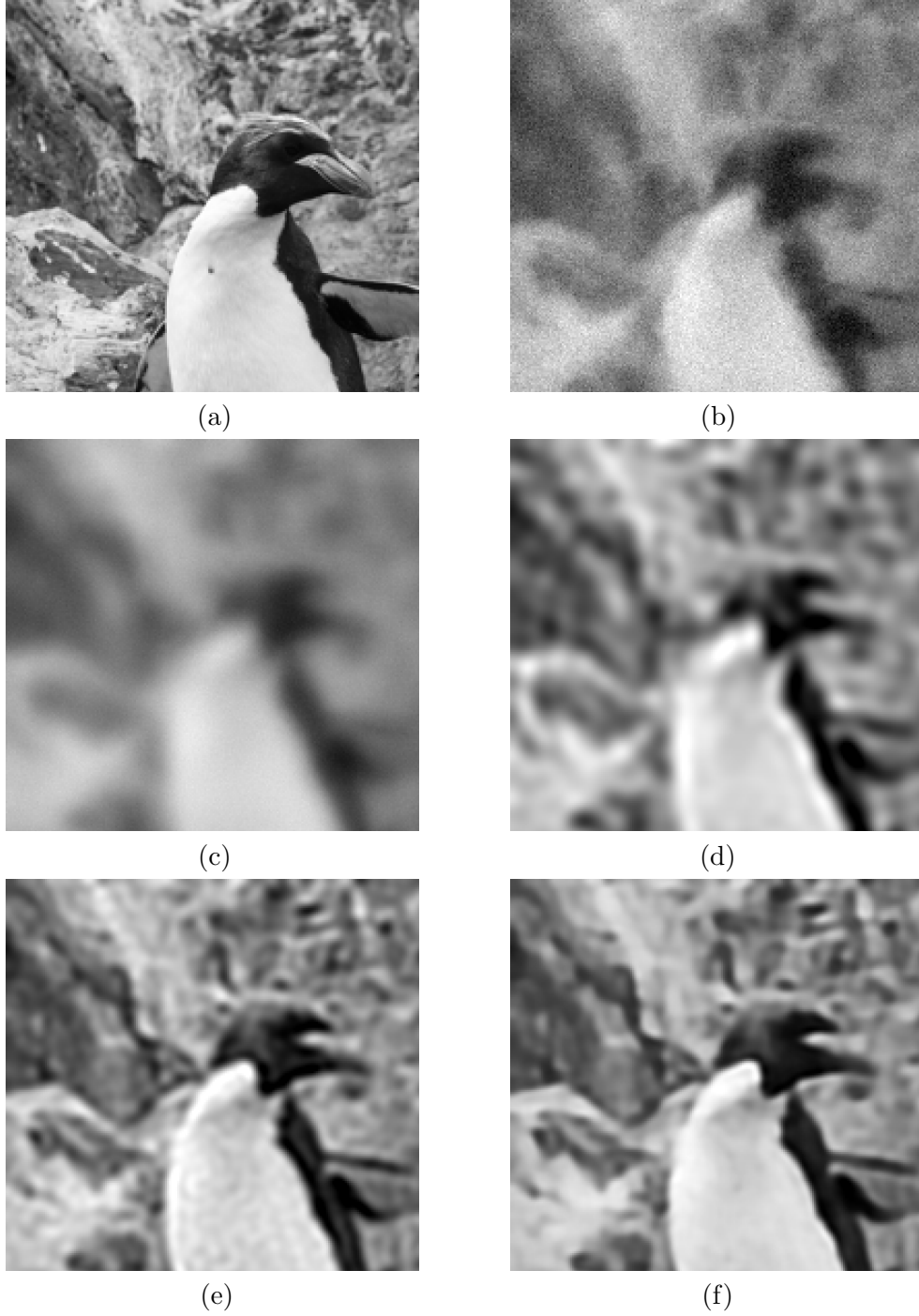


Figure 4.17: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 1. These results are for turbulence level 4, 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

