

ADAPTIVE MESH-GRID BASED SPATIALLY NON-UNIFORM VIDEO
MOTION DEBLURRING USING IMU-CAMERA SENSOR FUSION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AHMET ARSLAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2021

Approval of the thesis:

**ADAPTIVE MESH-GRID BASED SPATIALLY NON-UNIFORM VIDEO
MOTION DEBLURRING USING IMU-CAMERA SENSOR FUSION**

submitted by **AHMET ARSLAN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Engineering** _____

Prof. Dr. Afşar Saranlı
Supervisor, **Electrical and Electronics Engineering , METU** _____

Assist. Prof. Dr. Gökhan Koray Gültekin
Co-supervisor, **Electrical and Electronics Engineering, AYBÜ** _____

Examining Committee Members:

Prof. Dr. A. Aydın Alatan
Electrical and Electronics Engineering, METU _____

Prof. Dr. Afşar Saranlı
Electrical and Electronics Engineering, METU _____

Prof. Dr. Gözde Bozdağı Akar
Electrical and Electronics Engineering, METU _____

Assoc. Prof. Dr. Fatih Kamışlı
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Gökhan Koray Gültekin
Electrical and Electronics Engineering, AYBÜ _____

Date: 06.09.2021



I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Ahmet Arslan

Signature :

ABSTRACT

ADAPTIVE MESH-GRID BASED SPATIALLY NON-UNIFORM VIDEO MOTION DEBLURRING USING IMU-CAMERA SENSOR FUSION

Arslan, Ahmet

M.S., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Afşar Saranlı

Co-Supervisor: Assist. Prof. Dr. Gökhan Koray Gültekin

September 2021, 81 pages

Motion blur is a common problem that produces blurry images with inevitable information loss and unpleasant visual quality. Relative motion between scene and image recording device during exposure time causes the motion blur. This artifact degrades the human perceptual quality and also challenges subsequent computer vision applications. Most of the studies in the literature assume uniform blur present in the image and focuses on removing motion blur with non-blind deconvolution algorithms because of their low computational cost. Low-light conditions and rotation around the optical axis violate this uniform motion blur assumption and decrease the deblurring performances. In this thesis, we developed a novel adaptive mesh-grid based motion deblurring algorithm to handle non-uniform motion blur and the high computation cost of existing methods. We use an inertial measurement unit (IMU) for obtaining the motion during the exposure period and generate a point spread function (PSF). The mesh-grid implementation is simply dividing an image into sub-block and performing blur kernel estimation and deblurring separately for each block. In addition, a variance parameter is proposed for representing the level of blur non-uniformity.

To increase PSF estimation accuracy, mesh-size changed according to this variance parameter. Using this approach, two adaptive mesh-size algorithms are developed for the best quality and optimal quality with low computation time. It is observed that the proposed adaptive mesh-size algorithm increases the performance and quality of the existing methods.

Keywords: Motion Blur, Motion Deblurring, PSF Estimation, Mesh-Grid



ÖZ

IMU-KAMERA SENSÖR BİRLEŞİMİ KULLANILARAK DEĞİŞKEN BOYUTLU MAZGAL YAPISI İLE DOĞRUSAL OLMAYAN VIDEO HAREKET BULANIKLIĞI GİDERME

Arslan, Ahmet

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Afşar Saranlı

Ortak Tez Yöneticisi: Dr. Öğr. Üyesi. Gökhan Koray Gültekin

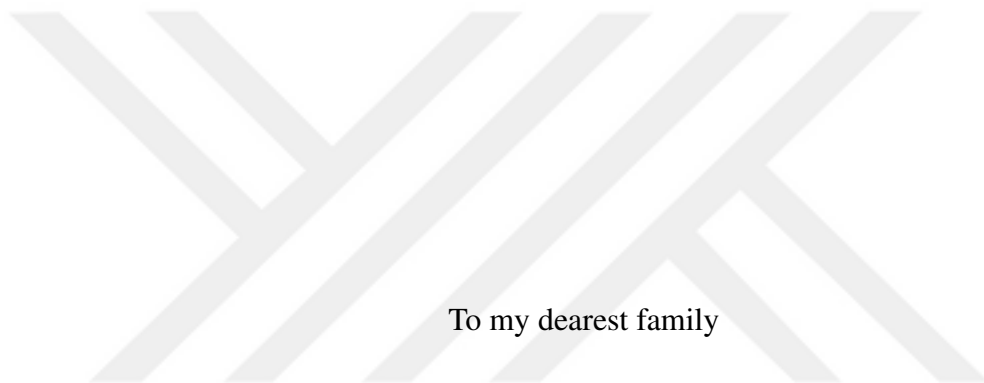
Eylül 2021 , 81 sayfa

Hareket bulanıklığı, görüntülerde bilgi kaybına ve görsel kalite düşüklüğüne sahip bulanık görüntüler üreten yaygın bir sorundur. Pozlama süresi boyunca sahne ve görüntü kayıt cihazı arasındaki kısmi hareket, hareket bulanıklığına neden olur. Bu bozunum, insanın görsel algı kalitesini düşürür ve aynı zamanda görüntü işleme algoritmalarının performanslarına etki eder. Literatürdeki çalışmaların çoğu, görüntüde tek tip bulanıklık olduğunu varsaymakta ve düşük işlem süreleri nedeniyle kör olmayan ters evrişim algoritmalarıyla hareket bulanıklığını gidermeye odaklanmaktadır. Düşük ışık koşulları ve optik eksen etrafındaki dönüş, bu tek tip hareket bulanıklığı varsayımını ihlal eder ve bulanıklık giderme algoritmalarının performanslarını azaltır. Bu tezde, düzgün olmayan hareket bulanıklığını ve mevcut yöntemlerin yüksek işlem sürelerini ele almak için yeni bir değişken boyutlu, ağ-ızgara tabanlı bulanıklık giderme algoritması geliştirdik. Pozlama süresi boyunca hareketi elde etmek için bir atalet ölçüm birimi (IMU) kullanıyoruz ve bir nokta yayılma fonksiyonu (PSF) oluşturuyoruz. Kafes ızgara uygulaması, bir görüntüyü alt bloklara bölmek ve her blok için ayrı ayrı PSF

tahmini ile bulanıklık giderme gerçekleştirmektir. Ek olarak, bulanıklığın düzgün olmama düzeyini temsil etmek için bir varyans parametresi önerilmiştir. PSF tahmin doğruluğunu artırmak için ağ boyutu bu varyans parametresine göre değiştirilmiştir. Bu yaklaşımı kullanarak, düşük işlem süresi ile en iyi kalite ve optimum kalite için iki değişken ağ boyutu algoritması geliştirilmiştir. Önerilen değişken ağ boyutu algoritmalarının mevcut yöntemlerin performansını ve kalitesini arttırdığı gözlemlenmiştir.

Anahtar Kelimeler: Hareket Bulanıklığı, Hareket Bulanıklığı giderme, PSF Tahmini, Mazgal





To my dearest family

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude and sincere appreciation to my supervisors, Prof. Dr. Afşar Saranlı and Assist. Prof. Dr. Gökhan Koray Güldekin for their continuous support and invaluable guidance. This study was able to emerge thanks to their valuable knowledge on the subject, years of academic experience and unique approach to problems that we encounter.

I would like to thank to Prof. Dr. Aydın Alatan for his spot-on comments that make me go in the right direction, advice and all the contributions during the time we spent together while doing projects.

I would like to thank the ROKETSAN Inc. for supporting our project and encouraging the postgraduate education with providing all kinds of assistance.

I would like to express my gratitude to Alper Kahvecioğlu from ROKETSAN company for providing an excellent working environment and enabling me to complete my work efficiently by giving all kinds of support generously.

I would like to express my thanks to Birkan Kurşun, Arda Özertem and Tuncer Soyer from ROKETSAN company for their patience, technical support and guidance.

Thanks to Hazal Özen for her moral support and being always with me. Thanks to her faith and trust, I found the strength to continue and complete my studies.

I would like to thank Ufuk Efe, who has an inseparable place in my business, academic and social life and has made significant contributions to my life.

Finally, I would like to thank my family for bringing me to where I am today and always being there for me with their endless support. None of this would have been even possible without the love and patience of them.

This work is financially supported by the Scientific and Technological Research Council of Turkey under BİDEB 2210-A program.

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
ACKNOWLEDGMENTS	x
TABLE OF CONTENTS	xi
LIST OF TABLES	xiv
LIST OF FIGURES	xv
LIST OF ABBREVIATIONS	xxii
CHAPTERS	
1 INTRODUCTION	1
1.1 Literature Review	2
1.2 Contributions	5
1.3 Outline of the Thesis	7
2 INERTIAL-AIDED NON-UNIFORM MOTION DEBLURRING	9
2.1 Mathematical Model of Motion Blur	9
2.1.1 Motion Blur Model as a Convolution	9
2.1.2 Point Spread Function	10
2.1.3 Inverse Filtering	11
2.1.4 Regularized Approaches	12

2.2	Inertial-aided Spatially Varying Motion Blur PSF Computation	15
2.2.1	Camera Model and Motion Projection	15
2.2.2	Inertial Measurements	19
2.2.3	PSF Estimation	20
3	PROPOSED SOLUTION FOR HANDLING NON-UNIFORM MOTION BLUR PROBLEM	25
3.1	Motivation	25
3.2	Mesh-Grid Decomposition for Non-Uniform Blur	27
3.2.1	Ringling Artifacts	30
3.2.2	Determination of Overlap Region	33
3.3	Proposed Method of Adaptive Mesh-Size	35
3.3.1	High-Quality Adaptive Mesh-Size Algorithm	36
3.3.2	Speed-Quality Balanced Adaptive Mesh-Size Algorithm	38
3.3.3	Experimental Determination of Speed-Quality Balanced Mesh- Size	39
4	EXPERIMENTS AND TEST RESULTS	45
4.1	Performance Metrics	45
4.2	Datasets	47
4.3	Evaluation of Non-Blind Algorithms	54
4.4	Experiments on Adaptive Mesh-Size Algorithm	60
4.4.1	Performance Evaluation of Adaptive Mesh-size Algorithms and Comparison with Classical Approach	60
4.4.2	Comparison of Adaptive and Constant Mesh-Size Structure .	63
4.4.3	Extending SQB Algorithm with a Tradeoff Control Parameter .	68

4.4.4	Observations and Limitations	70
5	CONCLUSIONS AND FUTURE STUDIES	75
	REFERENCES	77



LIST OF TABLES

TABLES

Table 2.1 Shows the comparison of some non-blind deconvolution methods according to likelihood and prior [33]	12
Table 4.1 PSNR and computation time results of the algorithms for Dataset-5 generated from kernel-1, kernel-2 and kernel-3. Average gain (dB) and time (sec) results are given.	55
Table 4.2 PSNR-gain (dB) and computation time (sec) results of adaptive and full-frame deblurring algorithms for ten frames selected on Dataset-2. . .	61
Table 4.3 Comparison of the constant mesh-size versus proposed SQB and HQ algorithm. The average PSNR gain and computation time results on Dataset-2, Dataset-3 and Dataset-4 are presented. (*) cases correspond to blur magnitude exceeding mesh-size deteriorating PSNR performance. . .	63

LIST OF FIGURES

FIGURES

- Figure 1.1 The DeepGyro CNN architecture. Rotational measurements are used for generating blur field which is equal to PSF [18]. 3
- Figure 1.2 (Left) A real-world image with spatially-variant motion blur. (Center) Deblurred image with constant mesh-sized deblurring method. (Right) Blur kernels calculated from IMU motion estimates [6]. 5
- Figure 2.1 Illustration of different regularization terms. Different prior functions penalize values differently. The Gaussian prior increases energy most quickly for large absolute values 14
- Figure 2.2 Coordinate axis definition for camera projection model. Z axis is principal axis, f is the focal length of the camera. World points are projected into image plane. 15
- Figure 2.3 Three pairs of images are shown which represent the motion fields and corresponding blurred images. PSFs generated from these motions and used to simulate blurred images. The blurred images given next to velocity field graphs. Rotational movements along the x axis (pitch), combination of the x axis and the y axis (yaw), the z axis (roll) are simulated top, middle and the bottom row respectively. Motions along the x and y axis are uniform while the z axis is non-uniform. The effect of motions can be observed from the blurred images. 18
- Figure 2.4 Camera motion trajectory generation process. K number of camera motion samples are obtained using the IMU measurements and corresponding motion trajectory in image plane is shown. 19

Figure 2.5	Inertial-measurements taken from smart-phone internal IMU. Only rotational motions are taken and combined into single data for observation.	20
Figure 2.6	Shows the histogram of the given data in Fig.2.5. Gaussian distribution fitted to this data which is collected from a stationary IMU device. PSF generated from Gaussian distribution and ideal one are represented in bottom images respectively.	21
Figure 2.7	PSF estimation from inertial measurements. Corresponding PSF generated from motion trajectory between start point end point using the (2.23),(2.27).The PSF shown in the right top corner.	22
Figure 2.8	Shows an estimation of non-linear PSF for given motion vectors in top graph. The motion consisting of 12 samples of IMU readings. Exposure period selected as 0.02 sec.	23
Figure 3.1	Motion vectors generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. Two red boxes are selected at top and bottom corners. Using the PSF generation formula (2.27), two corresponding PSFs generated and shown. The non-uniformity of the blur can be seen in PSFs and motion vectors.	26
Figure 3.2	Mesh-grid division process of image and introduction of overlap region. $[m, m]$ mesh-sized grids extended with overlap size n . Overlap region is introduced for removing ringing-artifacts. Overlap regions removed after deconvolution.	27
Figure 3.3	Image plane motion trajectory generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. The image size is $[480 \times 480]$, it is divided by $[40 \times 40]$ grids and estimated PSFs for each grid are shown in bottom figure. PSFs differ from region to region.	28

Figure 3.4	Image plane motion trajectory generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. The image size is [480x480], it is divided by [40x40] grids and estimated PSFs for each grid are shown in bottom figure. PSFs differ from region to region.	29
Figure 3.5	Original, blurred, and deblurred images are given at the top. Colored boxes from original and blurred ones are magnified in second and third rows respectively.	30
Figure 3.6	Shows the original and edge tapered images at the top figure. The effect of tapering can be seen in magnified parts of both images indexed with boxes.	31
Figure 3.7	Shows the deblurred image results with and without edge tapering at the top figure. The selected boxes are magnified and the decrease in the ringing artifacts can be seen in results.	32
Figure 3.8	Shows the blurred image and deblurred image with mesh-grid size of [60x60] pixels respectively. The mesh-grid artifacts appear on each grid's border.	33
Figure 3.9	Shows the deblurred image results with and without edge tapering and overlap region. The PSNR values given in dB. The overlap region started from 0 pixel to maximum blur size of the blur kernel. The best quality obtained with employing edge-tapering and overlap size of MBS.	34
Figure 3.10	Adaptive mesh-size illustration. As non-uniformity in blur decrease, computed frame variance also decreases. This allows us to increase mesh-size and decrease computational cost.	35
Figure 3.11	Shows the one grid cell structure that goes through the deblurring process with overlap region. Blue area is removed after deconvolution and orange part stays for actual deblurred image. The overlap region is determined as MSB previously.	36

Figure 3.12	PSNR-Gain and processing time versus mesh-size graph of sample image with variance of 12-pixel. Computational cost increases exponentially while increase in PSNR-Gain is more linear. Selecting a mesh-size in green boxed area gives a reasonable computation time and gain.	38
Figure 3.13	Shows original, blurred and deblurred images with different mesh-sizes. The mesh-size information represented as 'M.size', PSNR-Gain in db as 'P.Gain', and computation time in seconds as 'C.Time'. The selection of the speed-quality balanced points conducted based on PSNR-Gain, computational cost and visual quality.	39
Figure 3.14	Variance versus Time and Mesh-size results of selected speed-quality balanced points from Dataset-1 (Explained in the Chapter 4, Section 4.2). The Selection process conducted with hand-labeling. Mesh-sizes are common divisors of 480x480 image.	41
Figure 3.15	An exponential curve is fitted to speed-Quality balanced data points in the Dataset-1 hand-labeled results shown in Fig.3.14	42
Figure 4.1	Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-1. Blur magnitude calculation conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.	48
Figure 4.2	Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-2. Blur magnitude calculation is conducted by selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.	49

Figure 4.3	Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-3. Blur magnitude calculation is conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.	50
Figure 4.4	Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-4. Blur magnitude calculation is conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.	51
Figure 4.5	Shows the three blur kernels which used to generated uniform blur Dataset-5 for evaluation of non-blind methods. Sample original and blurred images are provided in bottom figure.	53
Figure 4.6	The average and standard deviation of critical parameters	57
Figure 4.7	The average and standard deviation of critical parameters	58
Figure 4.8	The average and standard deviation of critical parameters	59
Figure 4.9	Shows blurred and deblurred images from Dataset-2 with different mesh-sizes. The mesh-size information represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in seconds as ‘C.Time’. The selection of the SQB mesh-size conducted based on PSNR-Gain, computational cost and visual quality. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and high cost.	62

Figure 4.10 Shows blurred and deblurred frames with different mesh-sizes from Dataset-3. The mesh-size is represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in sec as ‘T’. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and cost. The best quality is obtained in the HQ method and decreases from left to right except for frames with low variance. 65

Figure 4.11 Shows blurred and deblurred frames with different mesh-sizes from Dataset-4. The mesh-size is represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in sec as ‘T’. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and cost. The best quality is obtained in the HQ method and decreases from left to right except for frames with low variance. 66

Figure 4.12 Shows the avg. PSNR gains and computation times of constant mesh-size and SQB method for three datasets given in Table 4.3. Constant mesh-sizes are (40,48,60,80,96,120,160,240,480). The SQB (*) results are over the constant mesh-size curves in all three dataset which indicates better performance. The green circle indicates the desired region where maximum PSNR gain with minimum time. 67

Figure 4.13 Shows the avg. PSNR gains and computation times of constant mesh-size, MSQB and SQB method for three datasets given in Table 4.3. Constant mesh-sizes are (40,48,60,80,96,120,160,240,480). λ is changed between 0.1 to 4 with the step of 0.1 which generates MSQB curves. These curves are always above the constant mesh-size curves in three datasets which indicates better performance. The blue SQB points indicates $\lambda = 1$ cases. The SQB in Dataset-1 is lower the bending point of MSQB point and further improvement is possible. 69

Figure 4.14 Shows total processed pixel number versus time (left), mesh-size versus time/total processed pixel number graphs (right). The first graph indicates the total pixel number and computation time is almost linear which means that the processing time of one pixel for all mesh-sizes are almost equal. In the second one, as mesh-size decreases the total processed pixel number increases exponentially with overlap region, which results in increase of computation time. MS stands for mesh-size. 70

Figure 4.15 Shows the mesh-size/time versus variance graph of SQB points. The variance sample limit is increased from the previous version. The green boxed area shows the violation of the concept defined as increasing the variance decreases the mesh-size. This results in PSF estimation error and damages the restored image quality. 71

Figure 4.16 Shows the blurred frame-27 from Dataset-3, and restored images with HQ and full-frame mesh-sizes. The SQB and HQ are same in this situation because of high variance. In the restored frame with mesh-size 96, the borders of the grids are becoming visible as an artifact because of high-roll motion. 73

LIST OF ABBREVIATIONS

SLAM	Simultaneous Localization and Mapping
UAV	Unmanned Aerial Vehicle
PSF	Point Spread Function
IMU	Inertial Measurement Unit
ECP	Extreme Channel Prior
GAN	Generative Adversarial Network
CNN	Convolutional Neural Network
HQM	Half-Quadratic Minimization
2D	2 Dimensional
6D	6 Dimensional
MSB	Maximum Size of Blur
HQ	High Quality
SQB	Speed Quality Balanced
IQ	Image Quality

CHAPTER 1

INTRODUCTION

Image sequences obtained from mobile platforms with image capturing devices are the main source of computer vision and image processing applications. In most cases, mobile platforms do not provide the ideal environment and conditions for image capturing. One of the results of this nonideality is motion blur caused by camera motion during the exposure time. Motion blur affects the performance of image processing algorithms and lowers the visual quality of the image. This degeneration is perceived severely especially for fast-moving platforms and applications such as simultaneous localization and mapping (SLAM), visual odometry, and unmanned aerial vehicle (UAV) recordings.

In order to remove motion blur from the blurred images, the point spread function (PSF) and the latent image should be estimated from the blurred image. This problem is an ill-posed problem. To solve this problem motion deblurring algorithms are developed and divided into two categories according to prior information about the PSF, i.e., blur kernel. Blind deconvolution means that PSF is not available and both PSF and latent image must be estimated. Blind algorithms were generally slow and inconsistent especially under large and spatially-variant blur since they should estimate PSF along with the latent image. On the other hand, with recent developments in neural networks, deep learning-based researches, such as [1]–[3] has successfully improved the blind deblurring quality. One drawback of these algorithms is their high computational cost and therefore the need for powerful GPUs which cannot be available for all platforms.

Information about the motion of the camera can be obtained from the inertial measurement units (IMU) with gyroscope and accelerometers. The PSF can be estimated

from the IMU data. Then one of the deconvolution techniques can be used to recover sharp image by using the estimated PSF. Since the PSF is estimated very quickly without too much computation required, deblurring process takes little time with less error compared to blind-deconvolution based image deblurring techniques. Studies [4]–[6] successfully combined IMU information and non-blind deconvolution for recovering sharp image. However, spatial-variance and computation time is still problem for these algorithms.

In this thesis, we present a deblurring algorithm that divides the blurred image into grids and estimates non-uniform and non-linear PSFs for each by using inertial measurements as proposed in [6]. Further improving the mesh-division idea with introduction of adaptive mesh-size concept. The mesh-size of grids are changing according to blur magnitude variance of the motion vectors for each frame. Two adaptive-mesh size algorithms are proposed; one determines the mesh-size for best quality other for balanced quality and computation time. The experimental results shows that our method increases the quality of deblurring with decreasing computational time in synthetic datasets.

1.1 Literature Review

Image blur is generally modeled as a convolution problem. The solution process includes estimation of blur kernel and latent image which is an ill-posed problem. Various blind and non-blind deconvolution algorithms are developed in order to recover sharp image. In addition to conventional deblurring algorithms, new brand of algorithms is developed with increasing usage of deep-learning and neural networks.

