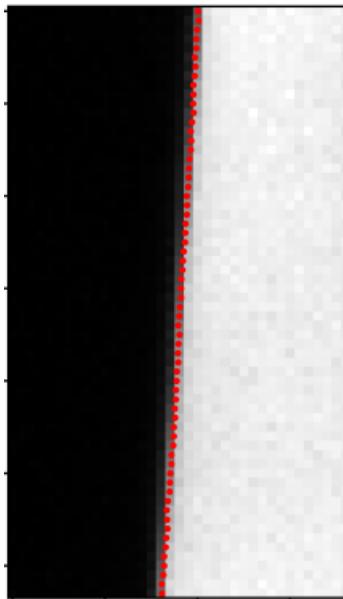


# Uncertainty Quantification In Slanted-Edge SFR



## Internship Report

Prasannavenkatesh Balaji

Student matriculation number: 10043034

31.05.2023

Supervisor(s):

Dominik Werner Wolf (M.Sc.)  
Dr. Jorge Humberto Urrea Quintero

Examiner(s):

Prof. Dr.-Ing. Udo Nackenhorst

# Plagiarism Declaration

I declare that this work is my own and I have not used any external material other than the provided sources with the appropriate citations. Furthermore, this work, or parts thereof, has not been previously submitted in the same or similar form to the examination office.

Hanover, 31.05.2023.....

Prasannavenkatesh.Balaji.....

Name, Family Name

## Abstract

Real-time imaging of the scene around the vehicle is a fundamental step in autonomous driving systems. This places high demands on the optical quality of the images because of the safety concerns revolving around highly autonomous driving. Eventually, appropriate metrics have to be developed to characterize the optical quality. One such metric is the modulation transfer function (MTF) defined in the frequency space and its non-harmonic version - the spatial frequency response. This work focuses on these metrics and the robustness of the methodology devised by ISO to calculate MTF. The sensitivity of the MTF to several parameters are studied in detail.

Firstly, the methodology is discussed in detail followed by listing out parameters that could potentially influence the MTF. This study is carried out in a two-fold manner - first through a monothetic analysis followed by studying correlated inputs. Two specific parameters from these parameter lists are analyzed extensively in the later sections as their influence is expected to be greater.

Following this analysis, the slant in the edge is modelled as a random variable input and its influence on the variance of the MTF is analyzed. This probabilistic approach is first validated using Monte Carlo simulations and then a surrogate model (polynomial chaos) is built to improve the estimations as well as reduce the computational resources required. Then, synthetic data generation is attempted at bridging the gap between insufficient data and data-driven uncertainty modelling. The Poisson-Gaussian model used for additive noise is discussed along with the challenges in dealing with synthetic data. Finally, a glimpse into the scope for further improvement in the ISO methodology is provided.

**Keywords:** Slanted-edge, ESF, PSF, MTF, ROI.

## Contents

<b>List of Figures</b>	<b>iii</b>
<b>List of Tables</b>	<b>iv</b>
<b>Glossary</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
<b>2 Problem Definition</b>	<b>3</b>
<b>3 Review of Literature</b>	<b>4</b>
<b>4 ISO:12233 - An Overview</b>	<b>7</b>
4.1 Code . . . . .	7
4.1.1 Pre-processing . . . . .	8
4.1.2 e-SFR . . . . .	9
<b>5 Analysis</b>	<b>11</b>
5.1 Sensitivity . . . . .	11
5.1.1 Monothetic analysis . . . . .	11
5.1.2 Correlated inputs - II order . . . . .	13
5.2 Demosaicing . . . . .	13
5.2.1 Quantum efficiency . . . . .	14
5.2.2 Spectrum of TV . . . . .	15
5.2.3 Results . . . . .	17
5.3 Derivative accuracy . . . . .	18
5.3.1 Finite Differences . . . . .	19
5.3.2 Higher order of truncation error . . . . .	19
5.3.3 Correction Factors . . . . .	21
5.3.4 Implementation . . . . .	23
5.3.5 Results . . . . .	24
<b>6 Input uncertainty</b>	<b>28</b>
6.1 Monte Carlo . . . . .	28
6.2 Polynomial Chaos . . . . .	31
<b>7 Synthetic Data</b>	<b>33</b>
<b>8 Future Work</b>	<b>35</b>

Contents

---

<b>9 Conclusion</b>	<b>36</b>
---------------------	-----------

## List of Figures

1.1	Autonomous driving process . . . . .	2
4.1	Slanted-edge SFR algorithm . . . . .	7
4.2	Pre-processing in e-SFR . . . . .	8
4.3	e-SFR algorithm . . . . .	9
5.1	Impact of findContour() . . . . .	12
5.2	Impact of findTarget() . . . . .	13
5.3	Quantum efficiency . . . . .	15
5.4	Spectrometer setup . . . . .	16
5.5	Spectrum of TV (source) . . . . .	16
5.6	Distribution of weighting factors . . . . .	17
5.7	MTF vs. weighting factors . . . . .	18
5.8	Spherical coordinate system . . . . .	24
5.9	$\phi$ , ID-080 - Left, Right . . . . .	25
5.10	$\phi$ , ID-080 - Top, Bottom . . . . .	25
5.11	$\theta$ , ID-080 - Left, Right . . . . .	26
5.12	$\theta$ , ID-080 - Top, Bottom . . . . .	26
6.1	Monte-Carlo illustration . . . . .	28
6.2	Monte-Carlo analysis . . . . .	29
6.3	Response functions . . . . .	30
6.4	Convergence of Monte Carlo . . . . .	31
6.5	Best approximation by second order PCE . . . . .	32
6.6	Poor approximation by seventh order PCE . . . . .	32
6.7	Convergence behaviour of $\mu$ and $\sigma$ . . . . .	33
7.1	Synthetic images . . . . .	34
7.2	MTF result of synthetic image . . . . .	34
7.3	External source - synthetic data . . . . .	35