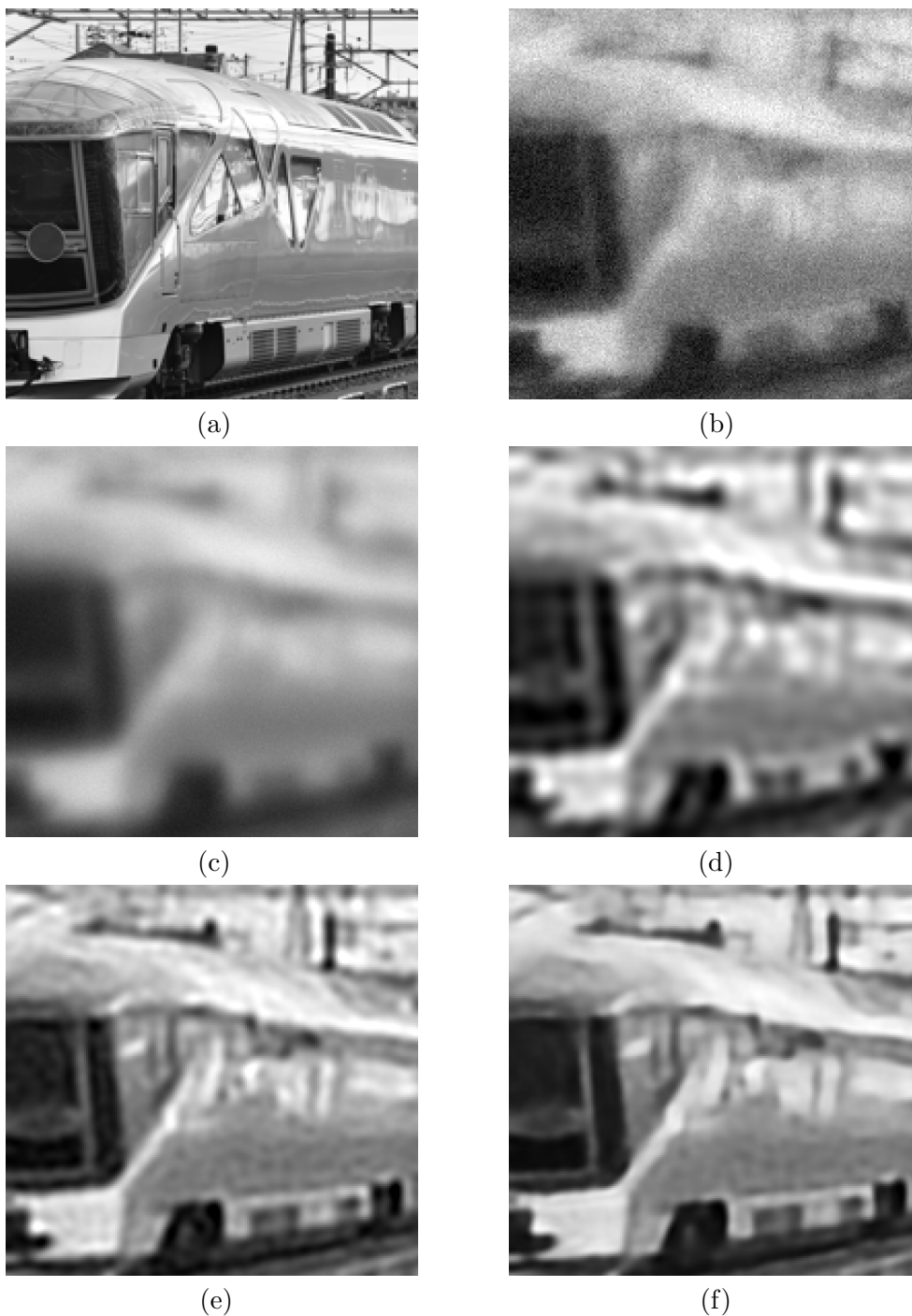


Figure 4.18: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 2. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