Blind deconvolution algorithms try to estimate both blur kernel and latent image simultaneously. Most algorithms formulate the problem as a minimization problem and employ regularizers or priors [7]–[13]. L_0 -regularized priors [12], [14], Gaussian image prior [9], dark channel prior [8], combination of bright channel prior and dark channel prior named Extreme Channel Prior (ECP) [15] are some of the modelling methods for natural image statistics. These methods generally tolerate small amount of noise present in the image and can handle outliers. However, their performance

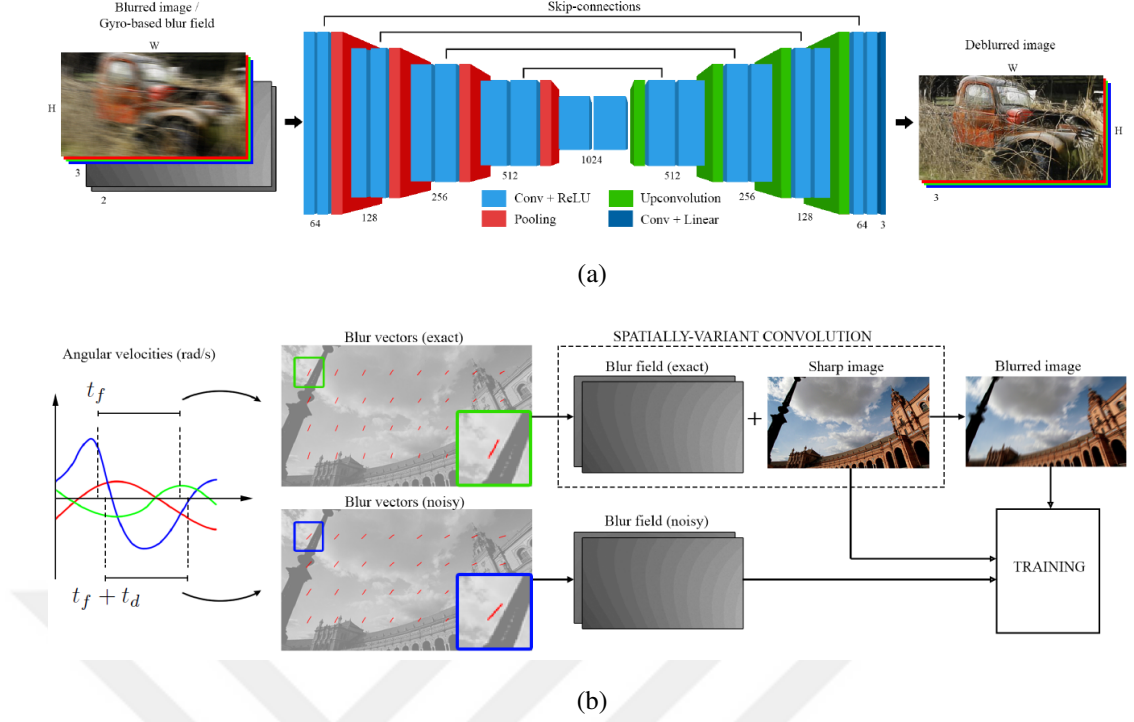


Figure 1.1: The DeepGyro CNN architecture. Rotational measurements are used for generating blur field which is equal to PSF [18].

significantly decreases in blurred images with saturated pixels and moderate noise. In addition, computation cost problem and unreliability under noise is still a drawback.

Deep learning and neural network based image deblurring methods are also beginning to arise and make some important contributions to this area lately [1]–[3], [16]–[18]. In study [1], a generative adversarial network (GAN) is proposed with introducing new feature pyramid network to deblurring, and further improving their previous study [19]. Multi-scale and scale-recurrent architectures are used with a multi-level convolutional neural network (CNN) model in [2] in order to lower the runtime and improve quality problems of deep-learning based methods. An inertial measurement based CNN network is proposed in [18]. It is the first study that successfully combines inertial measurements and deep-networks in deblurring area. The pipeline of the networks is shown in Fig.1.1 Despite the latest developments in this area, requirement of powerful GPU systems, dataset specific performance and large training times are still the undergoing problems.

In the presence of PSF, non-blind deconvolution methods are applied to blurred image to recover restored image. These methods also employ natural image statistics and formulate the solution as a minimization problem. The difference and success of these methods over blind algorithms comes from the knowledge of the PSF which must be estimated in blind methods along with the latent image. Several methods have been proposed to investigate the non-blind deblurring problem with natural image statistics [20]–[25]. Gaussian and hyper-Laplacian priors used in [20]. Employing Gaussian priors results in a linear system which can be easily solved. However, utilization of Gaussian priors introduce ringing-artifacts. In order to remove ringing-artifacts, [26] proposed a multi-scale approach with residual deconvolution in each scale. Usage of hyper-Laplacian priors require to deal with non-convex optimization.[27] proposed a method for solution of this problem with half-quadratic minimization (HQM). This method also used in $TV - L_1$ (total variation) in the study [22].

Non-blind deconvolution algorithms assumes that the PSF is available as an input. This PSF can be estimated using the data from the IMUs which are available in most of the image capturing devices. Considerable amount of work has been conducted on inertial-aided image deblurring algorithms with combining IMU generated PSF and non-blind deconvolution [4]–[6], [28], [29]. These studies use the gyroscope and/or accelerometer readings obtained during the exposure time of the camera in order to estimate blur kernel. This gives them a computational advantage over the blind algorithms and make them proper for real-time applications because of prior knowledge of PSF. However, most of the inertial-aided algorithms [4], [5], [28], [29] assume spatially-invariant blur in obtained images. This assumption does not hold under rotational motion along optical axis and decreases the quality of the deblurring in such cases.

To deal with spatially-variant blur, mesh-grid idea is developed and implemented in [6]. The image is divided into constant meshes and partially spatial-variant PSFs are estimated for each grid as it can be seen in Fig. 1.2. The PSFs are estimated using inertial measurements and assumed to be uniform inside each mesh. Similar grid-based approach is conducted in [30] but blur kernel is estimated blindly.

Considering under performance of inertial-based non-blind deblurring algorithms un-



Figure 1.2: (Left) A real-world image with spatially-variant motion blur. (Center) Deblurred image with constant mesh-sized deblurring method. (Right) Blur kernels calculated from IMU motion estimates [6].

der spatial-variance, we mainly focused on PSF estimation process and effect of spatial-variance on deblurring. Our work is based on the mesh-grid method proposed in [6]. The mesh-grid implementation simply divides blurred image into grids and perform PSF estimation and non-blind deconvolution for each. PSFs are estimated for center pixel of each grid using the inertial measurements obtained during the exposure time. In our method, mesh-size is not predetermined, it changes adaptively according to variance of the blur magnitude for each image. Blur magnitude variance is an indicator for spatial-variance present in blurred image. We realized that if this variance is small, the spatial-variance of PSF is small and mesh-size can be increased which does not affect quality but considerably decreases the computation time. Usage of adaptive mesh-size allows us to decrease PSF estimation error while keeping computational time at minimal level.

1.2 Contributions

The present study focuses on the motion blur problem for mobile platforms and investigates the current problems of computational cost and quality of existing methods. For this purpose, we evaluated the current motion deblurring methods and observed that non-blind motion deblurring algorithms with the usage of inertial measurements are the most effective solution among the existing methods. However, we realized their limitations in spatially-varying motion blur cases which is common in most of the applications. In addition, we focused on obtaining acceptable deblurring quality while keeping the computation times limited. To best of our knowledge, our adaptive-mesh size algorithm is the first one that changes the grid size according

to spatial-variance indicator and perform non-blind deconvolution. Our approach improves the performance of deblurring algorithm by decreasing computational time and increasing the quality according to our experimental results.

Our first contribution is developing an adaptive mesh-grid architecture for existing non-blind deconvolution methods in order to deal with the spatially-varying motion blur. This system allows us to estimate different PSFs for different regions of the blurred image and increases the quality of restored image. Realizing that each frame has different blur characteristic and magnitude, an adaptive mesh-size determination method introduced. This method finds the mesh-size which gives the deblurring results with best quality according to magnitude of the motion vectors obtained from inertial measurements. Our high quality mesh-size algorithm do not consider the time constraints and has high runtime.

Secondly, a variance parameter is introduced to represent the level of the blur non-uniformity in the image. This is achieved by taking variance of the motion vectors obtained from the inertial measurements. Variance parameter is used for changing mesh-size. High variance indicates PSF differences between regions are not negligible and more accurate estimation needed, so smaller mesh-sizes should be selected to decrease PSF estimation error. In case of small variance, the mesh-size can take higher values. Changing mesh-size according to variance allows us to lowering computation time without losing the gains in quality.

Finally, observing the high computation time of the proposed high quality mesh-size algorithm, we proposed another adaptive mesh-size algorithm which looks for speed-quality balanced mesh-size considering both runtime and quality. In order to determine this optimal point, we conducted several tests on synthetic datasets. Using hand-labeled results, an exponential equation is obtained which gives speed-quality balanced mesh-size according to the variance of the input blurred frame. This method further improved with a tradeoff parameter that can be adjusted for obtaining desired computation time and quality within the existing limits of the proposed algorithm.

1.3 Outline of the Thesis

The organization of the thesis is as follows: a brief introduction and literature review is presented in Chapter 1. After this introduction part, mathematical background, convolution model for motion blur and existing solution methods are provided. Also, PSF generation process from inertial measurements for spatially-varying motion blur is clarified and detailed in Chapter 2. In Chapter 3, under performance of current non-blind algorithms is exploited, and proposed architecture of adaptive mesh-size is introduced. Two adaptive mesh-size algorithms and their development processes along with a variance parameter are given. Chapter 4 provides an evaluation over current existing non-blind algorithms. Selected algorithm is used to test proposed method of adaptive mesh-size structure. Quantitative comparison and evaluation of the proposed method are presented in this chapter. The results and findings of our studies along with limitations and possible future work are summarized in Chapter 5.



CHAPTER 2

INERTIAL-AIDED NON-UNIFORM MOTION DEBLURRING

In this chapter, we provide the mathematical model and background information to the problem of motion blur along with the existing solution approaches. This is followed by motion trajectory estimation and final PSF generation process from inertial measurements. The aim of this section is to obtain a representation for blur kernels from inertial measurements that can be used in non-uniform motion blur.

2.1 Mathematical Model of Motion Blur

2.1.1 Motion Blur Model as a Convolution

In order to solve the motion blur problem, image generation process should be modelled. In the ideal case, every point in the scene should project onto single point in captured image and do not contribute the formation of other points. In the presence of relative motion between image capturing device and scene, this assumption is violated. Scene points may contribute the formation of more than one pixel because of this motion during the exposure period.

A mathematical model is provided [31] in below equation which gives the relation between ideal sharp image $f(x, y)$ and the observed blurred image $g(x, y)$.

$$g(x, y) = n \left(\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h(y, x; r, c) \cdot f(r, c) dr dc; y, x \right) \quad (2.1)$$

The Point Spread Function $h(y, x)$ defines the spreading of a point induced by motion. Also, a term n exists in image formation model for image noise, quantization error

and non-linear camera response curve.

In above model, every pixel has its own PSF which is not practical and feasible. For this reason, single PSF is estimated for all pixels which is spatially invariant over the image. In order to make this assumption, scene points must be same distance from the capturing device. With this assumption PSF becomes:

$$\forall y, x, r, c : h(y, x; r, c) = h(y - r, x - c) \quad (2.2)$$

An additive term with zero-mean included for representing the non-idealities generated during process. This assumption is generally valid and model simplifies as:

$$g(y, x) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h(y - r, x - c) \cdot f(r, c) dr dc + n(y, x) \quad (2.3)$$

To obtain discrete model, we convert the above continuous time signal as:

$$\begin{aligned} g(y, x) &= \sum_{-\infty}^{+\infty} \sum_{-\infty}^{+\infty} h(y - r, x - c) \cdot f(r, c) + n(y, x) \\ &= h(y, x) \otimes f(y, x) + n(y, x) \end{aligned} \quad (2.4)$$

The above equation shows the mathematical model of blurred image as convolution of point spread function h and ideal image f with additive noise term n . In order to obtain the latent sharp image, this convolution problem should be solved using an image deconvolution process. Image deconvolution can be categorized to the blind and non-blind cases according to prior information about PSF.

2.1.2 Point Spread Function

Kernel or convolution kernel is a small matrix which is generally used in image processing area for blurring, sharpening, edge detection etc. The convolution kernel for blurring is referred as PSF or blur kernel. The elements of the PSF convolution kernel should be non-negative and must satisfy the energy conservation constraint. This

means that the energy of the image should not change after convolution process. The following equation must be satisfied by PSF :

$$\sum_r \sum_c h(r, c) = 1 \quad (2.5)$$

After modelling the blurred image as a convolution problem, recovering process of latent image is called deconvolution.

2.1.3 Inverse Filtering

The simplest deconvolution method is the inverse filtering. Method uses the convolution theorem. Convolution in spatial domain corresponds to multiplication in frequency domain:

$$g = h \otimes f \quad \Leftrightarrow \quad G = H \cdot F \quad (2.6)$$

where F, G and H are the Fourier transforms of the original image, blurred image and convolution kernel respectively. The restored image $\tilde{f}$ can be found by inverse discrete Fourier transform as:

$$\tilde{f} = \mathcal{F}^{-1} \left(\frac{G}{H} \right) \quad (2.7)$$

This solution seems to be effective and simple. However, when additive noise is present in the image the quality of the filtering decreases dramatically. After including the additive noise term n to the equation:

$$g = h \otimes f + n \quad \Leftrightarrow \quad G = H \cdot F + N \quad (2.8)$$

Inverse discrete Fourier transform gives us the restored image $\tilde{f}$ as:

$$\begin{aligned} \tilde{f} &= \mathcal{F}^{-1} \left(\frac{H \cdot F + N}{H} \right) \\ &= \mathcal{F}^{-1} \left(F + \frac{N}{H} \right) \end{aligned} \quad (2.9)$$

In the above equation, the division of N by small H values amplifies the noise. Therefore, the deblurring quality decreases as the noise increase.

2.1.4 Regularized Approaches

To solve the deconvolution problem, more complicated methods are developed [11], [20], [25], [27], [32]. These approaches generally try to minimize an energy function consisting of a data term E_{data} and a regularization term E_{reg} . The data term represents the difference between observed blurred image and convolved image with blur kernel as:

$$E_{data} = \Phi(\mathbf{h} \otimes \mathbf{f} - \mathbf{g}) \quad (2.10)$$

In above equation, Φ represent the distance function. Most commonly used distance function is L_2 norm. The second term E_{reg} is given as $\Psi(\mathbf{f})$. The restored image can be estimated by minimizing the energy function below:

$$\min_{\mathbf{f}} \|\mathbf{h} \otimes \mathbf{f} - \mathbf{g}\|^2 + \lambda \Psi(\mathbf{f}) \quad (2.11)$$

where λ is relative weight between terms. Data term enforces similarity to the captured image g , while the regularization term adds penalty based on defined constraints.

Table 2.1: Shows the comparison of some non-blind deconvolution methods according to likelihood and prior [33]

	Likelihood	Prior
Wiener [34]	Gaussian	Gaussian
L-S [32]	Gaussian	Gaussian
Chan [24]	Gaussian	Laplacian
Wang [22]	Gaussian	Laplacian
Shan [14]	Gaussian	Concatenating
Yuan [26]	Gaussian	Non-local(bilateral)
Krishnan [27]	Gaussian	Hyper-Laplacian
Yang [22]	Laplacian	Laplacian
Xu [12]	Laplacian	Laplacian

Main bulk of methods use square regularization constraints. Some examples of these regularization forms are $\Psi(\mathbf{f}) = \|\nabla \mathbf{f}\|^2$ and $\Psi(\mathbf{f}) = \|\mathbf{f}\|^2$. These operators called

Gaussian and Tikhonov regularizers which enforce smoothness on image gradients. We can take Tikhonov regularization method and use it in our blur convolution model as:

$$\min_{\mathbf{f}} \|\mathbf{h} \otimes \mathbf{f} - \mathbf{b}\|^2 + \lambda \|\nabla \mathbf{f}\|^2 \quad (2.12)$$

where λ represent weight which is generally a small value. The overall energy minimization can be solved with Tikhonov method by taking the first order derivative with respect to f and setting it to zero. After this operation equation becomes:

$$\begin{aligned} E &= \|Hv(\mathbf{f}) - v(\mathbf{g})\|^2 + \lambda \|v(\mathbf{f})\|^2 \\ &= v(\mathbf{1})^T H^T H v(\mathbf{f}) - 2v(\mathbf{g})^T H v(\mathbf{f}) + v(\mathbf{g})^T v(\mathbf{f}) + \lambda v(\mathbf{f})^T v(\mathbf{f}) \end{aligned} \quad (2.13)$$

where v is the operator which transforms the image into vector form, and H is a sparse convolution matrix generated from h . The derivative is denoted as:

$$\frac{\partial E}{\partial v(\mathbf{f})} = 2H^T H v(\mathbf{f}) - 2H^T v(\mathbf{g}) + 2\lambda v(\mathbf{f}) \quad (2.14)$$

Optimal solution f^* can be obtained by setting the above equation to zero,

$$v(\mathbf{f}^*) = \frac{H^T}{H^T H + \lambda \Lambda} v(\mathbf{g}) \quad (2.15)$$

where Λ represents an identity matrix of size $H^T H$.

In the previous steps, we have obtained an expression for restored image by Tikhonov method. In addition to this method, various new regularizers are developed for dealing with the noise and ringing-artifacts caused by blur kernel estimation errors. The main idea in these approaches is reducing the amplification and penalizing the outliers. Otherwise, effect of noise and estimation to restored image increases. Some of the regularizer penalty difference can be seen in Fig.2.1.

In [24] a total variation regularizer proposed by setting prior to:

$$\Psi(\mathbf{1}) = \|\nabla \mathbf{f}\|^1 \quad (2.16)$$

In above equation, $\nabla \mathbf{f} = (\partial_x \mathbf{f}, \partial_y \mathbf{f})$ represents the first-order derivative. and L_1 -norm

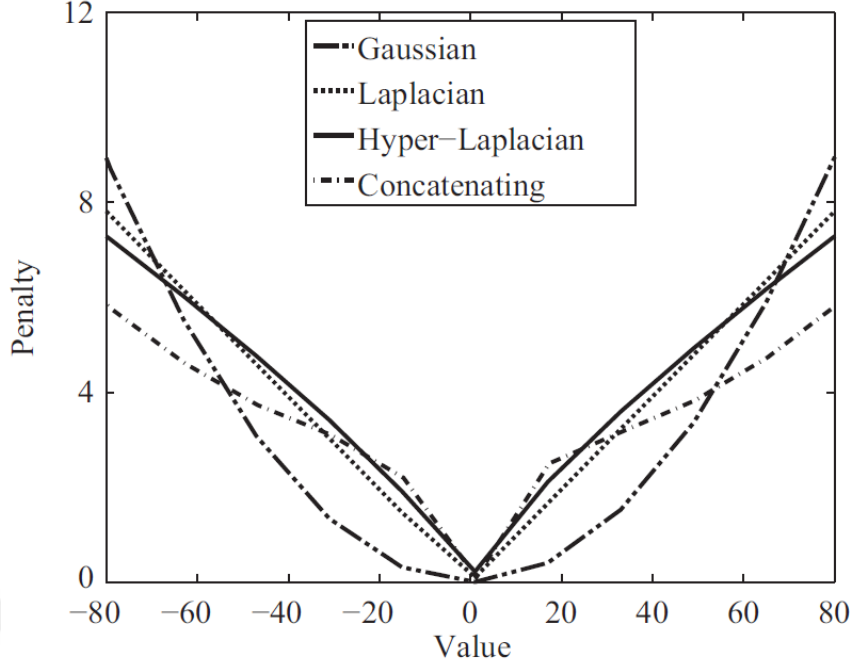


Figure 2.1: Illustration of different regularization terms. Different prior functions penalize values differently. The Gaussian prior increases energy most quickly for large absolute values

is used for image gradients. In another method proposed in [14], two piece-wise continuous convex functions are concatenated for regularization.

$$\Psi(\mathbf{f}_i) = \begin{cases} a |\nabla \mathbf{f}_i| & |\nabla \mathbf{f}_i| \leq \xi \\ b (\nabla \mathbf{f}_i)^2 + c & |\nabla \mathbf{f}_i| > \xi \end{cases} \quad (2.17)$$

where i is pixel index, $\nabla \mathbf{f}_i$ stands for partial derivative in the direction of x or y . ξ represents the position where two functions are concatenated. a , b and c are curve fitting parameters and selection of these can be found in [14].

To make image less smooth and more sharp, [27] proposed a hyper-Laplacian prior for $0 < \alpha < 1$ which is represented as:

$$\Psi(\mathbf{f}) = \|\nabla \mathbf{f}\|^\alpha \quad (2.18)$$

Changing the energy minimization term by adding this hyper-Laplacian prior to over-

all equation becomes:

$$\min_{\mathbf{f}} \|\mathbf{h} \otimes \mathbf{f} - \mathbf{g}\|^2 + \lambda \|\nabla \mathbf{I}\|^\alpha \quad (2.19)$$

Above equation can be solved by variable splitting method. Variable splitting is simply separating the convolution term from other terms and introducing auxiliary variables, so that they can be estimated individually by Fourier transform. This separation increases the speed and reliability. The mathematical derivation and quadratic solution process can be found in detail in [27], [33].

2.2 Inertial-aided Spatially Varying Motion Blur PSF Computation

2.2.1 Camera Model and Motion Projection

We present the pinhole camera model and derive apparent image plane motion of an object plane point due to the 6D camera motion in space. In Fig.2.2, the image plane is located on a plane parallel to the x-y plane with a distance from f from the origin, optical axis is along the z axis.

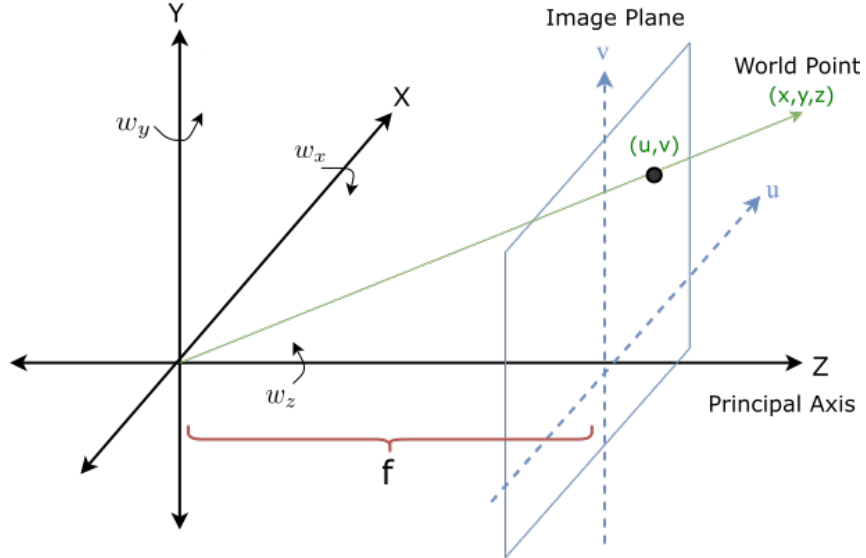


Figure 2.2: Coordinate axis definition for camera projection model. Z axis is principal axis, f is the focal length of the camera. World points are projected into image plane.

Rotation around the optical axis z is defined as roll, rotations of x and y are defined as pitch and yaw respectively. Using the pinhole camera model world coordinates (X, Y, Z) are projected into image plane coordinates as (x, y) ,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \frac{X}{Z} \\ f \frac{Y}{Z} \end{bmatrix}, \quad \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \frac{Zx}{f} \\ \frac{Zy}{f} \end{bmatrix} \quad (2.20)$$

where f represents the focal length of the camera. Taking derivatives of the equation with respect to time gives us the velocities of image points.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} x \\ y \end{bmatrix} = f \begin{bmatrix} \frac{\dot{X}Z - X\dot{Z}}{Z^2} \\ \frac{\dot{Y}Z - Y\dot{Z}}{Z^2} \end{bmatrix} = \begin{bmatrix} \frac{f}{Z} & 0 & -\frac{fX}{Z^2} \\ 0 & \frac{f}{Z} & -\frac{fY}{Z^2} \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (2.21)$$

Actual world point velocities are zero. However, the camera motion generates these velocities. So, by moving the world point instead of camera, same effect appears on the image plane as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & -\omega_Z & \omega_Y \\ \omega_Z & 0 & -\omega_X \\ -\omega_Y & \omega_X & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (2.22)$$

Substituting this into (2.21) with conversion in (2.20) gives us the following equation:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \omega_Z y - \omega_Y f - \frac{\omega_Y}{f} x^2 + \frac{\omega_X}{f} xy \\ \omega_Z x - \omega_X f - \frac{\omega_Y}{f} xy + \frac{\omega_X}{f} y^2 \end{bmatrix} \quad (2.23)$$

The above equation gives us the image plane velocities considering the rotational movements. If we add the translational velocities, final 6D image plane velocity equation becomes :

$$[H] \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \omega_Z Y - \omega_Y f - \frac{\omega_Y}{f} X^2 + \frac{\omega_X}{f} XY - \frac{f}{Z} v_X + \frac{u}{Z} v_Z \\ \omega_Z X - \omega_X f - \frac{\omega_Y}{f} XY + \frac{\omega_X}{f} Y^2 - \frac{f}{Z} v_Y + \frac{v}{Z} v_Z \end{bmatrix} \quad (2.24)$$

Using the above equation with knowledge of rotational and translational motion, we can obtain the image plane velocity vectors. This motion knowledge will lead us to compute PSF. In above motion equation, the effect of the translational movements depend on the distance z which is the distance between camera and object. When this distance between camera and object is large, translational movements can be neglected. However, rotational movements are independent of the distance. For this reason, we will consider only rotational movements and neglect the translational motion assuming that the distance between capturing device and object scene is considerably large. In our study, we make our calculation according to equation (2.23) where only rotational movements are taken into consideration.

In below Fig.2.3, rotational movement along the x axis (pitch), y axis (yaw) and z axis (roll) are simulated and the corresponding blurred images are shown next to motion velocity fields for observing the effect of the different motions along the axis. The motions along x and y axis are uniform and equal through whole image. This results in uniform motion blur in the simulated images also. In that case, estimated blur kernels are equal and do not differ from region to region which most of the non-blind algorithms assume. This situation changes when rotational motion exists along the optical axis z . This generates non-uniform motion which changes from the region to region and has a devastating effect on captured images. The main cause of the non-uniformity and PSF estimation error in the non-blind methods is this roll motion. The motion is smaller and almost zero at the center region. The magnitude of the field increases as moving away from the center. The effect can be easily seen in the motion fields and the blurred image at the bottom in Fig.2.3 .

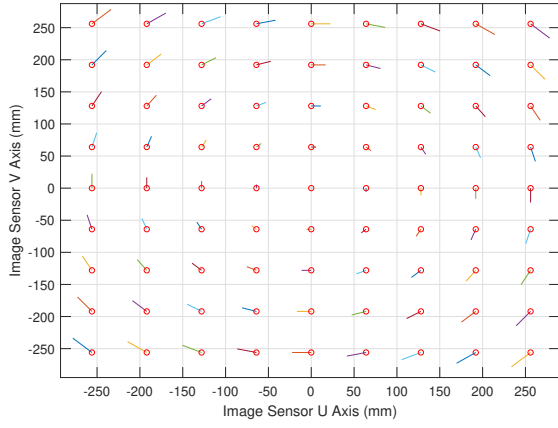
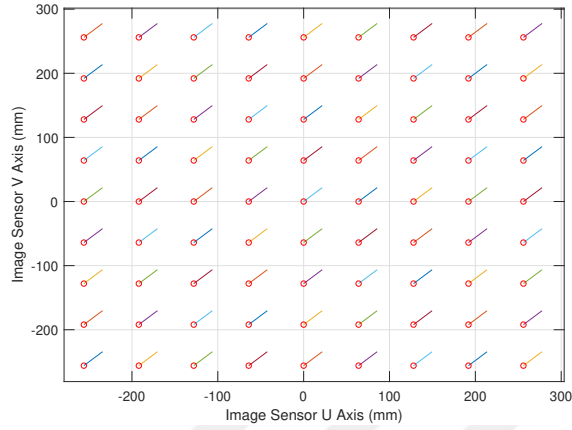
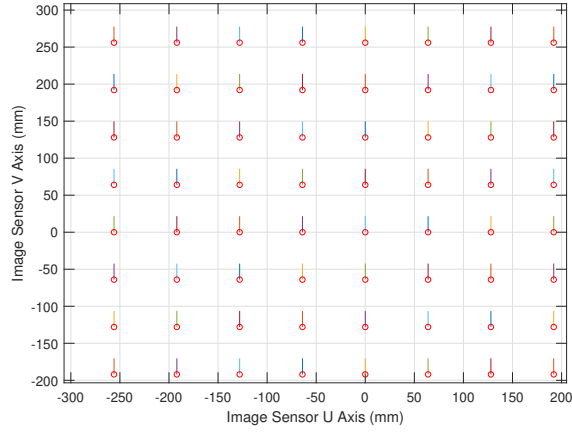


Figure 2.3: Three pairs of images are shown which represent the motion fields and corresponding blurred images. PSFs generated from these motions and used to simulate blurred images. The blurred images given next to velocity field graphs. Rotational movements along the x axis (pitch), combination of the x axis and the y axis (yaw), the z axis (roll) are simulated top, middle and the bottom row respectively. Motions along the x and y axis are uniform while the z axis is non-uniform. The effect of motions can be observed from the blurred images.

2.2.2 Inertial Measurements

In the equation (2.24), we have determined the 6D camera motion from rotational and translational movements and neglect the translational component assuming large distance.

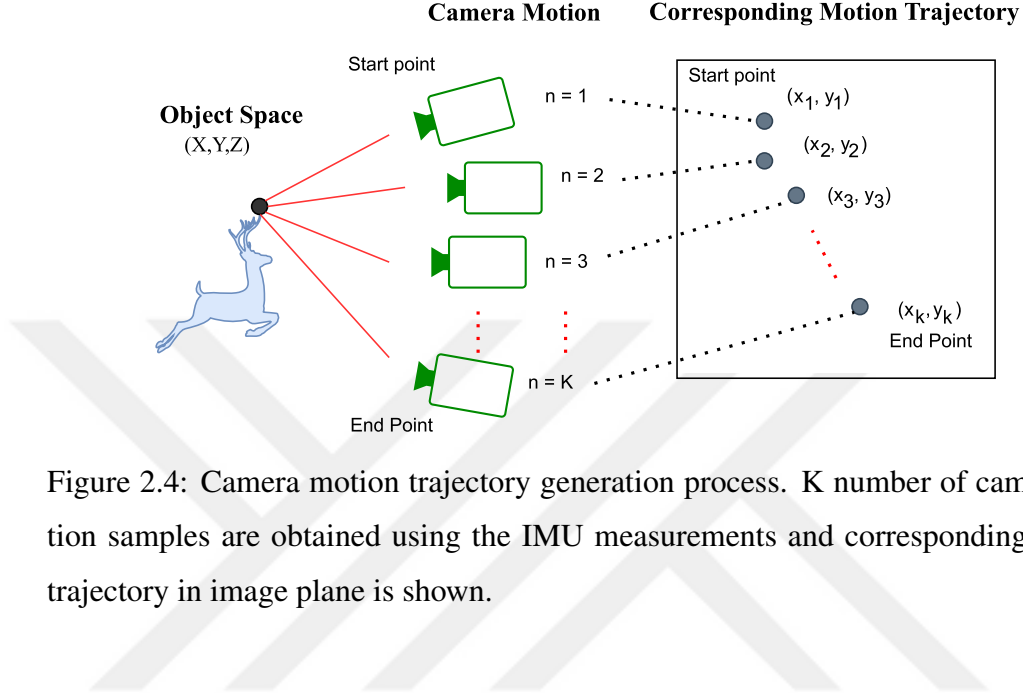


Figure 2.4: Camera motion trajectory generation process. K number of camera motion samples are obtained using the IMU measurements and corresponding motion trajectory in image plane is shown.

The remaining rotational movement information for estimating the motion on the image plane is obtained from the inertial measurements. In Fig.2.4, motion of the camera within exposure time and its corresponding effect on the image plane is illustrated. The (X, Y, Z) world point projected to different points in the image plane. Assuming in each camera location an inertial measurement is available, the corresponding vector in the image plane is obtained by using equation (2.23),

$$(x_n, y_n) = \sum_1^n (u_n, v_n) * \Delta t \quad \text{for } n = 1 \text{ to } k \quad (2.25)$$

where k equals to the inertial measurement sample number, Δt is the time between samples and (u_n, v_n) represents the speed at the n th measurement. Using this equation, the motion in the image plane is obtained for PSF estimation by using inertial measurements.

2.2.3 PSF Estimation

We have projected the world point (X, Y, Z) onto K different positions (x_k, y_k) assuming K different poses are sampled by inertial measurements. This motion trajectory is generated directly from the inertial measurements and relies on the accuracy of them. Ideally, this projected motion in image plane is equal to the PSF of the camera motion. The PSF for all image is represented as:

$$h(m, n) = \frac{1}{K} \sum_{k=1}^K \delta(m - x_k, n - y_k) \quad (2.26)$$

However, the inertial measurements are noisy and this accuracy assumption is not valid. Noisy measurements causes wrong PSF estimation which led to poor deblurring results. In the below Fig. 2.5, inertial measurements are taken from standing still smart-phone IMU device. In ideal case the measurements should be approximately zero. But, measurements generally have some noise which is almost present in all devices especially in low-cost IMUs.

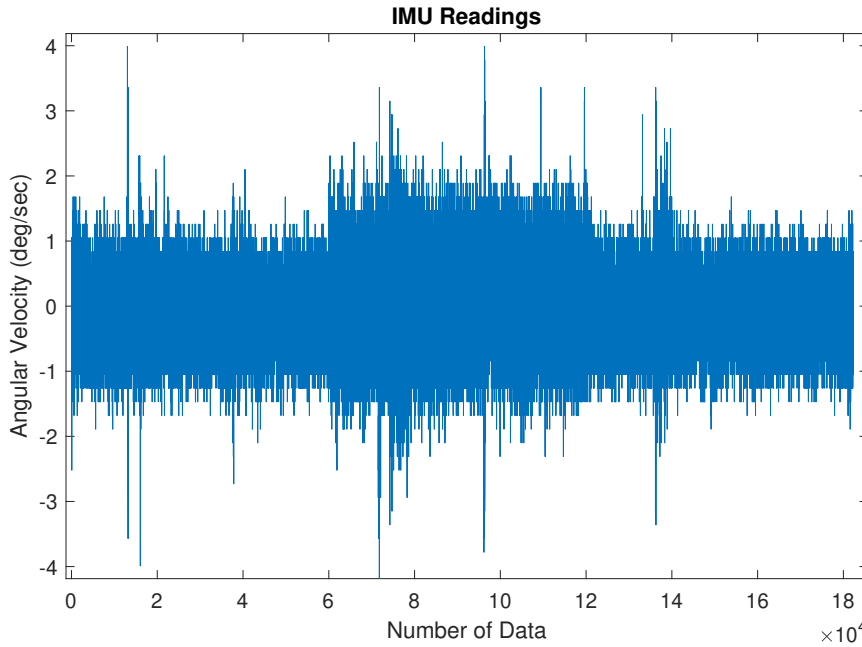


Figure 2.5: Inertial-measurements taken from smart-phone internal IMU. Only rotational motions are taken and combined into single data for observation.

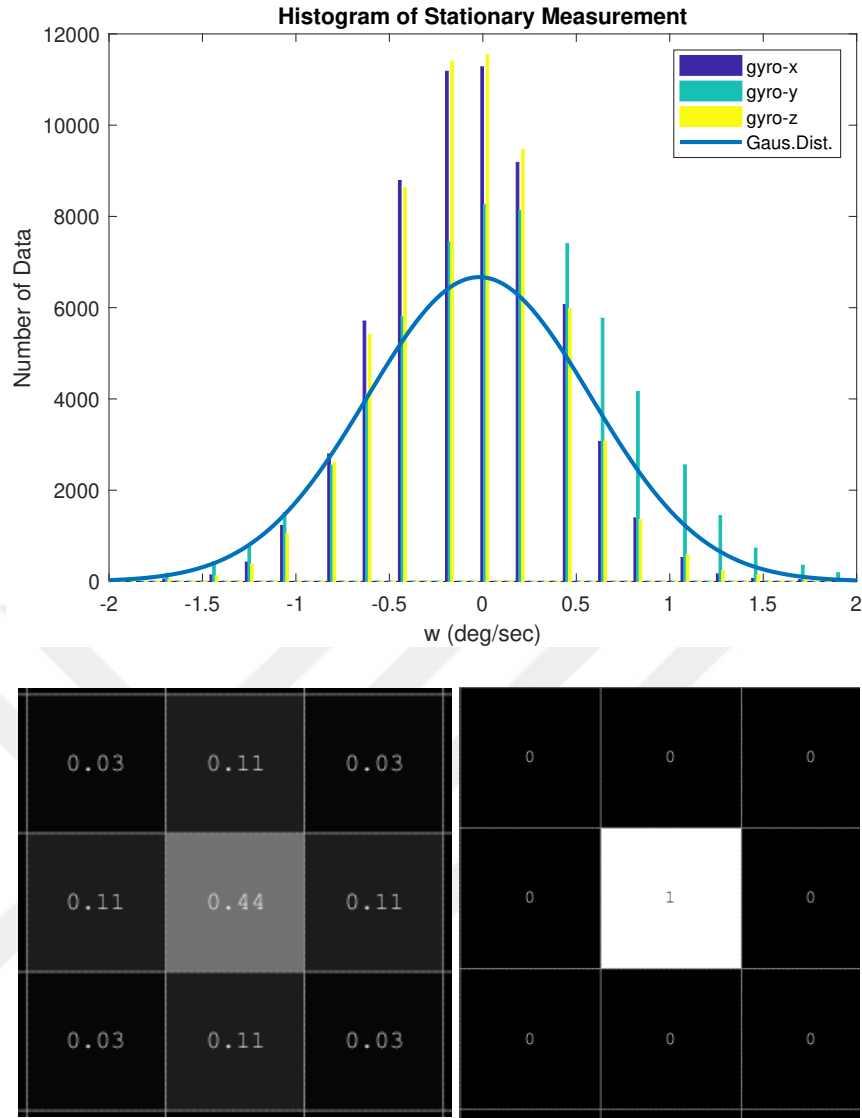


Figure 2.6: Shows the histogram of the given data in Fig.2.5. Gaussian distribution fitted to this data which is collected from a stationary IMU device. PSF generated from Gaussian distribution and ideal one are represented in bottom images respectively.

Considering noisy measurements, we assume that a point on the image plane has a Gaussian distribution as proposed in [35]. This Gaussian distribution can be determined using the inertial measurements taken from the IMU device when it is standing still. In the above Fig.2.6, a Gaussian distribution is fitted to x,y,z axis angular velocity data which are taken from the standing still position. The combined data shown

in Fig.2.5. The generated single point distributions can be seen in the bottom images of Fig.2.6. The left image shows the single point PSF with Gaussian distribution and the right one shows the ideal case.

Changing the equation in (2.26) with implementing the Gaussian distribution idea, the resulted motion in the image plane can be represented as the sum of Gaussian distributions as:

$$h(m, n) = \frac{1}{K_G} \sum_{k=1}^K G(m - x_k, n - y_k,) \quad (2.27)$$

where G represents a two-dimensional Gaussian distribution determined with above method and K_G is the normalization constant for energy conservation constraint.

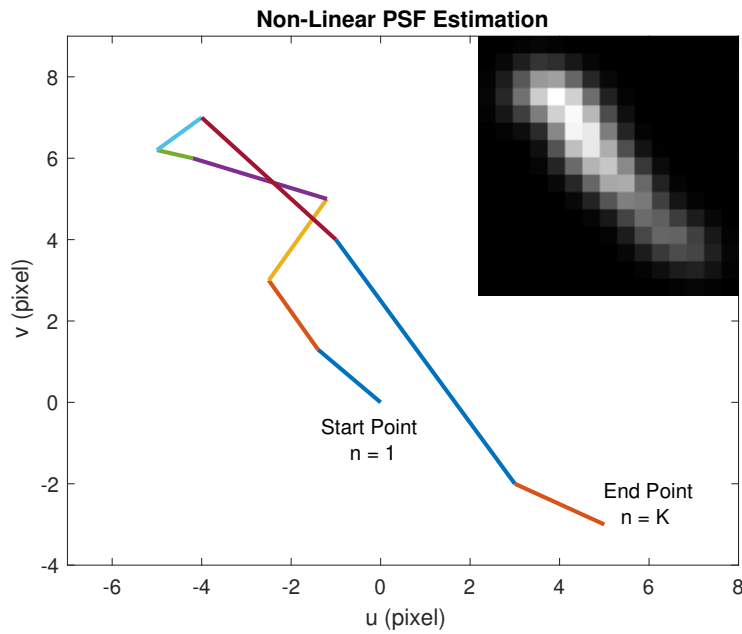


Figure 2.7: PSF estimation from inertial measurements. Corresponding PSF generated from motion trajectory between start point end point using the (2.23),(2.27).The PSF shown in the right top corner.

In above Fig.2.7, the motion during exposure period is represented and using the (2.27), the PSF is generated for deconvolution process from the inertial measurements.

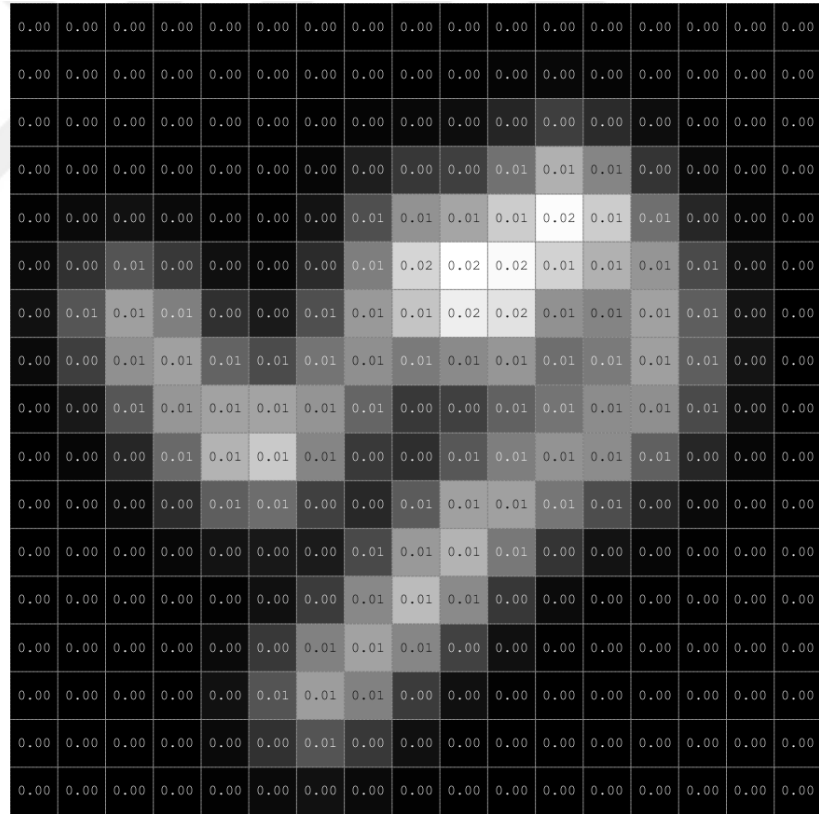
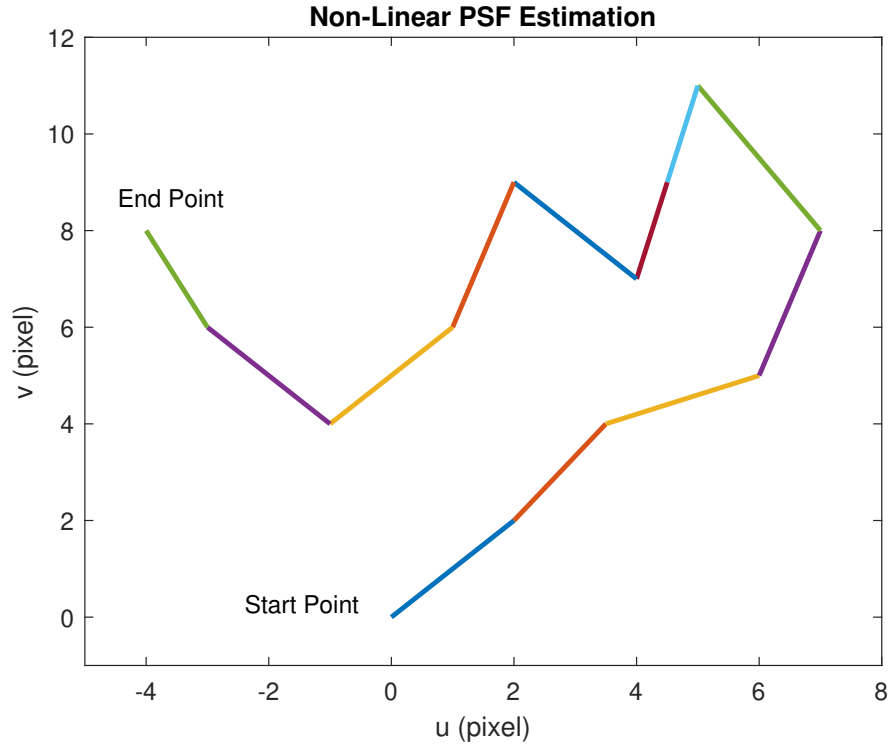


Figure 2.8: Shows an estimation of non-linear PSF for given motion vectors in top graph. The motion consisting of 12 samples of IMU readings. Exposure period selected as 0.02 sec.

The Fig.2.8, an example of non-linear PSF estimation for given motion vectors in image plane is simulated. Each color in the top figure represents a motion vector obtained from inertial measurements. if the motion vectors concentrated in one area, the density of PSF increases for that region, and become more brighter in the resulted blur kernel function. . The process of estimating a non-uniform PSF is explained in this section. Single PSF is estimated and simulated in the previous samples from inertial measurements, in mesh-grid structure there will be multiple PSFs representing each grid which is explained in the next section.



CHAPTER 3

PROPOSED SOLUTION FOR HANDLING NON-UNIFORM MOTION BLUR PROBLEM

3.1 Motivation

Existing non-blind deblurring algorithms generally assume uniform motion blur and perform deconvolution process with a single PSF for all the image pixels. However, the motion blur in real-world generally shows non-linear and non-uniform characteristics. For this reason, the performance of the non-blind methods decreases severely. The reason is that the estimated PSF is not valid for the all the image. Deblurring a spatially-varying blurred image with the same PSF everywhere generally make it worse than blurred image with generating large number of artifacts.

The below Fig.3.1 shows the motion vectors generated from 10 samples of IMU readings and generated PSFs. The observed motion includes heavy rotational movement along the optical axis. Two red boxes are selected which indicate two different regions of the image. Magnified versions of the corresponding vectors and the generated PSFs can be seen at the bottom image. The differences between PSFs is very high which is an index for the level of non-uniformity. If the classical approach of the existing non-blind methods is used for deblurring, there will be a single PSF for all the image pixels. However, this PSF will be valid only around one part of the image and cause large estimation error for the rest of the image.

In this study our goal is finding a solution for this type of motion blur and improve the performance of the non-blind deconvolution methods. These non-blind methods are effective when PSF is accurate. The mesh-grid idea is developed for accurate and fast PSF estimation.