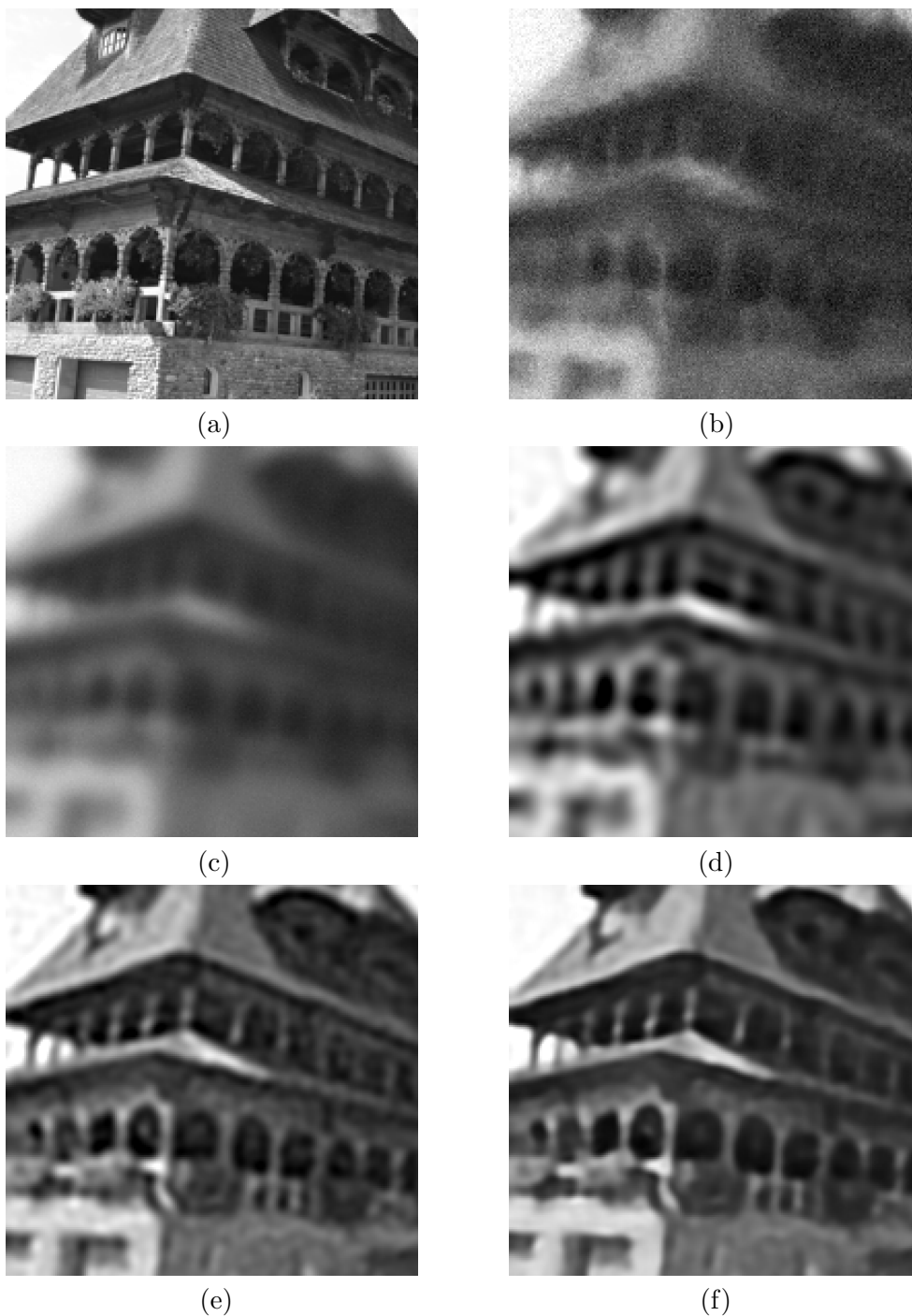
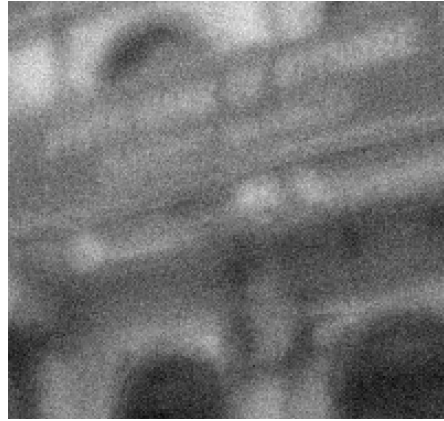


Figure 4.19: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 3. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.



(a)



(b)



(c)



(d)



(e)



(f)

Figure 4.20: Turbulence mitigation results using the anisoplanatic simulator data as testing data Image 4. These results are for turbulence level 4 and 30 input frames, and noise standard deviation of 10. (a) Truth, (b) SE, (c) LE, (d) LE-WF, (e) BM-WF, (f) BM-CNN.

CHAPTER V

Conclusions

A deep learning approach to atmospheric turbulence is introduced in this thesis. We have proposed a new TM method that is based on BM-WF [1]. In our new approach, we have replaced the WF with a CNN and employ DL to train the network. Our network is based on FIFNet [2] which has a lightweight CNN architecture with relatively few layers. This allows us to train the network from scratch without the need for transfer learning in approximately 100 minutes.

Our network is trained on data generated by a fast warping simulator. The fast warping simulator is nearly 100 times faster than that of the “gold standard” anisoplanatic simulator [3]. Our fast simulator is based on the method described by Schwartzman *et al* [28] but includes some novel modifications. These include new tilt correlation statistics and the addition of the average short exposure LSI PSF blurring, as described in Chapter II.