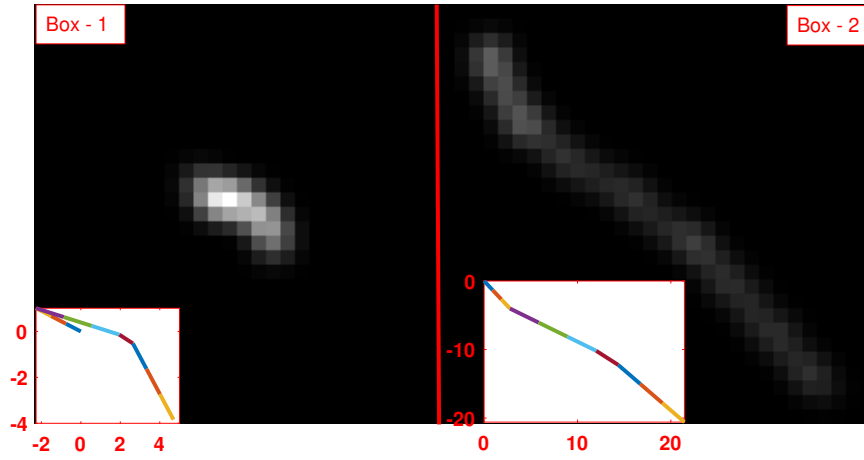
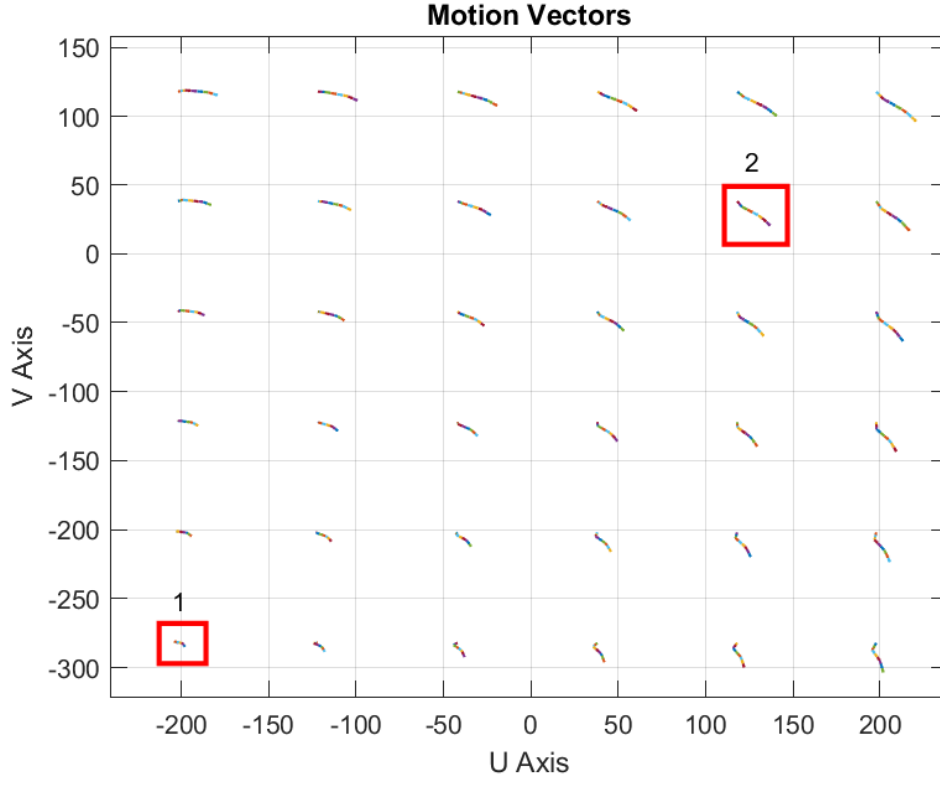


Figure 3.1: Motion vectors generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. Two red boxes are selected at top and bottom corners. Using the PSF generation formula (2.27), two corresponding PSFs generated and shown. The non-uniformity of the blur can be seen in PSFs and motion vectors.

3.2 Mesh-Grid Decomposition for Non-Uniform Blur

To deal with non-uniformity, we developed a mesh-grid based deblurring algorithm. We divide the blurred image into grids as it can be seen in Fig.3.2. Then, We estimate blur kernel for the center of each mesh by using the inertial measurements. Then, deconvolution is performed for each mesh separately and independently. After completing the deconvolution and obtaining deblurred grids, we combine them and get the deblurred whole image.

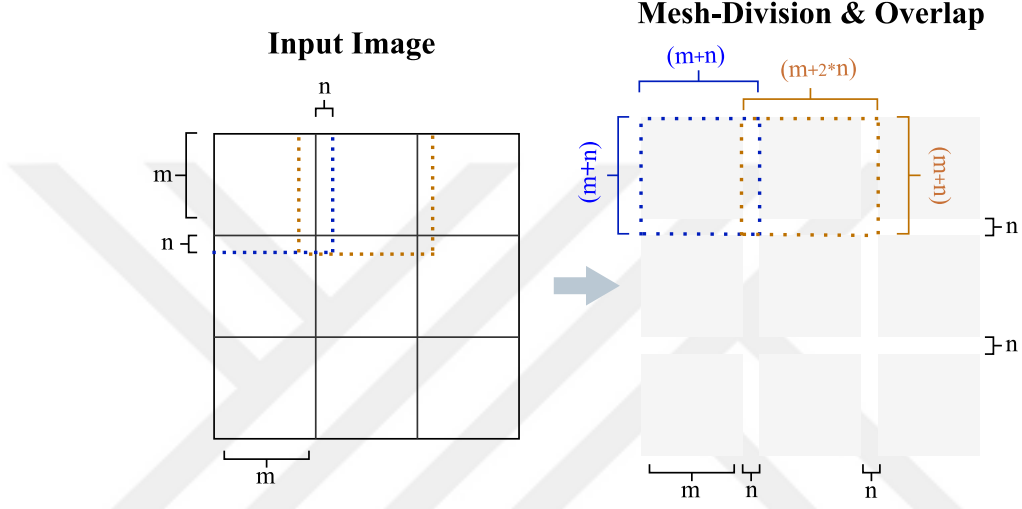
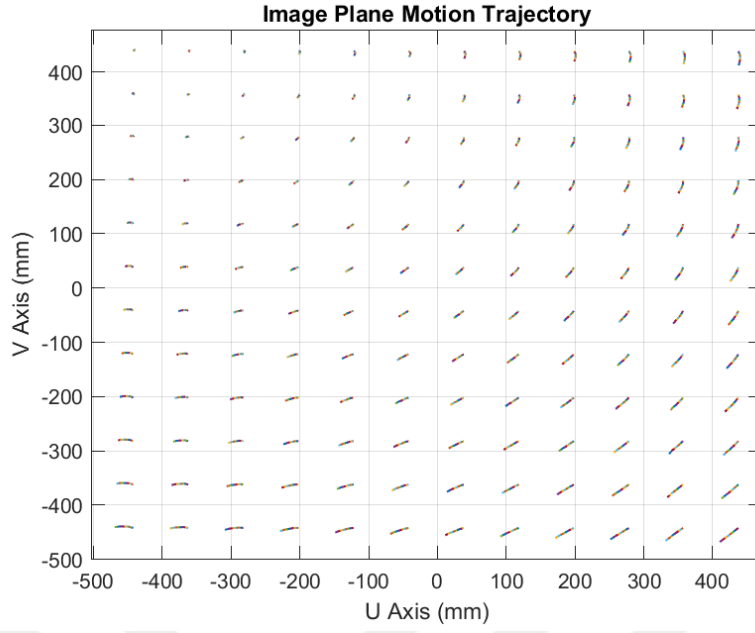


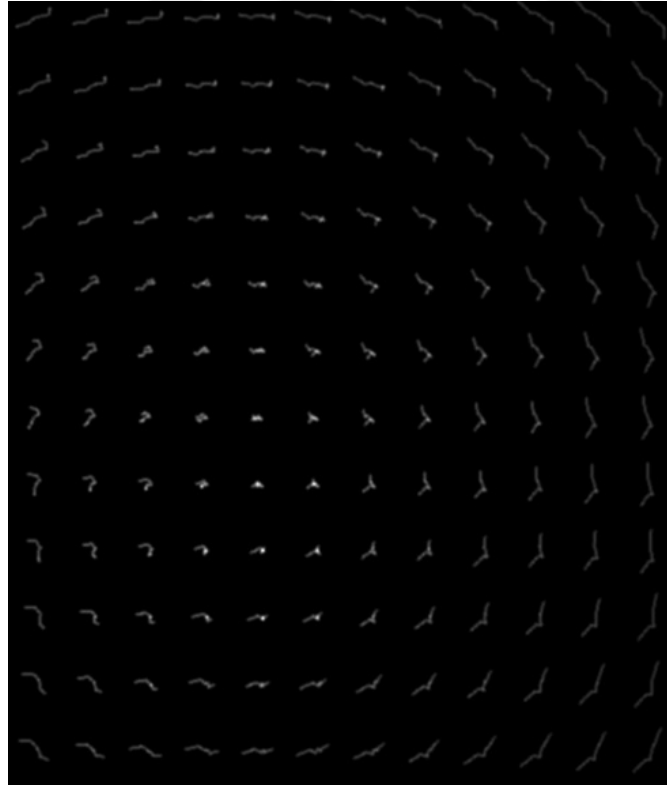
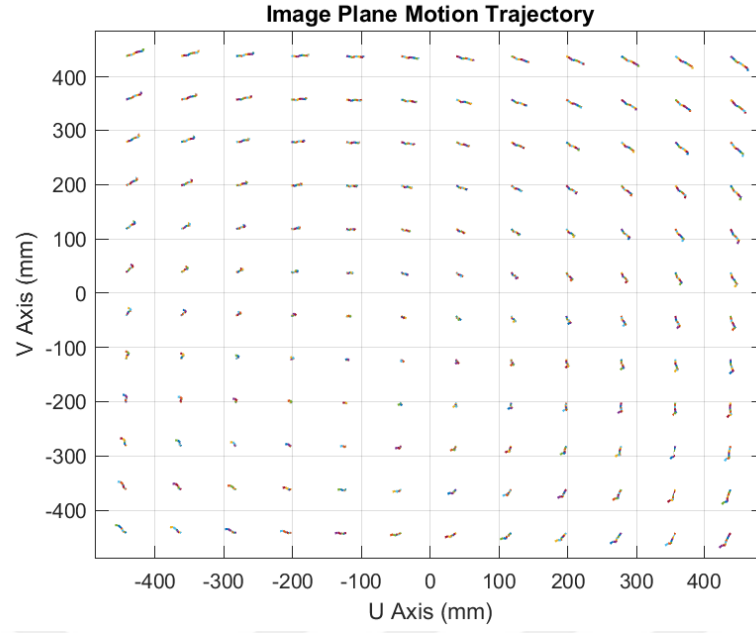
Figure 3.2: Mesh-grid division process of image and introduction of overlap region. $[m, m]$ mesh-sized grids extended with overlap size n . Overlap region is introduced for removing ringing-artifacts. Overlap regions removed after deconvolution.

In above Fig.3.2, the process of division and implementation of overlap region can be seen. The size of the grid-cell is determined adaptively which will be explained in the following section. After selecting the cell size, the image is divided into meshes denoted as $M_{i,j}$ where i, j represents row and column number of each grid. An additional overlap region is introduced in order to avoid ringing artifacts at borders of each cell. These parts are deleted after deblurring process is completed. The relative motion between camera and the object plane is determined according to (2.23). This dense estimation will give us each pixel's motion during the exposure time which is non-linear and non-uniform. Then, we estimate the PSF of each grid according to its center pixel with (2.27). This is sparse representation of previously obtained dense motion. Cell PSF's are represented as $H_{i,j}$ similar to mesh grids.



PSFs

Figure 3.3: Image plane motion trajectory generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. The image size is $[480 \times 480]$, it is divided by $[40 \times 40]$ grids and estimated PSFs for each grid are shown in bottom figure. PSFs differ from region to region.



PSFs

Figure 3.4: Image plane motion trajectory generated from 10 samples of IMU readings shown in top figure for non-uniform blur observation. The image size is [480x480], it is divided by [40x40] grids and estimated PSFs for each grid are shown in bottom figure. PSFs differ from region to region.

In above Fig.3.3 and Fig.3.4, two examples of non-uniform motion with 10 samples of IMU readings in 0.02 sec exposure time is simulated. Motion trajectory graphs and corresponding PSFs are shown for $[40 \times 40]$ pixel mesh-size in the $[480 \times 480]$ pixel image. As it can be seen from the generated PSFs, there are considerable differences between estimated blur kernels.

3.2.1 Ringing Artifacts

Ringing is one of the most common artifact in image deblurring. It presents as wave-like patterns near to sharp-edges, strong discontinuities and generally around image borders. Light and dark waves periodically appear after deblurring process, example in below Fig.3.5.



Figure 3.5: Original, blurred, and deblurred images are given at the top. Colored boxes from original and blurred ones are magnified in second and third rows respectively.

The ringing artifacts usually comes from conversion of discontinuous signal into and out of Fourier domain. Some solutions have been proposed for dealing with this problem in [36],[37],[38] and [39]. The general approach in these algorithms is introducing a cost function considering the ringing artifacts and try to minimize it along with the latent image. These methods can be effective to deal with the ringing artifacts, however they add a considerable amount of computational cost to the deconvolution process.

The deal with the ringing artifacts especially at the edges of the image without increasing the computational cost too much, we used the edge tapering method. Edge tapering is simply blurring the edges of the image with a given PSF function. This is a smoothing process at the border of the image. The edge tapered image is the weighted sum of the input image and its blurred image with the given PSF. The weights are determined by the autocorrelation function of PSF. It makes output equal to the input image at the center area, and equal the blurred version of it around the border which is determined by the PSF size. These figures shows the general structure of mesh-grid idea and PSF generation for each grid.

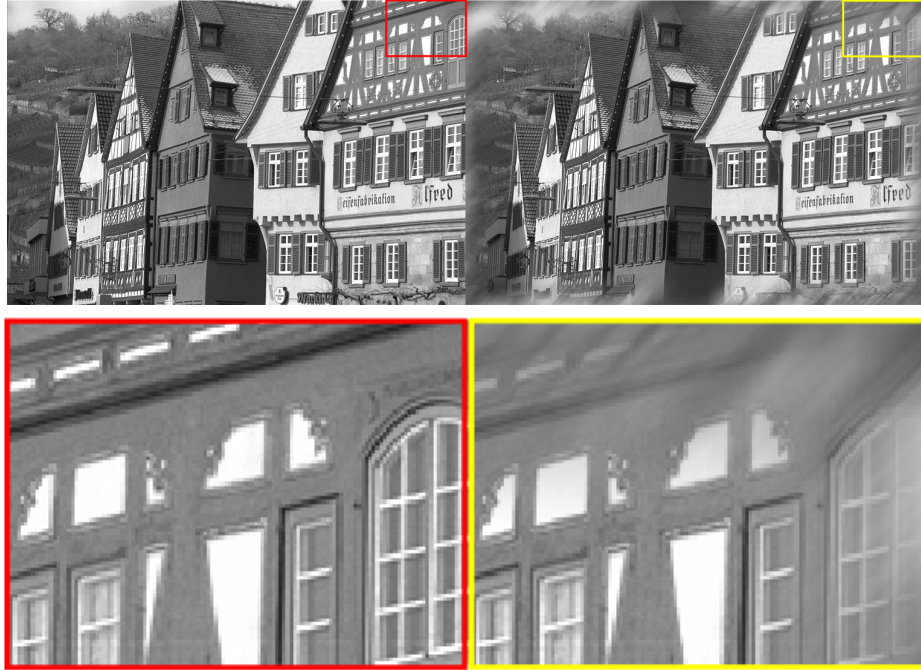


Figure 3.6: Shows the original and edge tapered images at the top figure. The effect of tapering can be seen in magnified parts of both images indexed with boxes.

In Fig.3.6, the edge tapering and its effect on image edges is shown. In that example, we give the input as a sharp image for observing the effect clearly. In our case, input image is blurred image and it will tapered with estimated blur kernel.



Figure 3.7: Shows the deblurred image results with and without edge tapering at the top figure. The selected boxes are magnified and the decrease in the ringing artifacts can be seen in results.

The results of edge tapering can be seen in above Fig.3.7. The ringing artifacts are still existing but their effect is decreased especially in the image borders. Note that image edges are especially important for us since we divide image into grids which ringing appears all of the grid edges.

3.2.2 Determination of Overlap Region

The existing approach of the mesh-grid decomposition is given in previous section in Fig.3.2 with a small difference of overlap region. Existing method [6] does not employ this kind of structure. We introduce overlap region and edge-tapering in order to deal with ringing artifacts and removing the visibility of the mesh-grid structure. We observed that without any overlap region mesh-grid lines appear in the deblurred image as artifact. In addition, each grid acts as a small image with borders, this causes heavy ringing waves everywhere. The mesh-grid artifacts can be seen in below Fig.3.8 as obvious discontinuity in grid borders.



Figure 3.8: Shows the blurred image and deblurred image with mesh-grid size of [60x60] pixels respectively. The mesh-grid artifacts appear on each grid's border.

The effect of both edge tapering and overlap region can be seen in below Fig.3.9. It is observed that not using both methods causes heavy ringing artifacts and worsening the blurred image. Just using edge-tapering reveals the mesh-grid lines and tapering effect at the each grid's borders. Observing these results, we decided to use both overlap region and edge tapering together. The overlap size is changed according to the blur magnitude same as edge tapering. Increasing further to that value does not change the result. So, adding an overlap region in each dimension same size as the maximum PSF gives the best result in terms of both ringing and mesh-grid artifacts.

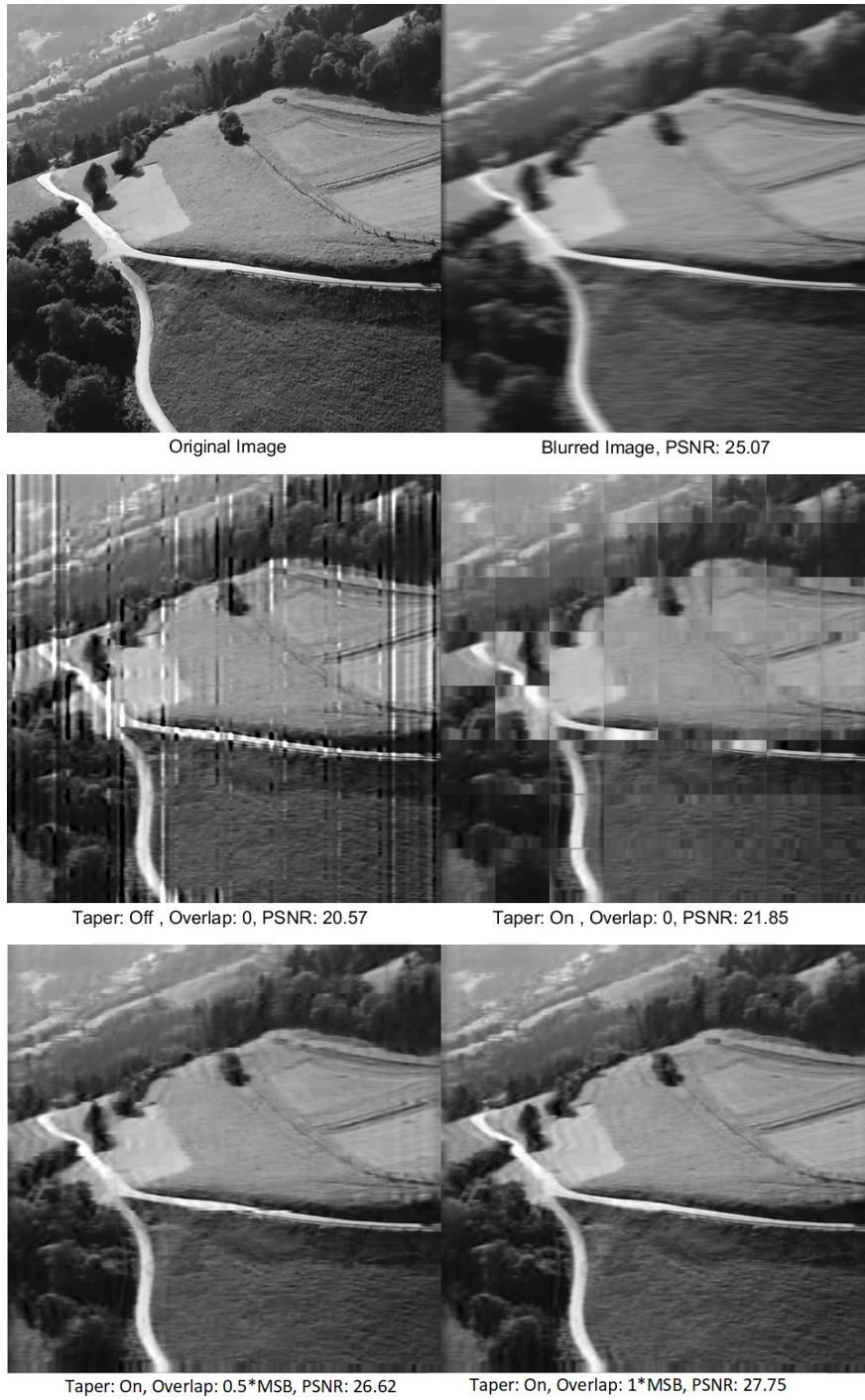


Figure 3.9: Shows the deblurred image results with and without edge tapering and overlap region. The PSNR values given in dB. The overlap region started from 0 pixel to maximum blur size of the blur kernel. The best quality obtained with employing edge-tapering and overlap size of MBS.

3.3 Proposed Method of Adaptive Mesh-Size

Dividing into meshes and performing deblurring in each grid is proposed in some studies [6]. We take this idea and further improved it with adaptivity of the grid size. This concept can be seen in Fig.3.10. The basic idea of adaptive-mesh size algorithm is obtaining a prior information about non-uniformity of the blur present in image and change the mesh-size according to this information. We use variance of motion vectors magnitude to measure this non-uniformity and deal with spatially-variant blur effectively.

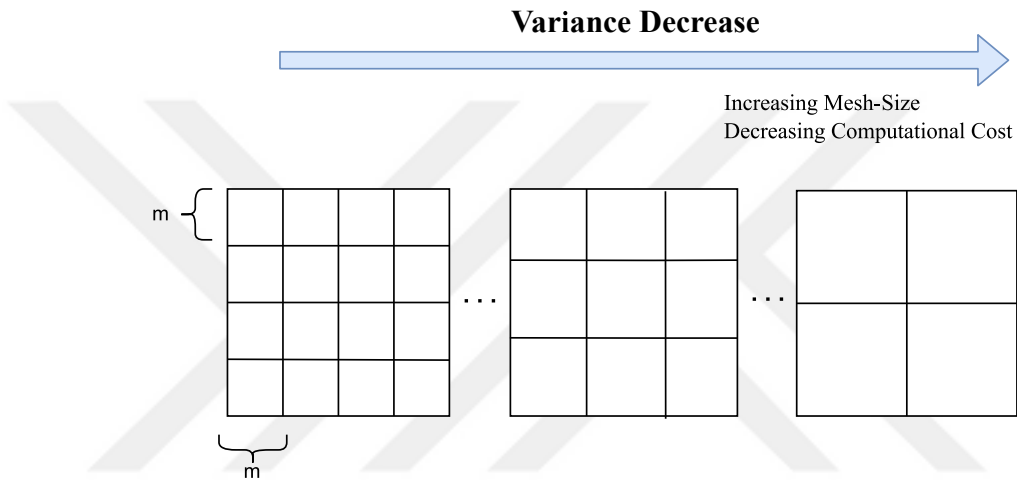


Figure 3.10: Adaptive mesh-size illustration. As non-uniformity in blur decrease, computed frame variance also decreases. This allows us to increase mesh-size and decrease computational cost.

The type of camera motion affects the characteristic of the blur. Increase in the rotation around optical axis (roll) will increase the non-uniformity of the motion. This will violate the assumption of the uniform motion blur. In other words, as roll movement increase, the accuracy of the estimated PSF of each grid decrease. This reduces the quality of the deblurring. Ideally, as the mesh size decreases, the PSF's gets closer the ground truth, and estimation error decreases. However, blur magnitude for the estimated PSF is a limitation on the grid size. The grid size must be greater than the PSF size for successful deblurring process. Our adaptive mesh-size decision algorithm searches this minimum grid size for best PSF estimation in order to maximize PSNR gain. It finds the closest mesh-grid dimension from the maximum size of blur

magnitude (MSB) obtained from IMU readings. Outcome of the possible smallest mesh-grid size is the computational cost which exponentially increases with smaller grid-size. Taking the speed of the deblurring process into consideration, we have observed that if the roll movement of the camera is small, the motion vectors become uniform. This leads us to increase the mesh-size to optimal point with small loss in quality, but large gain in speed.

3.3.1 High-Quality Adaptive Mesh-Size Algorithm

The maximum PSNR gain is obtained when the estimated PSF error is lower. This PSF estimation error can be reduced by decreasing the mesh-size to smallest possible value. However same limitations are available in terms of mesh-size such as ringing artifacts, mesh-grid line artifacts and blur magnitude. In previous sections, we determined the overlap region and dealt with the artifact problems. In Fig. 3.11, a single grid structure with overlap region and edge tapering is given.

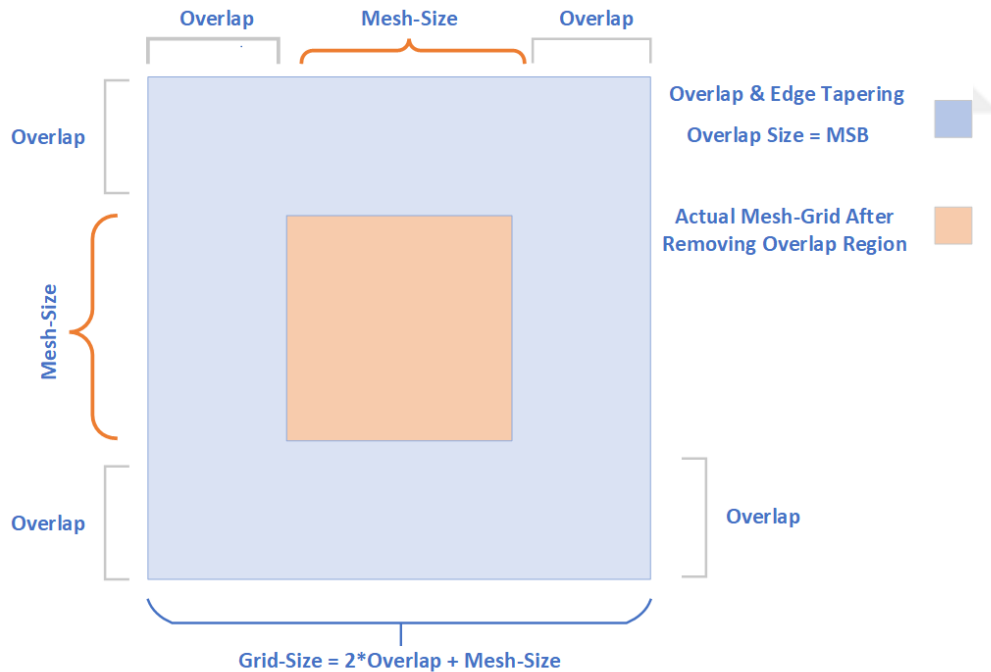


Figure 3.11: Shows the one grid cell structure that goes through the deblurring process with overlap region. Blue area is removed after deconvolution and orange part stays for actual deblurred image. The overlap region is determined as MSB previously.

The orange area stated as mesh-size are combined together and form deblurred image. Overlap region and edge tapering are in the blue area as the size of MSB in all dimensions given in (3.1) determined before in Fig.3.9. In terms of minimal grid-size we have two constraints, grid size should be greater than MSB and for edge tapering should be two times MSB as stated in [27]. Our method of overlap deals with these two issues since overlap size is selected equal to MSB in each dimension, which adds total grid size two times MSB.

$$Overlap = MSB = \max(\text{size}(H_{i,j})) \quad (3.1)$$

where i, j are the mesh-grid numbers in row and column respectively. This overlap region structure allow us to go into very small mesh-sizes. Problem in this is the most of the computation comes from the overlap region not from the deblurring process. For example, assume we have 30 pixels blur magnitude, and we selected [2x2] pixel mesh-size, the overlap region is 30 pixel in each dimension. The total grid-size that will be processed is [62x62] pixel but actual used part for deblurred image is [2x2] pixel region. Almost all of the processing time consumed in overlap region which will be later removed. Taking this into consideration, we determined that the mesh-size should be greater or equal to overlap size. By doing this, the computation time can reflect the effect of mesh-size, not dominated by overlap region.

Taking these assumptions and constraints, the maximum PSNR gain is obtained with smallest possible mesh-size represented as mesh-size high quality (MS_{HQ}) in (3.2). The magnitude of the blur is a limit for smallest mesh-size along with edge-tapering and overlap region time consumption. With overlap region in both sizes, the minimum mesh size should be two times the blur magnitude for each cell.

$$MS_{HQ} = 2 * MSB = 2 * \max(\text{size}(H_{i,j})) \quad (3.2)$$

where i, j are the mesh-grid numbers in row and column respectively.

3.3.2 Speed-Quality Balanced Adaptive Mesh-Size Algorithm

The developed adaptive high-quality (HQ) mesh-size algorithm performance is satisfying in terms of quality. However, the computation time is not acceptable in all situations. The further developing the adaptivity idea, we start to investigate the trade-off between quality and speed. In below Fig.3.12, PSNR and processing time versus mesh-size results of a sample image are given. The processing time increases exponentially towards to smaller mesh-sizes while PSNR gain stays almost same with little increase (purple box region). Our goal is to find some speed-quality balanced (SQB) point in the region shown in green box where reasonable PNSR-gain with small processing time can be achieved.

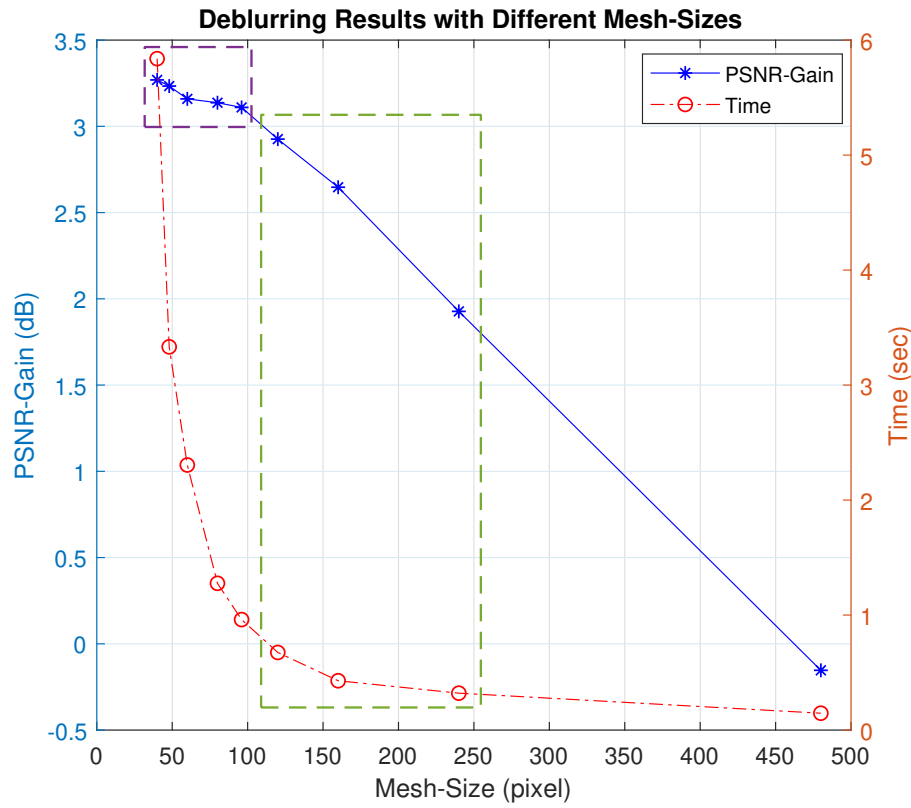


Figure 3.12: PSNR-Gain and processing time versus mesh-size graph of sample image with variance of 12-pixel. Computational cost increases exponentially while increase in PSNR-Gain is more linear. Selecting a mesh-size in green boxed area gives a reasonable computation time and gain.

The speed-quality balanced mesh-size should take into consideration the non-uniform blur. It should select a larger mesh-size when the differences of PSFs between regions is small, and should go to smaller mesh-sizes when this difference is large. This information is represented using the variance parameter. We use the variance of the motion vectors magnitude along the whole image for the information about level of blur non-uniformity.

3.3.3 Experimental Determination of Speed-Quality Balanced Mesh-Size

In order to find a relation between variance, computation time and deblurring quality, we generated synthetic datasets and experimentally look for optimal mesh-size by hand labeling. The dataset generated with IMU samples which are collected while walking. This dataset is explained in the Chapter 4, section 4.2 as Dataset-1. First, Dataset-1 is used to obtain a variance versus mesh-size graph by hand selecting the speed-quality balanced results as considering the green boxed area. Then, an exponential curve is fitted which gives us the speed-quality balanced mesh-size using variance information of blurred image.

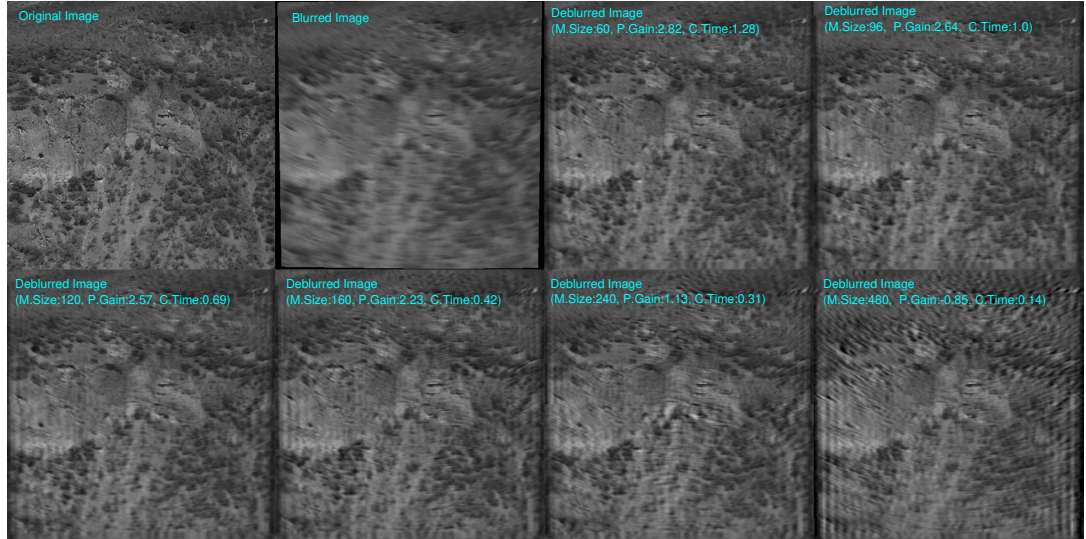


Figure 3.13: Shows original, blurred and deblurred images with different mesh-sizes. The mesh-size information represented as 'M.size', PSNR-Gain in db as 'P.Gain', and computation time in seconds as 'C.Time'. The selection of the speed-quality balanced points conducted based on PSNR-Gain, computational cost and visual quality.

A sample blurred image is deblurred with different mesh-sizes and obtained results are given in Fig.3.13. Visual results, mesh-sizes, PSNR-gains and computation times of each deblurred image is provided. This sample is given for better understanding of optimal points selection process. The gain difference between mesh-sizes 96,120 and 160 pixel is small. Considerable drop in quality happens at mesh-size 240 pixel with 2.23 dB to 1.13 dB. This should be a stop point for maximum mesh-size. Best quality result is obtained at mesh-size 60 pixel with PSNR-gain 2.82 dB. However, the computation time is 1.26 seconds approximately 10 times slower than full-frame (mesh-size 480 pixel) deblurring. When we look at the mesh-sizes 96 and 120 pixel, we see a very small difference between PSNR-gains as 0.07 dB. So, just by looking at the gain differences and computational time an optimal point emerges. The optimal points are selected by following this method on each frame of prepared synthetic dataset.

In addition, the conventional full-frame (mesh-size 480 pixel, right bottom image in Fig.3.13) deblurring result is -0.85 db which is clearly lowest visual quality among all deblurred images. Deblurring decreases the quality of the blurred images instead of enhancing because the PSF is estimated for its center which is not valid for all frame under rotational movement along optical axis. This shows us how mesh-division method is effective dealing with spatial-variant blur.

Procedure of the selecting speed-quality balanced point is explained using the results of a sample blurred and deblurred images in above Fig.3.13. Using this approach, we selected some SQB mesh-sizes on the Dataset-1 and results are given in Fig.3.14. The data is sorted for better observation of both effect of variance, mesh-size and time among each other.

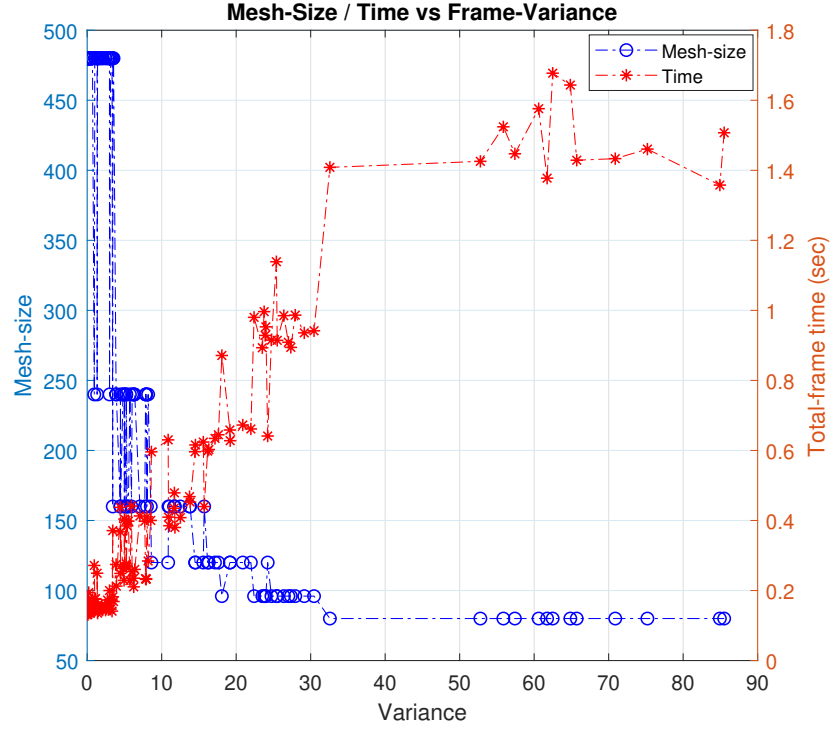


Figure 3.14: Variance versus Time and Mesh-size results of selected speed-quality balanced points from Dataset-1 (Explained in the Chapter 4, Section 4.2). The Selection process conducted with hand-labeling. Mesh-sizes are common divisors of 480x480 image.

The variance versus speed-quality balanced mesh-size graph in Fig.3.14 is constructed based on two main points. The first critical point is PSNR-gain and second one is computation time. In terms of PSNR-gain, our HQ mesh-size algorithm converges the minimal mesh-size that can be possible. This minimal mesh-size gives the best quality deblurring result. However, this result comes with high computation time as it can be seen from the results in Fig.3.12. In some cases where variance is high, going to minimal mesh-size is logical and necessary. However, if variance is small, which means spatial-variance is low, using HQ mesh-size is not efficient since the difference between estimated PSFs is negligible. In that case, increasing the mesh-size does not affect the PSNR-gain too much but decreases computation time dramatically. So, there is a relation between computation time and PSNR-gain which is affected by variance level of blurred image.

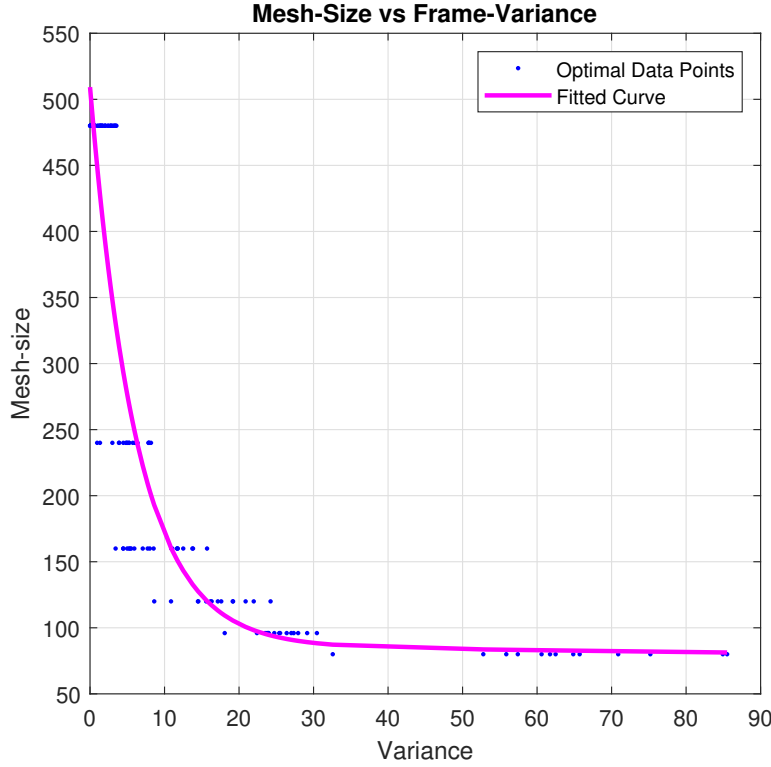


Figure 3.15: An exponential curve is fitted to speed-Quality balanced data points in the Dataset-1 hand-labeled results shown in Fig.3.14

After obtaining the speed-quality balanced point results in Fig.3.14, it is observed that an exponential relation exists between variance and mesh-size. In Fig.3.15, the result of an exponential curve fitting to given data is shown. The curve gives the SQB mesh-size when the variance of the frame is provided. The speed-quality balanced mesh-size is represented as $MS_{SQB}(var)$ and equation is given as:

$$MS_{SQB}(var) = 422 * e^{(-0.16*var)} + 87.25 * e^{(-0.000824*var)} \quad (3.3)$$

where var represents the blur magnitude variance which indicates the non-uniformity level present in the blurred frame. In addition, there is blur magnitude constraint in the SQB mesh-size which is also the limit of the MS_{HQ} . Therefore, SQB mesh-size algorithm cannot be smaller than the two times of the blur magnitude which means that SQB cannot be smaller than the MS_{HQ} .

To sum up, the developing phase and dealing with the common problems of artifacts using the overlap structure and edge-tapering are explained in this section. This investigation for improving the performance of the inertial-aided deblurring methods resulted in two adaptive mesh-size algorithms named as MS_{HQ} and $MS_{SQB}(var)$ are introduced. First one look for high-quality results without considering computational time. The second one looks for an optimal point mesh-size regarding both the quality and computation time according to the variance.





CHAPTER 4

EXPERIMENTS AND TEST RESULTS

In previous parts, proposed methods of adaptive mesh-size algorithms and their developing phases are explained in detail. In this section, we evaluated their performances and compare them with classical and constant mesh-sized deblurring methods. This section includes the performance metrics for objective evaluation, datasets which includes different types of motion blurred frames and comparison results of our method followed by a limitation and observation part.

4.1 Performance Metrics

The performance evaluation of image processing algorithms is an important field. Various image quality (IQ) metrics are developed for effectively measuring the quality and performances of evolving new methods. The IQ metrics can be divided into two categories as subjective and objective. The involvement of the human observers and perspective is the subjective approach. The objective approach uses numerical values and mathematical expressions to assess IQ. The objective IQ metrics are the most commonly used and most reliable ones in terms of performance. Some of the commonly used image quality metrics are explained as follows.

- i **MSE (Mean Squared Error)** is a metric that calculates the average square error between the original image $I_{or}(i, j)$ and the degraded image $I_{deg}(i, j)$ as given in (4.1):

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (I_{or}(i, j) - I_{deg}(i, j))^2, \quad (4.1)$$

where M and N represents the column and row number indexed with i and j . The lower MSE indicates the better quality since it measures the error between the two images.

- ii **PSNR (Peak Signal to Noise Ratio)** is an IQ metric for the ratio between the maximum possible power value of a signal and the power of corrupting noise that affects the quality of its representation. PSNR is given in (4.2) as:

$$PSNR = 20 \log \left(\frac{MAX_I}{\sqrt{MSE}} \right) \quad (4.2)$$

where MAX_I represents the maximum possible value for the image. Higher PSNR indicates higher quality. Note that, PSNR metric is in logarithmic scale and unit is decibels (dB).

- iii **NCC (Normalized Cross Correlation)** is a metric that measures the similarity between two images. The main advantage of it is insensitivity to the illumination changes. Its representation is given in (4.3) as:

$$NCC = \frac{\sum_{i=1}^M \sum_{j=1}^N (I_{or}(i, j) \cdot I_{deg}(i, j))}{\sum_{i=1}^M \sum_{j=1}^N (I_{or}(i, j) \cdot I_{or}(i, j))} \quad (4.3)$$

- iv **SSIM (Structural Similarity Index)** is a perceptual metric that quantifies the image degradation proposed in [40]. The index is based on three factors which are luminance (l), contrast (c), and structure(s) comparison. It is calculated from local windows in the image defined by x and y of common size $N \times N$. The representation of SSIM given as:

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \quad (4.4)$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad (4.5)$$

$$s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \quad (4.6)$$

where μ_x and μ_y are the mean intensities of the signals. The σ_x , σ_y are the standard deviations, and σ_{xy} is cross-covariance of the signals. The SSIM metric is calculated by combining these three terms given in (4.4), (4.5) and (4.6) with the weights of α , β and γ given in 4.7 as:

$$SSIM(x, y) = [l(x, y)^\alpha \cdot c(x, y)^\beta \cdot s(x, y)^\gamma] \quad (4.7)$$

v **LR (Logistic Regression)** is a learning based no-reference image quality metric proposed in [41]. They conducted a massive user study along with employing traditional IQ methods to train a network for considering naturalness, noise, ringing and sharpness. Details of the LR metric can be found in [41].

Some of the prominent IQ metrics in the literature have been explained. The PSNR and SSIM metrics are especially widely used in the qualification of the image deblurring algorithms. However, they require the reference image. Therefore, We perform our tests on the synthetically generated datasets which allow us to use these metrics. For evaluating our proposed deblurring algorithm performance, we selected the PSNR metric.

4.2 Datasets

To evaluate the proposed methods and their performance, we have synthetically prepared 5 different datasets using the IMU recordings. The inertial measurements are taken from MPU-6050 model IMU device during fast walking for capturing images having wide range of blur characteristics and magnitude. 5 datasets are created named as Dataset-1 to Dataset-5 for different purposes with different contents and blur characteristics. Dataset-1 is generated for experimental determination of SQB mesh-size. We analyze the deblurring results of this dataset and determine the optimal mesh-sizes in terms of speed and quality by hand selection. The Dataset-2, Dataset-3 and Dataset-4 are used for the evaluation of our methods along with the classical deblurring approach. Up to this point, the generated datasets are non-uniform and non-linear. However, the Dataset-5 is generated by 3 blur kernels and it is uniform. This dataset is used for the evaluation of some pioneer non-blind deconvolution algorithms. The selected algorithm is used in the deconvolution part of our adaptive mesh-size method.