Testing is done here using the fast warping simulator and the anisoplanatic simulator. The BM-CNN trained on the warping simulator gives almost identical testing results for testing data generated by both the warping simulator and the anisoplanatic simulator. We believe this shows that computationally faster method is a good method for training. We believe it is very significant that we have demonstrated that the fast simulator can be effectively used for training in this application. The use of the fast simulator dramatically increases the speed of training data generation and makes the DL process practical in this application.

Our proposed network is compared to several benchmark methods in Chapter IV. In all of the quantitative performance experiments conducted in this thesis, the BM-CNN was the

best performing method. The BM-WF was the next best performing method. In subjective evaluation of the images, we have observed that the big advantage of the BM-CNN appears to be in how it handles noise. The BM-CNN does more noise suppression in the flat areas of the imagery and less in dynamic areas. This gives a very pleasing result and improved performance metrics. We believe our study has demonstrated the potential benefit of DL in the tackling the problem of TM.

BIBLIOGRAPHY

- [1] R. C. Hardie, M. A. Rucci, A. J. Dapore, and B. K. Karch, “Block matching and Wiener filtering approach to optical turbulence mitigation and its application to simulated and real imagery with quantitative error analysis,” *Optical Engineering*, vol. 56, no. 7, pp. 1 – 16, 2017. [Online]. Available: <https://doi.org/10.1117/1.OE.56.7.071503>
- [2] H. Elwarfalli and R. C. Hardie, “Fifnet: A convolutional neural network for motion-based multiframe super-resolution using fusion of interpolated frames,” *Computer Vision and Image Understanding*, vol. 202, p. 103097, 2021.
- [3] R. C. Hardie, J. D. Power, D. A. LeMaster, D. R. Droege, S. Gladysz, and S. Bose-Pillai, “Simulation of anisoplanatic imaging through optical turbulence using numerical wave propagation with new validation analysis,” *Optical Engineering*, vol. 56, no. 7, pp. 1 – 16, 2017. [Online]. Available: <https://doi.org/10.1117/1.OE.56.7.071502>
- [4] M. C. Roggemann and B. M. Welsh, *Imaging through Turbulence*, ser. Laser and Optical Science and Technology. CRC Press, January 1996.
- [5] R. E. Hufnagel and N. R. Stanley, “Modulation transfer function associated with image transmission through turbulent media,” *J. Opt. Soc. Am.*, vol. 54, no. 1, pp. 52–61, Jan 1964. [Online]. Available: <http://www.osapublishing.org/abstract.cfm?URI=josa-54-1-52>
- [6] A. W. M. van Eekeren, K. Schutte, J. Dijk, P. B. W. Schwering, M. van Iersel, and N. J. Doelman, “Turbulence compensation: an overview,” *Proc. SPIE*, vol. 8355, pp. 83 550Q–83 550Q–10, 2012. [Online]. Available: <http://dx.doi.org/10.1117/12.918544>
- [7] A. Lambert, F. Li, D. Bowman, and D. Fraser, “Experiments in turbulence induced super-resolution in surveillance imagery,” in *28th International Congress on High-Speed Imaging and Photonics*, H. Kleine and M. P. B. Guillen, Eds., vol. 7126, International Society for Optics and Photonics. SPIE, 2009, pp. 359 – 368. [Online]. Available: <https://doi.org/10.1117/12.823245>
- [8] Y. Liu, Ph.D. dissertation, 2016. [Online]. Available: <http://unsworks.unsw.edu.au>
- [9] C. L. Matson, K. J. Schulze, P. A. Billings, and D. W. Tyler, “Multiframe blind deconvolution and bispectrum processing of atmospherically degraded data: a comparison,” *Proc. SPIE*, vol. 4792, pp. 55–66, 2002. [Online]. Available: <http://dx.doi.org/10.1117/12.451796>
- [10] G. E. Archer, J. P. Bos, and M. C. Roggemann, “Reconstruction of long horizontal-path images under anisoplanatic conditions using multiframe blind deconvolution,” *Optical Engineering*, vol. 52, no. 8, pp. 083 108–083 108, 2013. [Online]. Available: <http://dx.doi.org/10.1117/1.OE.52.8.083108>