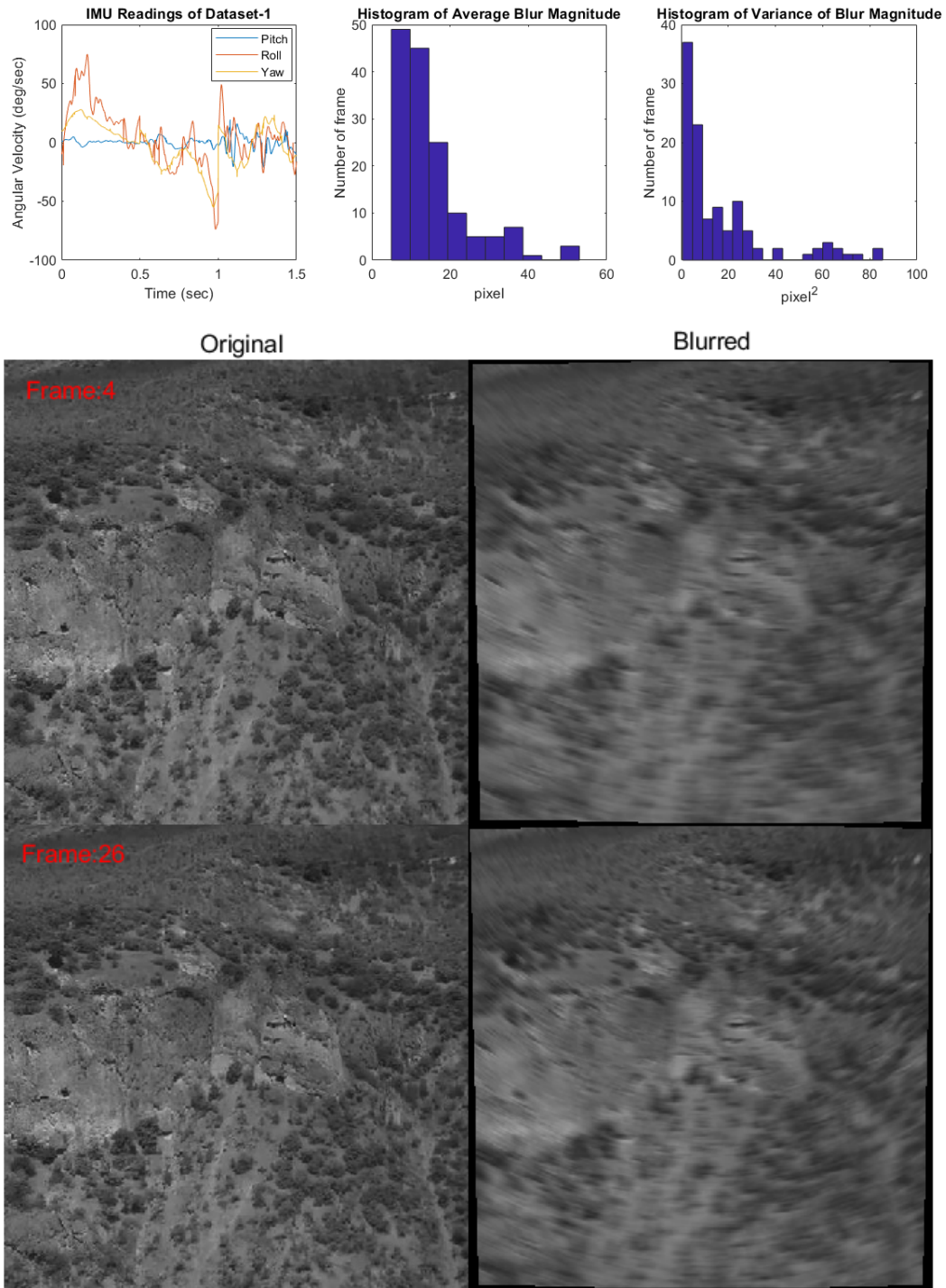
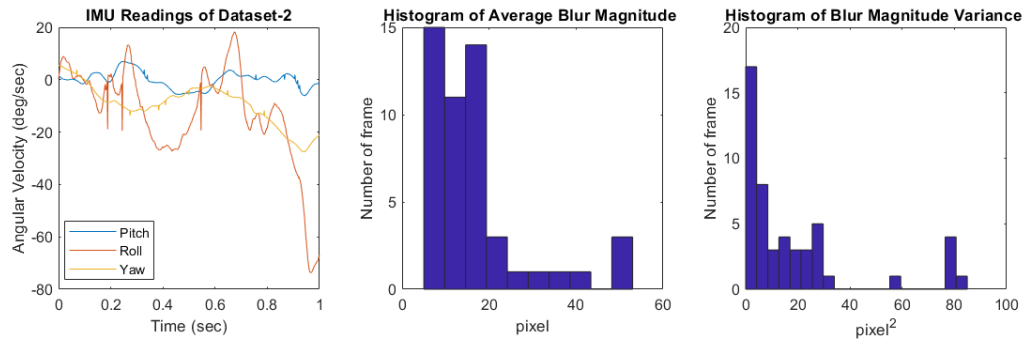


Figure 4.1: Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-1. Blur magnitude calculation conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.



Original

Blurred

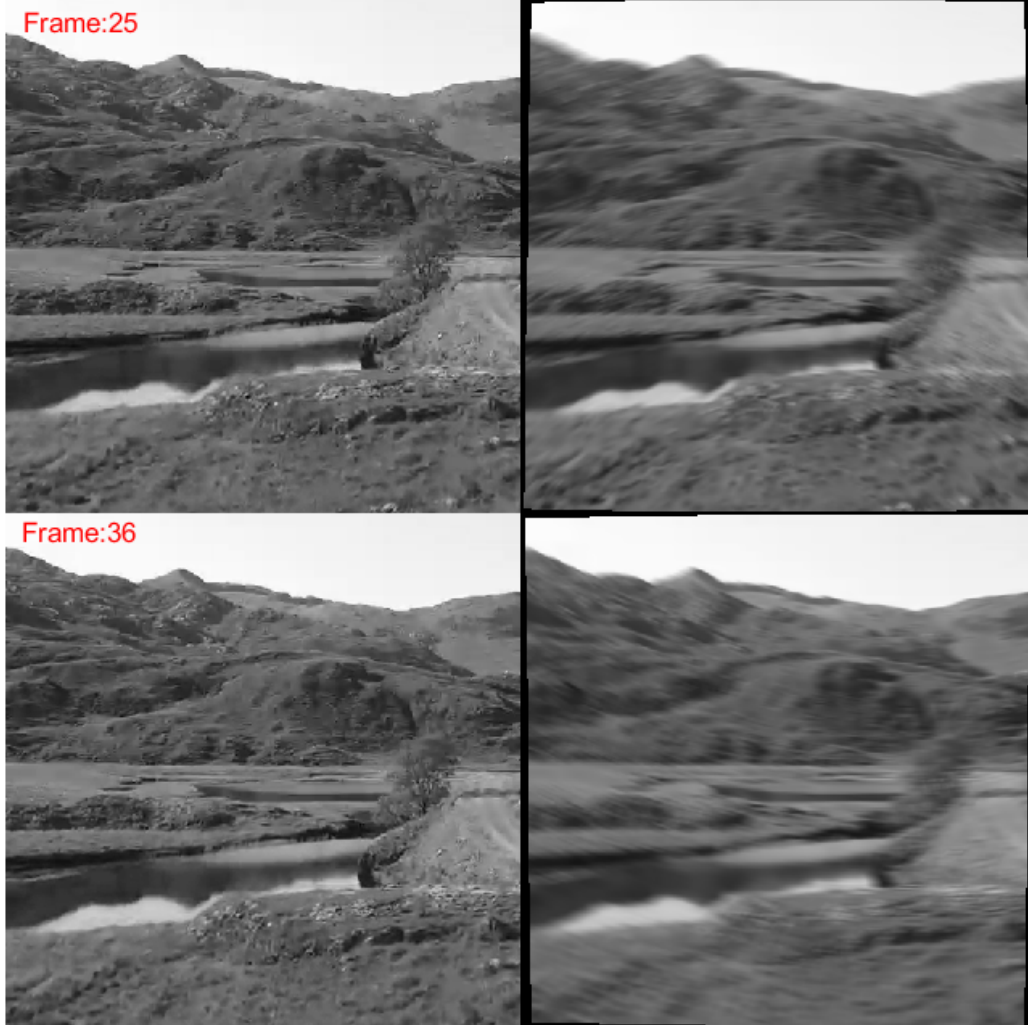


Figure 4.2: Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-2. Blur magnitude calculation is conducted by selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.

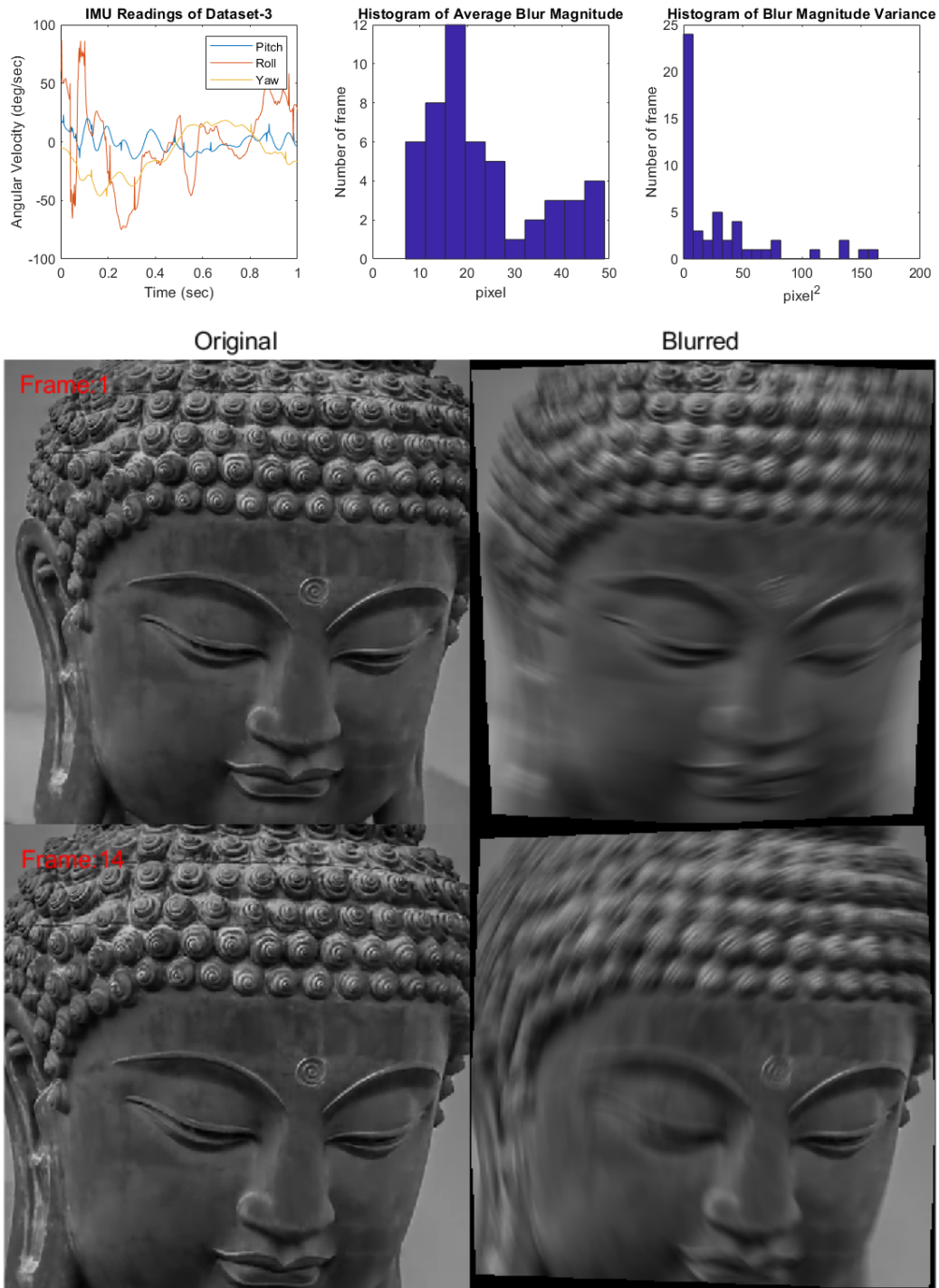
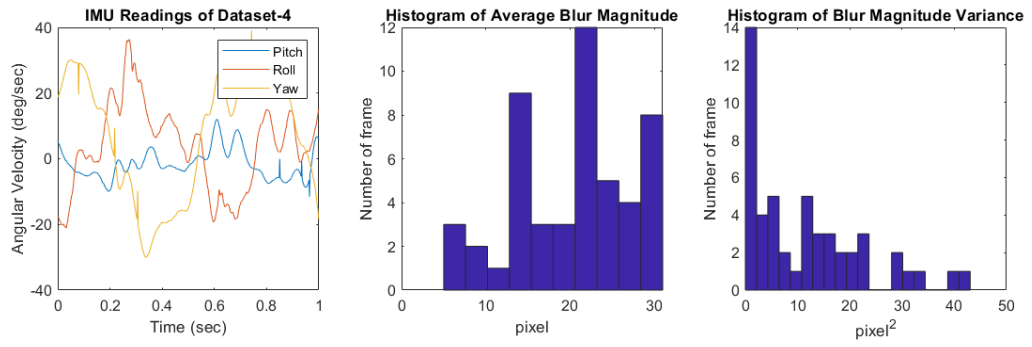


Figure 4.3: Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-3. Blur magnitude calculation is conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.



Original

Blurred



Figure 4.4: Shows the inertial measurements, histogram of average blur magnitude and variance of it for each frame in the Dataset-4. Blur magnitude calculation is conducted selecting the maximum size of the estimated PSFs. Two samples of original and blurred images are shown in bottom.

The synthetic blur is generated by determining the motion vectors and corresponding PSFs from the inertial measurements as stated in Chapter 2. In synthetic blurring, PSF is generated for each pixel and blurred image is obtained by convolving sharp image for each one of them with its PSF.

Sample images and information about the blur characteristics of the Dataset-1 is given in Fig. 4.1. This dataset consists of 100 images having a size of 480x480 pixel. Histogram of average blur magnitude and variance are also provided. The calculation of blur magnitude for each frame is conducted by finding the sizes of blur vectors generated from the inertial measurements according to (2.23). This data indicates level of blur magnitude present in the dataset. Another information about the blur characteristic in the dataset is variance of blur magnitudes. It gives information about the non-uniformity level of the images and the dataset. Dataset-1 is used for the experimental determination of the speed-quality balanced mesh-size algorithm. The inertial measurements are taken while walking, and indicates an daily scenario.

Example original and blurred images from Dataset-2, Dataset-3 and Dataset-4 are shown in Fig.4.2, Fig.4.3 and Fig.4.4 respectively. The histogram of blur magnitude and variance graphs of the datasets are also provided. The difference of these datasets is the characteristics of the blur present in the images. The Dataset-2 includes more uniformly blurred images which can be seen from its variance except some spikes in a couple of frames. In other words, the dataset is more pitch and yaw dominant. On the other hand, the Dataset-3 includes some heavy rotational movement along the optical axis. This can be seen from the variances which reach high levels compared to the Dataset-2. The average blur magnitude is also distributed very well with involving large magnitudes. The final non-uniform dataset is Dataset-4. It includes a more balanced motion in terms of three rotational movements. Combination of similar magnitudes along the all axis are observed in this dataset.

The camera parameters are selected such that, focal length as 1.4 *mm* and sensor size as 3.39 μm . The camera is assumed to be ideal and aligned with the scene. In addition, MPU-6050 IMU is employed with sample rate 500 *Hz* for collecting inertial measurements. The exposure time is taken as 0.02 seconds so that 10 IMU samples are collected for each frame.

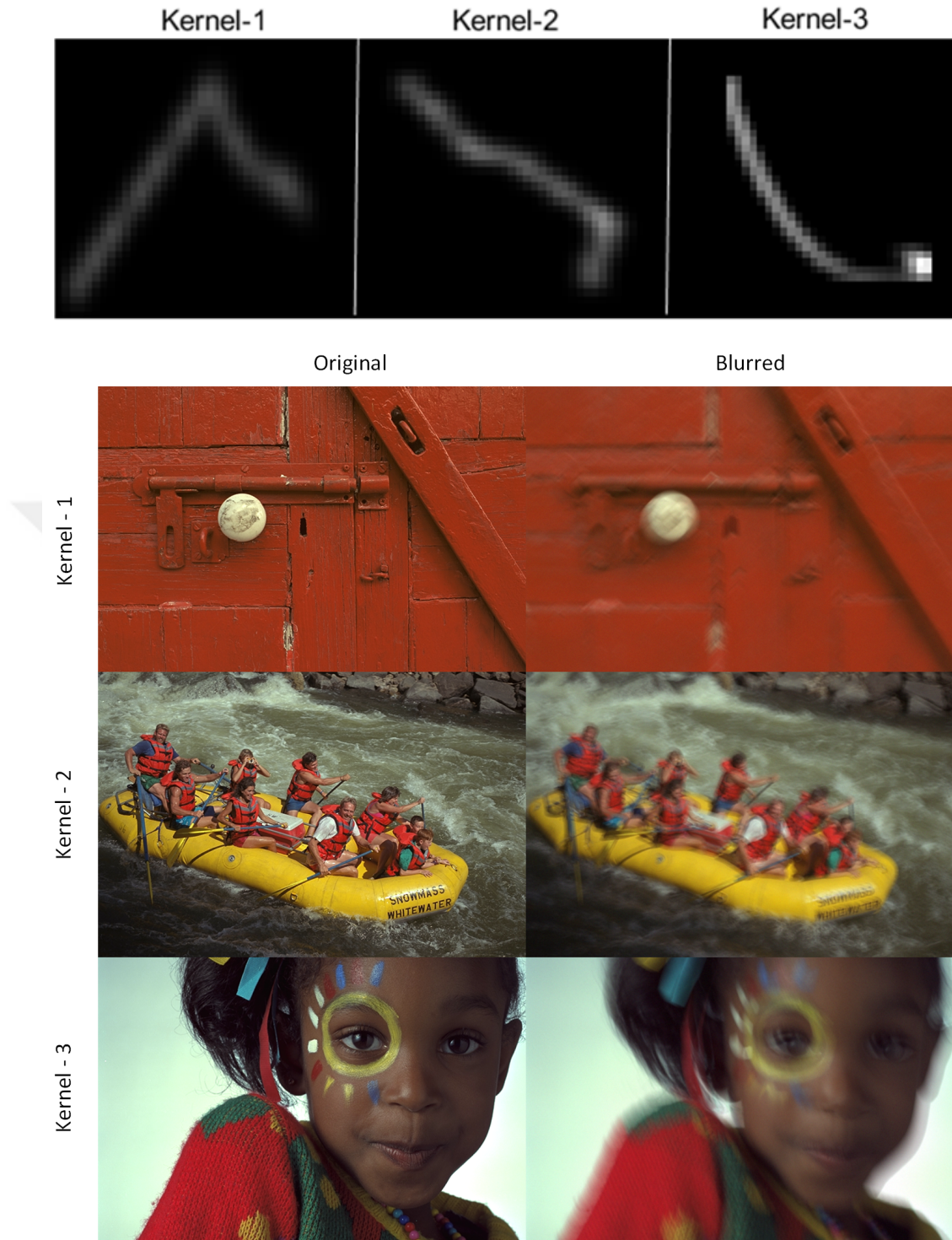


Figure 4.5: Shows the three blur kernels which used to generated uniform blur Dataset-5 for evaluation of non-blind methods. Sample original and blurred images are provided in bottom figure.

The final dataset is the Dataset-5 which is the only uniformly blurred dataset. Some sample images and the blur kernels that are used for the blurring the images are given in Fig.4.5. The images are taken from the Kodak Lossless True Color Image Suite [42]. 15 images are selected from this source and blurred by the three blur kernels. The uniform blur dataset is generated for evaluating some of the important existing non-blind deconvolution methods. These methods assumes uniform blur present in the image and their performance is significantly deteriorated under the non-uniform blur. For this reason, the evaluation is conducted on the uniformly blurred image frames.

4.3 Evaluation of Non-Blind Algorithms

In this section, we evaluated the performances of some non-blind deconvolution algorithms. Our main goal is to find a deconvolution method that performs well in terms of speed and quality. Any of the methods can be used for our algorithm, since we focused on improving the performance of deblurring by decreasing the PSF estimation error. The selected deconvolution methods are [43], [23], [20], [24],[22], [14] and [27] named as Lucy, Zhou, L_2 , TV, L_1 , Shan and Krishnan. These methods propose wide range of approaches to the problem of deconvolution such as total variation regularization, employment of l_1 and l_2 norms in minimization problem along with hyper-laplacian prior usage and iterative reweighted least square. The basic differences are explained in the introduction section. We don't go into details of the deconvolution algorithms, since the scope of our method is the performance improvement with PSF estimation and dealing with the non-uniform blur.

The Dataset-5 which consists of blurred images generated from three different non-linear blur kernel is used to evaluate the performances of the given non-blind deconvolution methods. The quality and computation time results are given in Table 4.1. Some of the visual results are also provided for observing the blur effect and visual quality results of the deblurring methods.

The results of seven non-blind deconvolution algorithms over Dataset-5 is given in Table 4.1. PSNR values of blurry and deblurred images of each method are given

Table 4.1: PSNR and computation time results of the algorithms for Dataset-5 generated from kernel-1, kernel-2 and kernel-3. Average gain (dB) and time (sec) results are given.

KERNEL-1	Deconvolution Methods							
Image	Blurry	Lucy	Zhou	L_2	TV	L_1	Shan	Krishnan
01	19.45	22.40	22.61	22.88	22.19	20.88	22.55	23.48
02	25.69	27.99	26.65	27.28	29.61	29.18	29.34	29.31
03	24.78	28.57	25.58	26.95	29.73	30.10	30.62	30.53
04	24.09	27.65	25.17	25.93	28.95	28.31	28.83	28.93
05	17.23	22.83	22.32	23.08	23.67	23.15	22.85	23.83
Avg.Gain	-	3.64	2.22	2.98	4.78	4.08	4.59	4.97
Avg.Time	-	2.15	0.16	0.14	2.39	12.21	3.74	0.72
KERNEL-2	Blurry	Lucy	Zhou	L_2	TV	L_1	Shan	Krishnan
06	23.10	26.96	26.83	26.19	28.96	28.51	28.36	28.98
07	21.13	25.00	25.31	24.17	26.22	25.83	25.68	26.27
08	23.86	28.03	27.41	26.67	30.19	29.86	29.80	30.05
09	18.27	21.06	21.33	20.47	21.47	21.09	21.02	21.62
10	20.24	24.49	25.07	23.89	25.81	25.31	25.14	25.88
Avg.Gain	-	3.79	3.87	2.96	5.21	4.80	4.68	5.21
Avg.Time	-	2.62	0.66	0.18	2.64	15.32	3.82	0.76
KERNEL-3	Blurry	Lucy	Zhou	L_2	TV	L_1	Shan	Krishnan
11	20.38	27.83	27.02	27.61	30.03	29.65	29.91	30.07
12	23.34	26.87	26.89	27.26	28.43	27.01	28.22	28.19
13	20.91	26.92	26.63	27.10	27.21	28.13	28.34	28.46
14	19.63	25.68	25.43	25.73	26.10	26.22	26.24	26.30
15	19.13	24.64	24.05	24.42	25.12	24.97	25.05	25.48
Avg.Gain	-	5.71	5.32	5.74	6.77	6.52	6.87	7.22
Avg.Time	-	2.69	0.63	0.18	2.52	15.40	4.16	0.82

with average gain and computation time. Top left corner of each subtables shows which blur kernel is applied to the images.

In Kernel-1 blurred images, the Krishnan algorithm has the largest gain with 4.97 dB. The TV and Shan methods followed it with the gains of 4.78 and 4.59 dB respectively. In terms of computation time, L_2 method is the fastest one with 0.14 sec. The closest computation time to this value is the Zhou's method with 0.16 sec. However, the quality gain of these methods are 2.98 and 2.22 dB respectively which are the worst ones between all of them. Similar results are obtained in the kernel-2 and kernel-3. The Krishnan method lead the quality in kernel-2 with average PSNR gain 6.64 *db*, and in kernel-3 with 7.22 dB. TV and Shan methods are very close to Krishnan method in terms of quality. The difference around 4-5% range. However, the computation times are five times higher than the Krishnan method.