- [11] —, “Comparison of bispectrum, multiframe blind deconvolution and hybrid bispectrum-multiframe blind deconvolution image reconstruction techniques for anisoplanatic, long horizontal-path imaging,” *Optical Engineering*, vol. 53, no. 4, p. 043109, 2014. [Online]. Available: <http://dx.doi.org/10.1117/1.OE.53.4.043109>
- [12] D. Fraser, G. Thorpe, and A. Lambert, “Atmospheric turbulence visualization with wide-area motion-blur restoration,” *J. Opt. Soc. Am. A*, vol. 16, no. 7, pp. 1751–1758, Jul 1999. [Online]. Available: <http://josaa.osa.org/abstract.cfm?URI=josaa-16-7-1751>
- [13] D. H. Frakes, J. W. Monaco, and M. J. T. Smith, “Suppression of atmospheric turbulence in video using an adaptive control grid interpolation approach,” *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 3, pp. 1881–1884 vol.3, 2001.
- [14] D. Li, R. M. Mersereau, and S. Simske, “Atmospheric turbulence-degraded image restoration using principal components analysis,” *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 3, pp. 340–344, July 2007.
- [15] X. Zhu and P. Milanfar, “Image reconstruction from videos distorted by atmospheric turbulence,” *Proc. SPIE*, vol. 7543, pp. 75 430S–75 430S–8, 2010. [Online]. Available: <http://dx.doi.org/10.1117/12.840127>
- [16] —, “Removing atmospheric turbulence via space-invariant deconvolution,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 157–170, Jan 2013.
- [17] T. W. Lawrence, J. P. Fitch, D. M. Goodman, N. A. Massie, and R. J. Sherwood, “Experimental validation of extended image reconstruction using bispectral speckle interferometry,” *Proc. SPIE*, vol. 1237, pp. 522–537, 1990. [Online]. Available: <http://dx.doi.org/10.1117/12.19323>
- [18] T. W. Lawrence, J. P. Fitch, D. M. Goodman, N. A. Massie, R. J. Sherwood, and E. M. Johansson, “Extended-image reconstruction through horizontal path turbulence using bispectral speckle interferometry,” *Optical Engineering*, vol. 31, no. 3, pp. 627–636, 1992. [Online]. Available: <http://dx.doi.org/10.1117/12.56083>
- [19] C. J. Carrano, “Speckle imaging over horizontal paths,” *Proc. SPIE*, vol. 4825, pp. 109–120, 2002. [Online]. Available: <http://dx.doi.org/10.1117/12.453519>
- [20] —, “Anisoplanatic performance of horizontal-path speckle imaging,” *Proc. SPIE*, vol. 5162, pp. 14–27, 2003. [Online]. Available: <http://dx.doi.org/10.1117/12.508082>
- [21] J. P. Bos and M. C. Roggemann, “Mean squared error performance of speckle-imaging using the bispectrum in horizontal imaging applications,” *Proc. SPIE*, vol. 8056, pp. 805 603–805 603–11, 2011. [Online]. Available: <http://dx.doi.org/10.1117/12.884093>

- [22] J. P. Bos and M. C. Roggeman, “The effect of free parameter estimates on the reconstruction of turbulence corrupted images using the bispectrum,” *Proc. SPIE*, vol. 8161, pp. 816 105–816 105–9, 2011. [Online]. Available: <http://dx.doi.org/10.1117/12.893859>
- [23] J. P. Bos and M. C. Roggemann, “Robustness of speckle-imaging techniques applied to horizontal imaging scenarios,” *Optical Engineering*, vol. 51, no. 8, pp. 083 201–1–083 201–18, 2012. [Online]. Available: <http://dx.doi.org/10.1117/1.OE.51.8.083201>
- [24] —, “Blind image quality metrics for optimal speckle image reconstruction in horizontal imaging scenarios,” *Optical Engineering*, vol. 51, no. 10, pp. 107 003–1–107 003–9, 2012. [Online]. Available: <http://dx.doi.org/10.1117/1.OE.51.10.107003>
- [25] J. P. Bos and M. C. Roggeman, “Simulation of extended scenes imaged through turbulence over horizontal paths,” *Proc. SPIE*, vol. 8161, pp. 816 106–816 106–7, 2011. [Online]. Available: <http://dx.doi.org/10.1117/12.893869>
- [26] J. P. Bos and M. C. Roggemann, “Technique for simulating anisoplanatic image formation over long horizontal paths,” *Optical Engineering*, vol. 51, no. 10, October 2012.
- [27] G. Monnier, F.-R. Duval, and S. Amram, “GPU-based simulation of optical propagation through turbulence for active and passive imaging,” pp. 92 421R–92 421R–12, 2014. [Online]. Available: <http://dx.doi.org/10.1117/12.2067492>
- [28] A. Schwartzman, M. Alterman, R. Zamir, and Y. Y. Schechner, “Turbulence-induced 2d correlated image distortion,” in *2017 IEEE International Conference on Computational Photography (ICCP)*. Los Alamitos, CA, USA: IEEE Computer Society, may 2017, pp. 1–13. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/ICCPHOT.2017.7951490>
- [29] V. Jain, J. F. Murray, F. Roth, S. Turaga, V. Zhigulin, K. L. Briggman, M. N. Helmsstaedter, W. Denk, and H. S. Seung, “Supervised learning of image restoration with convolutional networks,” in *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.
- [30] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep cnn denoiser prior for image restoration,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [31] X. Mao, C. Shen, and Y.-B. Yang, “Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections,” in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 2802–2810. [Online]. Available: <http://papers.nips.cc/paper/6172-image-restoration-using-very-deep-convolutional-encoder-decoder-networks-with-symmetric-skip-connections.pdf>
- [32] Y. Chang, L. Yan, H. Fang, S. Zhong, and W. Liao, “Hsi-denet: Hyperspectral image restoration via convolutional neural network,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 2, pp. 667–682, 2019.