The visual results of the methods can be seen in Fig.4.6, Fig.4.7 and Fig.4.8. For each blur kernel single image is selected for visual comparison between methods. The evaluations are conducted without any additional changes. The ringing artifacts for each method can be seen. Especially the Zhou and L_2 methods are suffering from the ringing. The methods TV, Shan and Krishnan methods are close in terms of visual quality.

Both PSNR gains and visual results clearly shows the superiority of Krishnan algorithm. We decided to use Krishnan method for our deconvolution step in the light of this evaluation. This method is a non-blind deconvolution method which assumes that uniform blur is present in the image and fails under the presence of non-uniform blur. We use the adaptive mesh-size concept in order to deal with this problem.



(a) Original Image



(b) Blurred Image with Kernel-1



(c) Zhou



(d) L_2



(e) TV



(f) L_1



(g) Shan



(h) Krishnan

Figure 4.6: Shows the original, blurred and deblurred images with the Zhou, L_2 , TV, L_1 , Shan and Krishnan methods. This image blurred with kernel-1. The results can be seen in Table 4.1, image-01.



(a) Original Image



(b) Blurred Image with Kernel-2



(c) Zhou



(d) L_2



(e) TV



(f) L_1



(g) Shan



(h) Krishnan

Figure 4.7: Shows the original, blurred and deblurred images with the Zhou, L_2 , TV, L_1 , Shan and Krishnan methods. This image blurred with kernel-2. The results can be seen in Table 4.1, image-05.

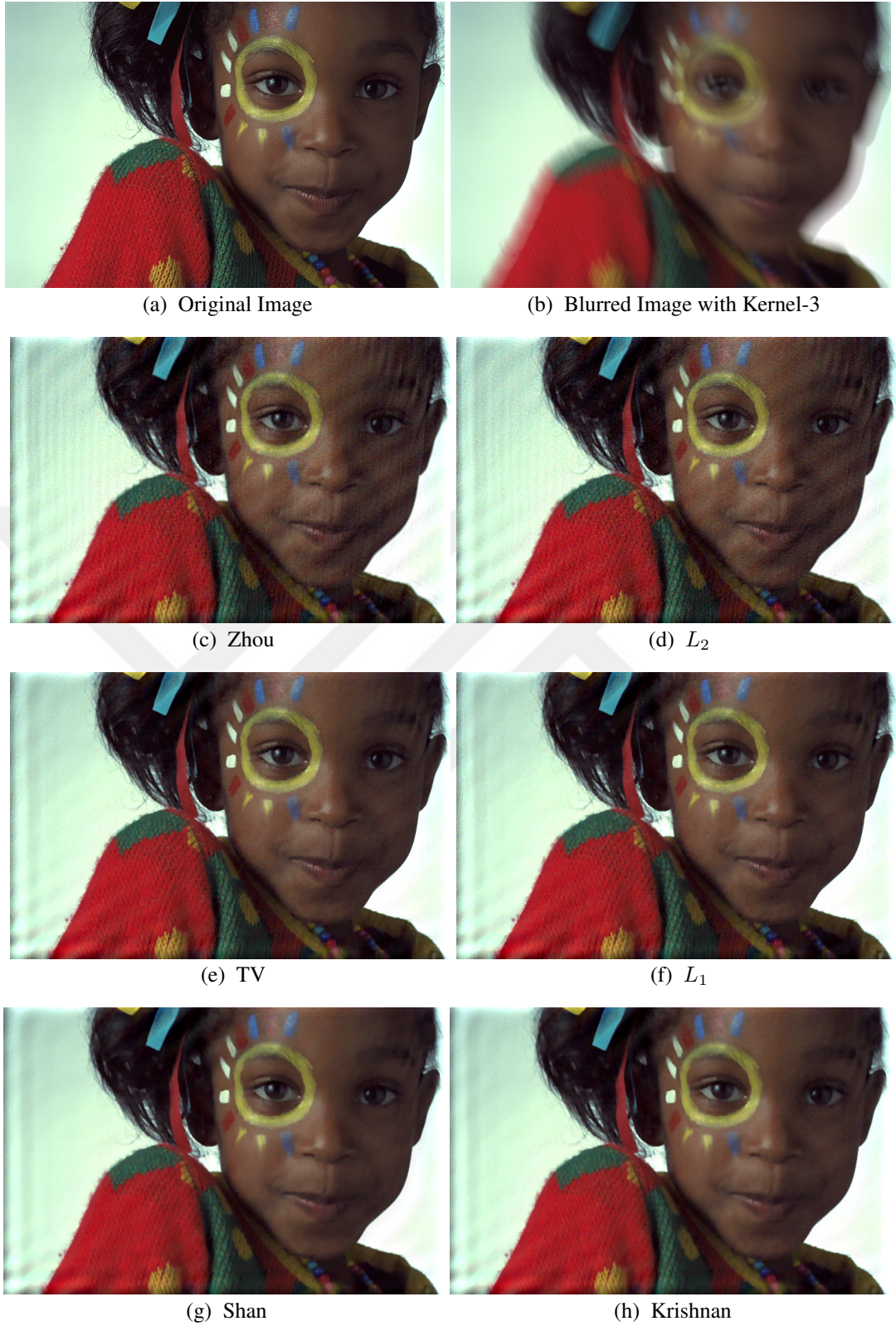


Figure 4.8: Shows the original, blurred and deblurred images with the Zhou, L_2 , TV, L_1 , Shan and Krishnan methods. This image blurred with kernel-3. The results can be seen in Table 4.1, image-06.

4.4 Experiments on Adaptive Mesh-Size Algorithm

4.4.1 Performance Evaluation of Adaptive Mesh-size Algorithms and Comparison with Classical Approach

Adaptive mesh-size algorithms named as high quality mesh-size (HQ) and speed-quality balanced mesh-size (SQB) are proposed in this thesis for improving the performance of existing deconvolution algorithms. In previous section, we selected the Krishnan algorithm for deconvolution part and conduct rest of the study employing this method in deconvolution part of our system. Our purpose in this part is showing the performance differences of HQ and SQB mesh-size algorithms along with full-frame traditional approach. For this purpose and also showing the different cases of blur characteristics present in the image sequence, the Dataset-2 is selected and some specific frames are examined in detail. The overall results on all of the available datasets and their performance evaluation are presented in the next section. This part looks for cases which has different range of blur magnitude and variances which reflects the concept of our adaptive structure and showing the full-frame classical approach under-performance specifically.

In Table 4.2, 10 frames are selected for evaluation and comparison from Dataset-2. The mesh-size, PSNR-gain and computational time of each method along with frame blur magnitude variance is given. Visual results are shown in Fig.4.9. Average gain of 3.24 db with HQ mesh-size, 2.18 dB with SQB mesh-size and 0.06 dB with full-frame results show the superiority of mesh-division algorithms over full-frame deblurring in terms of quality as expected. However, average computation times are 6.89 sec, 0.55 sec and 0.15 sec respectively.

Computational cost of HQ algorithm is considerably high compared the other methods. The speed-quality balanced mesh-size algorithm stands between these two methods in terms of both quality and computation time. It removes the unnecessary computation of high-quality mesh-size algorithm while dealing with the non-uniform motion blur unlike the full-frame deblurring.

Table 4.2: PSNR-gain (dB) and computation time (sec) results of adaptive and full-frame deblurring algorithms for ten frames selected on Dataset-2.

Image	Var	High-Quality			SQ-Balanced			Full-Frame	
		Size	Gain	Time	Size	Gain	Time	Gain	Time
1	13.9	32	2.89	6.50	120	1.56	0.61	-0.54	0.15
2	0.14	24	3.32	11.20	480	3.10	0.16	3.10	0.16
3	7.49	30	3.61	8.02	240	1.68	0.23	0.19	0.16
4	1.3	24	3.66	7.25	480	2.23	0.14	2.23	0.14
5	17.1	48	3.11	3.53	120	2.15	0.72	-1.56	0.17
6	10.2	48	3.26	3.43	160	2.46	0.41	-0.23	0.16
7	23.9	40	3.22	4.33	96	2.25	0.92	-1.10	0.15
8	30.4	40	2.96	4.24	80	2.19	1.23	-0.90	0.16
9	4.95	24	3.10	12.6	240	1.78	0.22	0.33	0.15
10	19.1	40	3.26	4.23	96	2.41	0.87	-0.91	0.14
avg			3.24	6.89		2.18	0.55	0.06	0.15

Looking at the results of these ten sample images, the efficiency of optimal mesh-size and the idea behind it can be seen. For example, image-2 with variance of 0.14, HQ mesh-size is 24 which is not necessary since the non-uniformity in image is negligible. So, the SQB mesh-size algorithm deblurs the image with mesh-size 480 pixel. The image 8, variance is 30.4, deblurred with HQ and SQB mesh-size 40 and 80 pixel respectively. Since variance is considerably large, SQB mesh-size algorithm decreases its mesh-size for more accurate PSF estimation.

In Fig.4.9, the visual results of some of the selected frames are shown. The visual results support our idea of optimal point and effectivity of adaptive mesh-size algorithm. In image-4, since the variance is small, the deblurring results of three methods are very close with small differences. However, as variance increases the quality differences become more obvious. The visual quality of full-frame deblurring highly decreases with spatial-variant blur. In image-4 with variance 1.3, PSNR-gain is obtained as 2.23 dB. In image-5 variance is increased to 17.1, PSNR-gain is -1.56 dB with obvious decrease in quality.

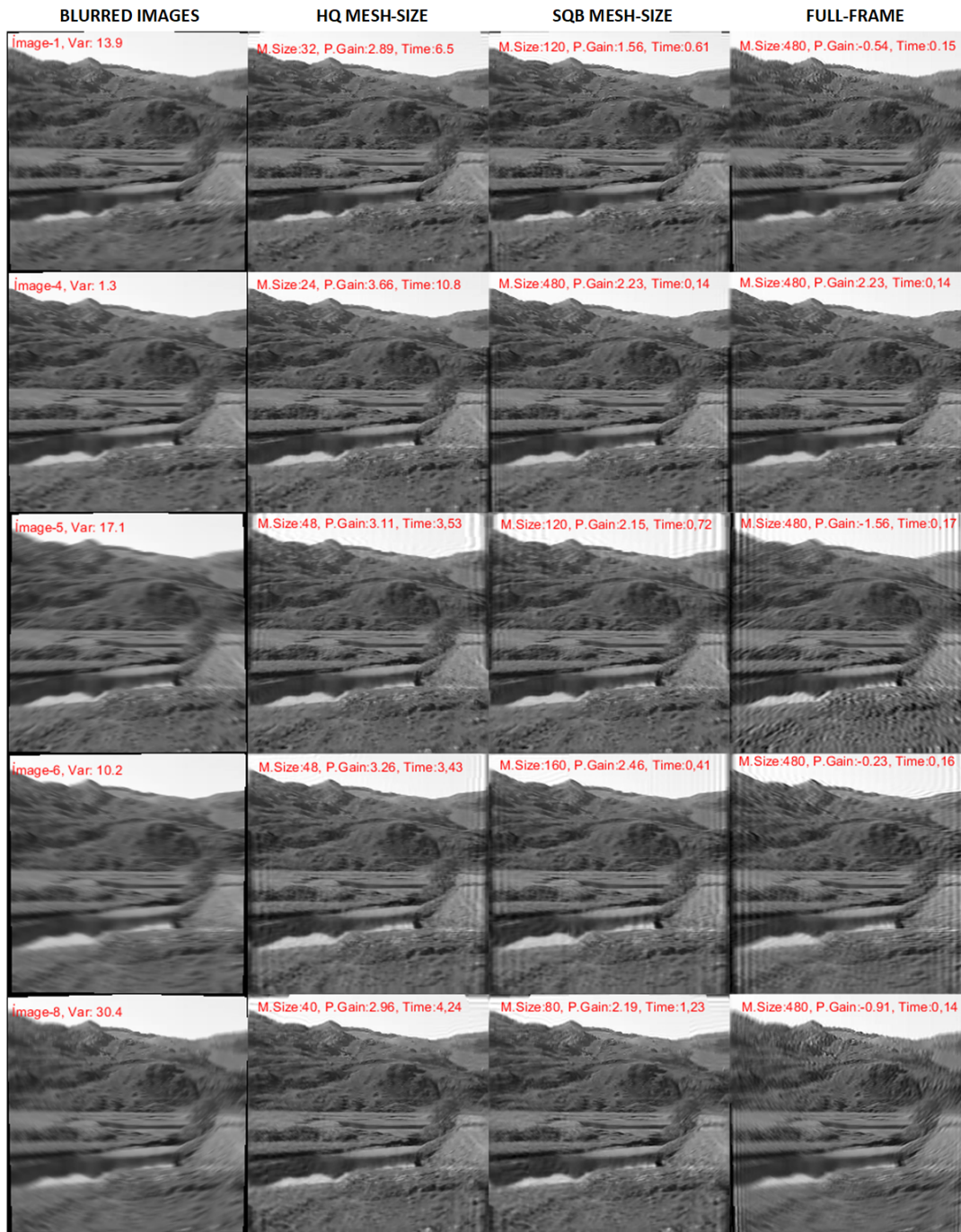


Figure 4.9: Shows blurred and deblurred images from Dataset-2 with different mesh-sizes. The mesh-size information represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in seconds as ‘C.Time’. The selection of the SQB mesh-size conducted based on PSNR-Gain, computational cost and visual quality. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and high cost.

4.4.2 Comparison of Adaptive and Constant Mesh-Size Structure

In this section, we compared our adaptive mesh-size algorithms and existing constant mesh-size approach in terms of deblurring performances. Our claim is that selecting the constant mesh-size is not effective in some situations and causes unnecessary computation. The constant approach either goes to small mesh-sizes in order to increase quality with better estimation of PSF. However, in the case of small blur magnitude or low rotational movement along the optical axis (roll) this is not necessary. The reason is that the motion can be assumed to be uniform and change of the PSF between regions is negligible.

Table 4.3: Comparison of the constant mesh-size versus proposed SQB and HQ algorithm. The average PSNR gain and computation time results on Dataset-2, Dataset-3 and Dataset-4 are presented. (*) cases correspond to blur magnitude exceeding mesh-size deteriorating PSNR performance.

Mesh-Size	Dataset-2		Dataset-3		Dataset-4	
	Avg.		Avg.		Avg.	
	PSNR Gain	Time (s)	PSNR Gain	Time (s)	PSNR Gain	Time (s)
480x480	1.34	0.15	-0.91	0.15	0.01	0.14
240x240	2.50	0.25	1.06	0.28	2.05	0.28
160x160	3.01	0.40	1.89	0.45	2.47	0.43
120x120	3.31	0.62	2.27	0.67	2.67	0.66
96x96	3.48	0.96	2.47	0.98	2.90	0.99
80x80	3.57	1.22	2.58	1.15	2.95	1.32
60x60	3.65	2.03	2.52*	2.21	2.89	2.05
48x48	3.74	3.12	2.39*	3.14	2.12*	3.06
40x40	3.49*	3.94	1.95*	3.98	1.45*	4.10
HQ	4.28	10.17	3.15	5.05	3.41	5.20
SQB	2.67	0.23	2.48	0.72	2.87	0.51

The comparison of our adaptive SQB (adaptive mesh-size) algorithm with the various constant mesh-size cases are given in Table 4.3. The Dataset-3 and Dataset-4 are

deblurred with given constant mesh-sizes and average gains and computation times are given. The table clearly illustrates that proposed SQB algorithm can achieve close to ideal PSNR gain in combination with low computational cost. Even if the user was able to select an ideal mesh-size constant through the video sequence, the SQB performance may not be achieved. In practice, even this apriori information is not available for such a selection. In contrast, proposed SQB algorithm does not utilize any apriori information since the adaptive mesh-size decision is based on blur variance within individual frames that can be computed in real time.

Therefore, we claim that the SQB algorithm outperforms the constant mesh-size methods. When we look at the results, for example, we see that SQB algorithm gives average PSNR of 2.87 dB with 0.51 sec computation time in Dataset-4. Our results stands between the results of mesh-size 96 and 160 in terms of average PSNR gain while achieving lower computational cost than both mesh-sizes. Similar results are observed in the Dataset-2 and Dataset-3 which is indicated by bold lines.

The other proposed method, HQ algorithm, stands at the top of all the methods in terms of PSNR gain as expected. However, its computation time is around ten times larger than the SQB method.

The difference between the results in three datasets is caused by their blur characteristics present in the frames. For example, frames in the Dataset-2 are more uniform and have small variance comparing the other ones. The Dataset-3 have highest average blur magnitude and variance. The Dataset-4 stands between two of them with moderate blur content. The results prove that our method can outperform both constant mesh-size implementation and full-frame deblurring methods. It directly converges to optimal mesh-size without prior information about the blur that is present in all of the video frames.

Note that the (*) mesh-sizes in given dataset means that the particular mesh-size is failed to deblur some frames because of the blur magnitude exceeding the mesh size. For this reason, the smaller mesh-sizes have bad performances. Normally they should have the best quality results.



Figure 4.10: Shows blurred and deblurred frames with different mesh-sizes from Dataset-3. The mesh-size is represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in sec as ‘T’. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and cost. The best quality is obtained in the HQ method and decreases from left to right except for frames with low variance.



Figure 4.11: Shows blurred and deblurred frames with different mesh-sizes from Dataset-4. The mesh-size is represented as ‘M.size’, PSNR-Gain in dB as ‘P.Gain’, and computation time in sec as ‘T’. As mesh-size increases, the quality of deblurring and computation time decreases while decreasing mesh-size results in high quality and cost. The best quality is obtained in the HQ method and decreases from left to right except for frames with low variance.

The visual results of some blurred frames from Dataset-3 and Dataset-4 are provided in the Fig.4.10 and Fig.4.11. The quality of the frames decreases from left to right. The HQ algorithms gives the best results as expected. The SQB stands in between full-frame deblurring and HQ method with moderate gains and computation time. Full-frame deblurring gives the worst performance except the frames which have very low variances.

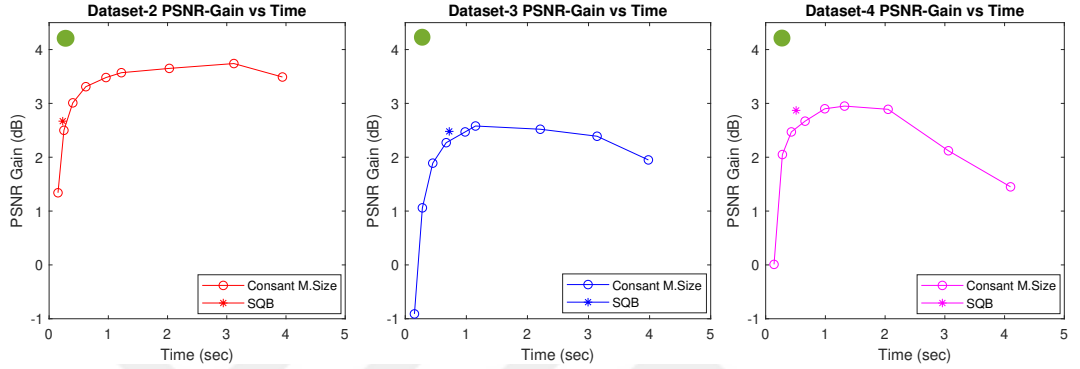


Figure 4.12: Shows the avg. PSNR gains and computation times of constant mesh-size and SQB method for three datasets given in Table 4.3. Constant mesh-sizes are (40,48,60,80,96,120,160,240,480). The SQB (*) results are over the constant mesh-size curves in all three dataset which indicates better performance. The green circle indicates the desired region where maximum PSNR gain with minimum time.

The results in Table 4.3 are plotted for better observation. The Fig.4.12 shows the average PSNR gain and computation time results of three datasets for constant mesh-sizes and SQB mesh-size. The green circle indicates direction wise desired ideal region where maximum PSNR gain is obtained in minimum time. Looking at these results, our method outperforms the constant-mesh size results along with full-frame deblurring (mesh-size 480). However, observing the result of Dataset-1, we realized that further improvement is possible in terms of quality. This can be achieved by modifying the SQB algorithm.

The modification of the SQB algorithm allow user to change mesh-size according to his/her desired performance requirement. For example, if the process is required to be completed in short time, selection of larger mesh-sizes decrease the computation

time. Also, this improve the further adaptivity of the algorithm since we generated SQB algorithm by hand-labeling and results maybe not optimal for some datasets and desired performance criterion of user.

4.4.3 Extending SQB Algorithm with a Tradeoff Control Parameter

SQB algorithm is generated by hand-labeling of optimal mesh-sizes in terms of speed and quality in a dataset. This speed-quality balanced mesh-size structure may not be valid or optimal in all datasets. Also, a user may want to change the mesh-size according to the desired performance. For this purpose, we modified the SQB method and add a tradeoff parameter (λ) and named it as modified speed-quality balanced (MSQB) mesh-size. The equation represented in (4.8) as:

$$\begin{aligned} MS_{MSQB}(\text{var}) &= \lambda * MS_{SQB}(\text{var}) \\ &= \lambda * (422 * e^{(-0.16*var)} + 87.25 * e^{(-0.000824*var)}) \end{aligned} \quad (4.8)$$

where var is the variance of the blur magnitude. Multiplying the SQB mesh-size with the λ , simply moves up and down the fitted curve described in the Fig.3.15.

The tradeoff parameter (λ) is changed from 0.1 to 4 with the step of 0.1 on the available datasets. The PSNR gains and computation results of each mesh-size, which is changed with λ , are united with the previous graphs in the Fig. 4.12. The combined results of each tradeoff parameter which is indicated as MSQB mesh-size, original SQB, and constant mesh-size deblurring results are plotted together in the given Fig.4.13. Looking at the obtained graphs, we see that MSQB mesh-size results formed a curve which is above the constant-mesh size counterpart. This indicates that in all situations the proposed method outperforms constant mesh-size and full-frame implementation, which is also a constant mesh-size method.

The MSQB curve is limited with HQ mesh-size. This is logical because we decrease the SQB mesh-size at a certain level by multiplying it with a value below 1. This pushes it to take smaller mesh-size. However, the blur magnitude is always a constraint for our method and non of the adaptive algorithms can go below that. So, with small tradeoff parameter, the MSQB converges to HQ mesh-size.

Results also show that, using tradeoff parameter the performance of our system can be changed easily. For example, in Fig.4.13 Dataset-2 results, SQB method (blue dot) is away from the desired left top corner where high PSNR and low computation time can be achieved. The MSQB can be changed to middle region in curve which gives more optimal results.