- [33] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, “Denoising prior driven deep neural network for image restoration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 10, pp. 2305–2318, 2019.
- [34] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” 2015.
- [35] X. Ji, Y. Lu, and L. Guo, “Image super-resolution with deep convolutional neural network,” in *2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*, 2016, pp. 626–630.
- [36] M. Kawulok, P. Benecki, S. Piechaczek, K. Hrynczenko, D. Kostrzewa, and J. Nalepa, “Deep learning for multiple-image super-resolution,” *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 6, p. 1062–1066, 2020.
- [37] J. Chen, X. He, H. Chen, Q. Teng, and L. Qing, “Single image super-resolution based on deep learning and gradient transformation,” in *2016 IEEE 13th International Conference on Signal Processing (ICSP)*, 2016, pp. 663–667.
- [38] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” *CoRR*, vol. abs/1511.04587, 2015. [Online]. Available: <http://arxiv.org/abs/1511.04587>
- [39] Mathworks, “combine.” [Online]. Available: <https://www.mathworks.com/help/images/single-image-super-resolution-using-deep-learning.html>
- [40] J. Gao, N. Anantrasirichai, and D. Bull, “Atmospheric turbulence removal using convolutional neural network,” 2019.
- [41] J. D. Schmidt, *Numerical Simulation of Optical Wave Propagation with Examples in MATLAB*, ser. SPIE Press Monograph. SPIE Press, August 2010.
- [42] R. C. Hardie, M. Rucci, B. K. Karch, A. J. Dapore, D. R. Droege, and J. C. French, “Fusion of interpolated frames superresolution in the presence of atmospheric optical turbulence,” *Optical Engineering*, vol. 58, no. 8, pp. 1 – 16, 2019. [Online]. Available: <https://doi.org/10.1117/1.OE.58.8.083103>
- [43] R. K. . Tyson, *Adaptive Optics Engineering Handbook*. CRC Press, August 1999.
- [44] L.-K. Yu, Z.-H. Hou, S.-C. Zhang, X. Jing, and Y. Wu, “Isoplanatic angle in finite distance: experimental validation,” *Optical Engineering*, vol. 54, no. 2, p. 024105, 2015. [Online]. Available: <http://dx.doi.org/10.1117/1.OE.54.2.024105>
- [45] J. W. Goodman, *Introduction to Fourier Optics*, 3rd ed. Roberts and Company Publishers, December 2004.

- [46] S. R. Bose-Pillai, J. E. McCrae, C. A. Rice, R. A. Wood, C. E. Murphy, and S. T. Fiorino, “Estimation of atmospheric turbulence using differential motion of extended features in time-lapse imagery,” *Optical Engineering*, vol. 57, no. 10, pp. 1 – 12, 2018. [Online]. Available: <https://doi.org/10.1117/1.OE.57.10.104108>
- [47] E. Agustsson and R. Timofte, “Ntire 2017 challenge on single image super-resolution: Dataset and study,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1122–1131.
- [48] “psnr.” [Online]. Available: <https://www.mathworks.com/help/images/ref/psnr.html>
- [49] “ssim.” [Online]. Available: <https://www.mathworks.com/help/images/ref/ssim.html>