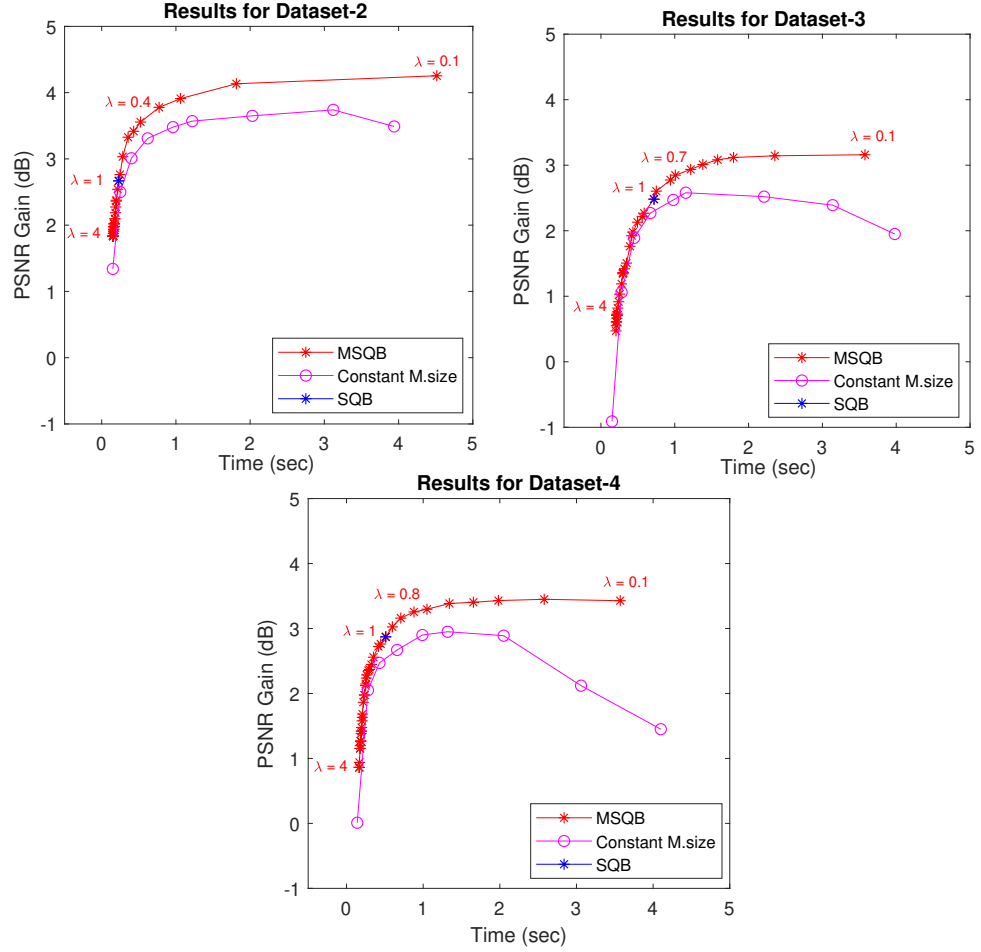


Figure 4.13: Shows the avg. PSNR gains and computation times of constant mesh-size, MSQB and SQB method for three datasets given in Table 4.3. Constant mesh-sizes are (40,48,60,80,96,120,160,240,480). λ is changed between 0.1 to 4 with the step of 0.1 which generates MSQB curves. These curves are always above the constant mesh-size curves in three datasets which indicates better performance. The blue SQB points indicates $\lambda = 1$ cases. The SQB in Dataset-1 is lower the bending point of MSQB point and further improvement is possible.

4.4.4 Observations and Limitations

The mesh-grid structure and proposed adaptive mesh-size concepts are tested and results are provided in previous sections. We have demonstrated that the mesh-size can be changed according to blur magnitude and variance of it in order to improve the quality and speed of the non-blind deblurring methods under the non-uniform blur. During experiment and developing phase, we have make some important observations and detected some limitations.

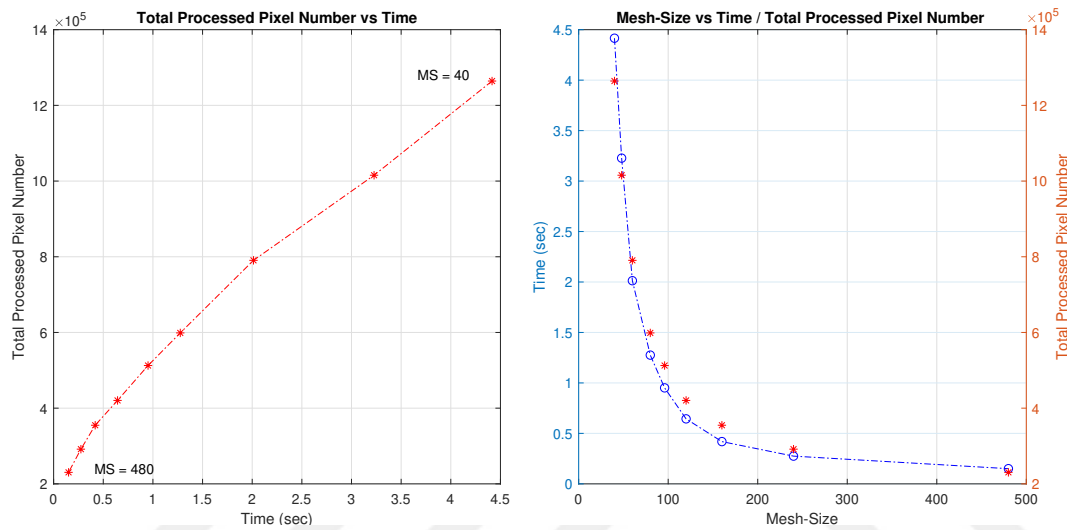


Figure 4.14: Shows total processed pixel number versus time (left), mesh-size versus time/total processed pixel number graphs (right). The first graph indicates the total pixel number and computation time is almost linear which means that the processing time of one pixel for all mesh-sizes are almost equal. In the second one, as mesh-size decreases the total processed pixel number increases exponentially with overlap region, which results in increase of computation time. MS stands for mesh-size.

In above Fig.4.14, total processed pixel number, computation time and mesh-size comparison graphs are given. This analysis is conducted for detecting the main source of the computational cost increase when smaller mesh-sizes are selected. The first graph (left) shows total processed pixel number and corresponding computation time. Total processed pixel number is obtained with summing the number of pixels of each grid. The graph shows that almost a linear relation between pixel total pixel number and processing time. This indicates that computation time for one pixel is almost

same in all mesh-sizes. When we look at the second graph (right), mesh-size versus time and total processed pixel numbers are given. As mesh-size decreases the total number of processed pixel is increases exponentially. This increase is caused by overlap region. When the mesh-size decreases, number of grids increases. Overlap region must be used for all of the grid borders for avoiding ringing and mesh-grid artifacts. These results show that main cause of exponential computation time increase when going smaller mesh-sizes is overlap region.

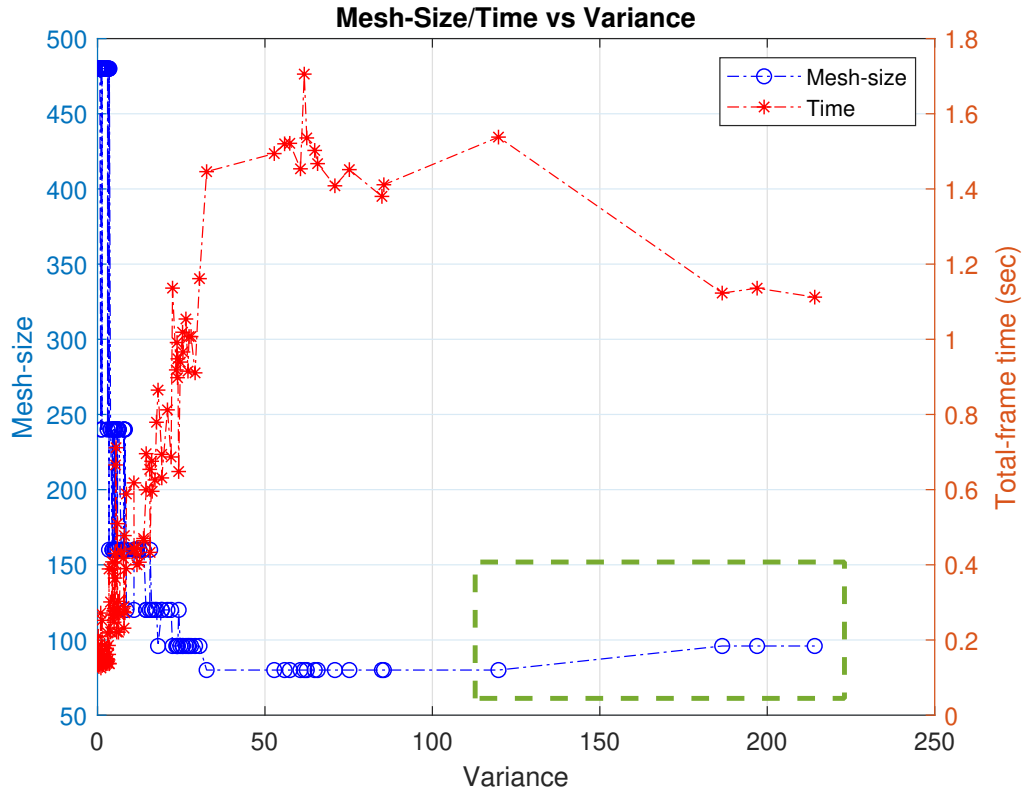


Figure 4.15: Shows the mesh-size/time versus variance graph of SQB points. The variance sample limit is increased from the previous version. The green boxed area shows the violation of the concept defined as increasing the variance decreases the mesh-size. This results in PSF estimation error and damages the restored image quality.

The blur magnitude variance parameter indicates the level of non-uniform blur present in the image. The idea is that as variance increases the mesh-size should be decreased for accurate PSF estimation. This is valid in most of the cases. However, as the vari-

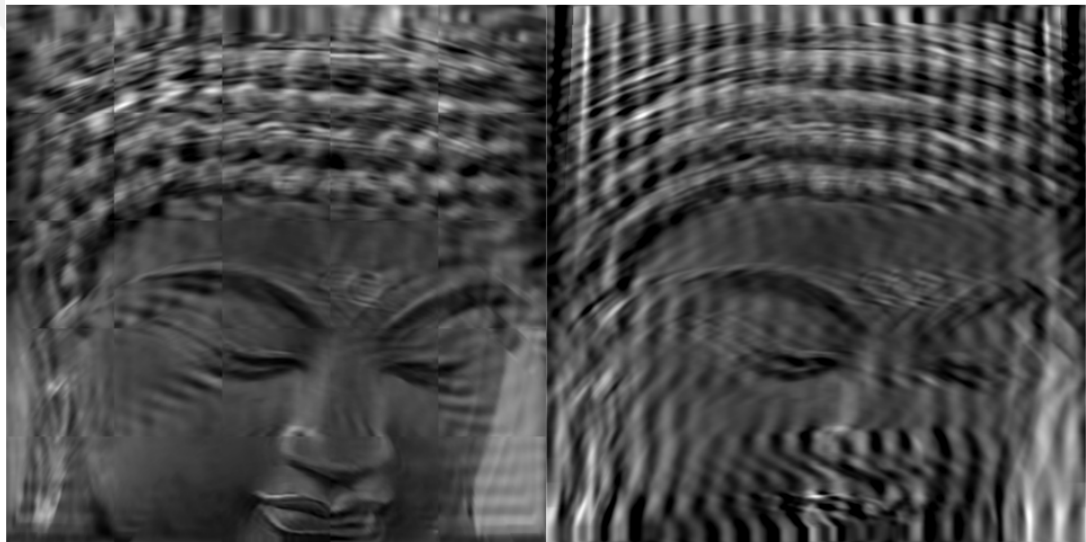
ance starts to increase in a certain level, this starts to change. The Fig.4.15, variance versus time and mesh-size graph is given. During the SQB mesh-size algorithm generation process, we fitted an exponential curve to this data and limit the mesh-size with blur magnitude. However, it is observed that under the high roll movements the mesh-size starts to increase as variance increase as it can be seen from green dashed box in the given figure. The reason is that the blur magnitude generated by roll motion also increases similar to variance. So, the blur magnitude keep mesh-size from decreasing. This situation also happens when high pitch and yaw motions are observed. They start to push blur magnitude up and put a limit on the mesh-size that the lower values are not possible. This results in higher mesh-size and increases the PSF estimation error.

In Fig.4.16, a sample case from Dataset-3 with variance of 164 is given. The high-quality algorithm in that case select mesh-size of 96 pixel similar to the SQB algorithm since when the variance is very high SBQ converges to HQ. The results along with full-frame (mesh-size 480 pixel) is given. As it can be seen from the restored frames and also PSNR results, the full-frame method completely fails to produce any improvement. The restoration results of HQ method is better than the full-frame, but it is not satisfactory with the grid border artifacts. Our method normally try to go smaller mesh-size in the presence of high variance, but blur magnitude puts a limit to it and causes deterioration of the quality.



Original Image

Blurred Image, Variance: 164



Deblurred Image, M.size:96,
PSNR Gain: -0.41

Deblurred Image, M.size:480,
PSNR Gain: -3.70

Figure 4.16: Shows the blurred frame-27 from Dataset-3, and restored images with HQ and full-frame mesh-sizes. The SQB and HQ are same in this situation because of high variance. In the restored frame with mesh-size 96, the borders of the grids are becoming visible as an artifact because of high-roll motion.



CHAPTER 5

CONCLUSIONS AND FUTURE STUDIES

In this thesis, we proposed an inertial measurement-based adaptive mesh-size structure for video motion deblurring. This structure divides the images into grids and estimates local PSFs for each grid cell. It is the first method that adaptively changing this grid size for accurate PSF estimation according to blur characteristics present in the image. Two adaptive mesh-size algorithms are developed, named HQ (high-quality) and SQB (speed-quality balanced). The HQ algorithm searches for the best quality results without considering computation time. On the other hand, the SQB method looks for a more balanced mesh-size regarding both computation time and quality. The deblurring algorithm structure is based on inertial measurements, which are utilized to obtain motion trajectories during the exposure period. The detailed derivation of the process is given for 3D rotational motion of the camera. After obtaining the motion present in the blurred image, a variance parameter is calculated for each frame representing the non-uniformity level of the blur from motion vectors magnitude. This parameter is used to change our mesh-size for accurate PSF estimation and efficient deblurring process. In addition to mesh-grid concept, complementary overlap region and edge-tapering is introduced to deal with the ringing and grid artifacts. The proposed method focuses on improving the performance of the non-blind deconvolution methods under the non-uniform motion blur. Results show a significant performance increase considering the computation time and visual quality of the deblurred image sequences evaluated in the given datasets compared to the traditional full-frame and existing constant mesh-size deblurring methods.

The HQ, SQB, and general structure of the adaptive mesh-size concept can be used in efficient employment of the non-blind deconvolution methods to deal with video

motion blur problem. The image recording platforms suffering from motion blur such as robots, UAVs or smartphones can utilize this method by using build-in IMU and increase the quality of the captured images.

The mesh-grid structure conducts deconvolution process individually for each grid cell which does not affect others' performances. This feature can allow us to perform parallel processing employing GPU or FPGA devices which will dramatically increase the speed of the deblurring. This development may lead to inertial-aided adaptive mesh-size algorithm for video motion deblurring which can work in real-time.

The proposed algorithm uses inertial measurements for PSF estimation process. The inertial measurement error in the IMU device directly affects the performance of the deblurring. This is especially important for uncalibrated low-cost inertial measurement devices. In addition, we assume that a large distance exists between scene and camera setup. This assumption can be deteriorate the performance of the proposed method under close range camera shootings.

REFERENCES

- [1] O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang, “Deblurgan-v2: deblurring (orders-of-magnitude) faster and better”, *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8877–8886, 2019.
- [2] H. Zhang, Y. Dai, H. li, and P. Koniusz, “Deep stacked hierarchical multi-patch network for image deblurring”, Jun. 2019, pp. 5971–5979. DOI: 10.1109/CVPR.2019.00613.
- [3] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. Khan, M.-H. Yang, and L. Shao, “Multi-stage progressive image restoration”, *ArXiv*, vol. abs/2102.02808, 2021.
- [4] N. Joshi, S. B. Kang, C. Zitnick, and R. Szeliski, “Image deblurring using inertial measurement sensors”, *ACM Trans. Graph.*, vol. 29, Jul. 2010. DOI: 10.1145/1833349.1778767.
- [5] Z. Hu, L. Yuan, S. Lin, and M.-H. Yang, “Image deblurring using smartphone inertial sensors”, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1855–1864. DOI: 10.1109/CVPR.2016.205.
- [6] J. Mustaniemi, J. Kannala, S. Särkkä, J. Matas, and J. Heikkilä, “Fast motion deblurring for feature detection and matching using inertial measurements”, *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 3068–3073, 2018.
- [7] J.-F. Cai, H. Ji, C. Liu, and Z. Shen, “Blind motion deblurring from a single image using sparse approximation”, in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 104–111. DOI: 10.1109/CVPR.2009.5206743.
- [8] J. Pan, D. Sun, H. Pfister, and M.-H. Yang, “Blind image deblurring using dark channel prior”, in *2016 IEEE Conference on Computer Vision and Pattern*

- Recognition (CVPR)*, 2016, pp. 1628–1636. DOI: 10.1109/CVPR.2016.180.
- [9] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman, “Removing camera shake from a single photograph”, *ACM Trans. Graph.*, vol. 25, pp. 787–794, 2006.
 - [10] D. Krishnan, T. Tay, and R. Fergus, “Blind deconvolution using a normalized sparsity measure”, in *CVPR 2011*, 2011, pp. 233–240. DOI: 10.1109/CVPR.2011.5995521.
 - [11] D. Perrone and P. Favaro, “Total variation blind deconvolution: the devil is in the details”, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2909–2916. DOI: 10.1109/CVPR.2014.372.
 - [12] L. Xu, S. Zheng, and J. Jia, “Unnatural l0 sparse representation for natural image deblurring”, in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1107–1114. DOI: 10.1109/CVPR.2013.147.
 - [13] A. Levin, Y. Weiss, F. Durand, and W. Freeman, “Efficient marginal likelihood optimization in blind deconvolution”, *CVPR 2011*, pp. 2657–2664, 2011.
 - [14] Q. Shan, J. Jia, and A. Agarwala, “High-quality motion deblurring from a single image”, *ACM SIGGRAPH 2008 papers*, 2008.
 - [15] K. Khongkrathan, A. Phonon, and S. Nuiphom, “An efficient blind image deblurring using a smoothing function”, *Applied Computational Intelligence and Soft Computing*, vol. 2021, pp. 1–10, Apr. 2021. DOI: 10.1155/2021/6684345.
 - [16] M. Asim, F. Shamshad, and A. Ahmed, “Blind image deconvolution using deep generative priors”, *IEEE Transactions on Computational Imaging*, vol. 6, pp. 1493–1506, 2020. DOI: 10.1109/TCI.2020.3032671.
 - [17] P. Tran, A. Tran, Q. Phung, and M. Hoai, “Explore image deblurring via blur kernel space”, *ArXiv*, vol. abs/2104.00317, 2021.
 - [18] J. Mustaniemi, J. Kannala, S. Särkkä, J. Matas, and J. Heikkilä, “Gyroscope-aided motion deblurring with deep networks”, *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1914–1922, 2019.

- [19] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, “Deblurgan: blind motion deblurring using conditional adversarial networks”, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8183–8192, 2018.
- [20] A. Levin, R. Fergus, F. Durand, and W. Freeman, “Image and depth from a conventional camera with a coded aperture”, *ACM SIGGRAPH 2007 papers*, 2007.
- [21] Z. Xu, H. Chen, and Z. Li, “Fast blind deconvolution using a deeper sparse patch-wise maximum gradient prior”, *Signal Processing: Image Communication*, vol. 90, p. 116 050, 2021, ISSN: 0923-5965. DOI: <https://doi.org/10.1016/j.image.2020.116050>.
- [22] Y. Wang, J. Yang, W. Yin, and Y. Zhang, “A new alternating minimization algorithm for total variation image reconstruction”, *SIAM J. Imaging Sci.*, vol. 1, pp. 248–272, 2008.
- [23] C. Zhou and S. Nayar, “What are good apertures for defocus deblurring?”, in *2009 IEEE International Conference on Computational Photography (ICCP)*, 2009, pp. 1–8. DOI: 10.1109/ICCPHOT.2009.5559018.
- [24] T. Chan and C.-K. Wong, “Total variation blind deconvolution”, *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998. DOI: 10.1109/83.661187.
- [25] A. Levin, “Deconvolution using natural image priors-lecture notes”, 2010.
- [26] L. Yuan, J. Sun, L. Quan, and H. Shum, “Progressive inter-scale and intra-scale non-blind image deconvolution”, *ACM SIGGRAPH 2008 papers*, 2008.
- [27] D. Krishnan and R. Fergus, “Fast image deconvolution using hyper-laplacian priors”, in *NIPS*, 2009.
- [28] S. H. Park and M. Levoy, “Gyro-based multi-image deconvolution for removing handshake blur”, *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3366–3373, 2014.
- [29] O. Sindelar and F. Sroubek, “Image deblurring in smartphone devices using built-in inertial measurement sensors”, *Journal of Electronic Imaging*, vol. 22,

- no. 1, pp. 1–8, 2013. DOI: 10.1117/1.JEI.22.1.011003. [Online]. Available: <https://doi.org/10.1117/1.JEI.22.1.011003>.
- [30] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Schölkopf, “Fast removal of non-uniform camera shake”, in *2011 International Conference on Computer Vision*, 2011, pp. 463–470. DOI: 10.1109/ICCV.2011.6126276.
- [31] G. K. Gültekin, “Single and multi-frame motion deblurring for legged robots: characterization using a novel fd-aroc performance metric and a comprehensive motion-blur dataset”, Ph.D. dissertation, Middle East Technical University, 2016.
- [32] R. A. Willoughby, “Solutions of ill-posed problems (a. n. tikhonov and v. y. arsenin)”, *SIAM Review*, vol. 21, no. 2, pp. 266–267, 1979. DOI: 10.1137/1021044. eprint: <https://doi.org/10.1137/1021044>. [Online]. Available: <https://doi.org/10.1137/1021044>.
- [33] A. Rajagopalan and R. Chellappa, “Motion deblurring: algorithms and systems”, 2014.
- [34] H. Wiener, “Structural determination of paraffin boiling points.”, *Journal of the American Chemical Society*, vol. 69 1, pp. 17–20, 1947.
- [35] E. Lee, E. Chae, H. Cheong, and J. Paik, “Fast motion deblurring using sensor-aided motion trajectory estimation”, *TheScientificWorldJournal*, vol. 2014, p. 649 272, Nov. 2014. DOI: 10.1155/2014/649272.
- [36] A. Mosleh, Y. elmi sola, F. Zargari, E. Onzon, and J. P. Langlois, “Explicit ringing removal in image deblurring”, *IEEE Transactions on Image Processing*, vol. 27, pp. 580–593, Feb. 2018. DOI: 10.1109/TIP.2017.2764625.
- [37] A. Mosleh, J. P. Langlois, and P. Green, “Image deconvolution ringing artifact detection and removal via psf frequency analysis”, Sep. 2014, ISBN: 978-3-319-10592-5. DOI: 10.1007/978-3-319-10593-2_17.
- [38] H. Zhao, H. Yang, H. Su, and S. Zheng, “Natural image deblurring based on ringing artifacts removal via knowledge-driven gradient distribution priors”, *IEEE Access*, vol. PP, pp. 1–1, Jul. 2020. DOI: 10.1109/ACCESS.2020.3007972.

- [39] T. Kudo, T. Fujisawa, T. Yamaguchi, and M. Ikehara, “Non-blind image deconvolution based on “ringing” removal using convolutional neural network”, *Electronic Imaging*, vol. 2020, pp. 181–1, Jan. 2020. DOI: 10.2352/ISSN.2470-1173.2020.10.IPAS-180.
- [40] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, “Image quality assessment: from error visibility to structural similarity”, *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. DOI: 10.1109/TIP.2003.819861.
- [41] Y. Liu, J. Wang, S. Cho, A. Finkelstein, and S. Rusinkiewicz, “A no-reference metric for evaluating the quality of motion deblurring”, *ACM Transactions on Graphics (TOG)*, vol. 32, pp. 1–12, 2013.
- [42] R. Franzen. (). “Kodak kodak lossless true color image suite”. Accessed: 2021-70-10.
- [43] W. H. Richardson, “Bayesian-based iterative method of image restoration*”, *Journal of the Optical Society of America*, vol. 62, no. 1, p. 55, 1972. DOI: 10.1364/josa.62.000055. [Online]. Available: <https://app.dimensions.ai/details/publication/pub.1065152408>.