## List of Tables

4.1	Parameters featuring in pre-processing . . . . .	9
4.2	Parameters featured in e-SFR . . . . .	10
5.1	Colour channel weighting factors . . . . .	17
5.2	Derivative filter stencil . . . . .	21

## Glossary

**ESF** Edge Spread Function. 4–6, 19, 21–23

**FoV** Field of View. 2

**MTF** Modulation Transfer Function. 1–4, 8, 11, 23

**PSF** Point Spread Function. 12, 19, 23, 27

**SFR** Spatial Frequency Response. 4

---

# 1 Introduction

In the past decade, research in autonomous driving has been consistently improving with multiple countries and automobile manufacturers competing with each other to develop not just a reliable but also a safe self-driving car. An important obstacle to cross before self-driving cars are available to the common man is to convince regulatory authorities like the UNECE, of the reliability of the automated features in a driving system. This is critical because on road, the performance of the driving system constantly impacts not just the occupants but everyone around the vehicle.

SAE (Society of Automotive Engineers) distinguishes an important difference between a vehicle being '*autonomous*' and '*automated*'. This is because when one wants to move from point A to B with a vehicle, a vehicle with a truly autonomous driving system could decide on its own whether it *wants* to move the occupants or not ! But, for the sake of convenience, both the terms are used interchangeably in this report. Thus, the SAE has defined a hierarchy highlighting different levels of autonomy of the driving system. The SAE in its taxonomy for driving automation systems defines six levels in its classification in which level-4 stands for 'high driving automation' [1]. The key difference between level-3 and level-4 is that in level-4, the system is equipped with two key abilities - one to be fully self-reliant and autonomous within a specified operational time domain and two, the ability to intervene in a situation that demands it [2].

## 1.1 Motivation

The fundamental motivation behind this work is to aid in the development of automobiles with level-4 autonomy, as defined by SAE. Volkswagen AG, like other major automobile manufacturers is on the path to develop cars with level-4 autonomy. And looking into the whole development process of such complex technology, it is evident that the project requires support from multiple research groups. It is known that the field of autonomous driving research heavily derives from the advances in deep learning. On one hand, Large language models (LLMs) like ChatGPT are being lauded for their excellent performance surpassing human ability by multiple magnitudes but on the other, autonomous driving systems' performance has not yet achieved that level of success because of the associated risks. Tasks generally aced by deep learning models are generally ones where the penalty of error is not life threatening but that is the case in autonomous cars. So, autonomous driving is a very unique application of deep learning where the performance accuracy needs to be as high as possible. While the penalty

associated with an error on-road could be death or injury to the occupants and pedestrians. While in principle, it is possible to integrate the optimal planning of routes, optimal speeds of driving etc. with the driving system itself leading to improved efficiency, hard constraints like pre-existing traffic rules (speed limits, overtaking rules etc.) need to be adhered to. This project pertains to an important and critical sub-branch of autonomous driving research - imaging by ADAS (Advanced Driver Assistance Systems) cameras. The working of any autonomous driving system starts with the imaging and subsequently an in-depth analysis of the "scene" surrounding the vehicle. This scene is obtained in the form of real-time imaging by the cameras mounted on the car, either inside the vehicle behind the windshield or sometimes outside the vehicle (depending on the design choice and other factors). The Fraunhofer Institute for Cognitive Systems (IKS) classifies three critical requirements or steps that needs to be met by every highly autonomous driving system (Level-3 and higher), which is illustrated in Fig. 1.1 [3].

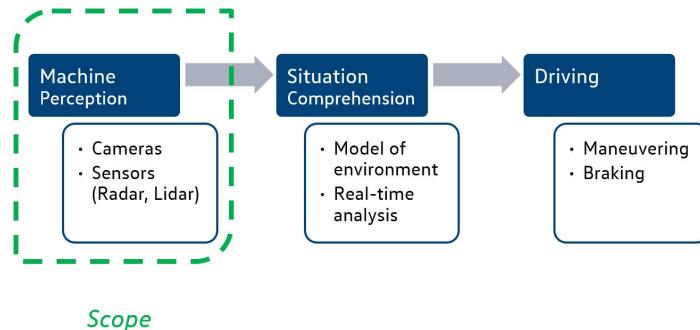


Figure 1.1: Three steps to self-driving

This indicates that the imaging process falls within the scope of the first step. Cameras mounted inside the car behind the windshield capture images, with their optical quality being influenced by the optical aberrations induced by the windshield [4]. The glass float and polyvinyl butyral (a thermoplastic resin) used in laminating glass layers are the main contributors to these aberrations. Thus, the windshield needs to be incorporated into the thorough analysis of how an image's quality is deteriorated. An important indicator of ADAS camera performance is the pixels-per-field-angle in the camera's field of view.FoV [5]. As cars are designed to operate at high speeds, they are required to detect small objects at faraway distances well enough. This results in a resolution requirement. Modeling the intensity incident on every picture element in the sensor array with a Poisson distribution, it is found that the standard deviation of MTF is roughly 2 % [6]. Therefore, uncertainties and systematic deviations higher than

---

this magnitude need to be investigated and much smaller contributions can be neglected due to this physical precision limitation.

## 2 Problem Definition

The tasks that I have focused on, through the course of my work as an intern can be summarized as

- Verifying if the slanted-edge SFR algorithm implemented at VW is fully compliant with ISO:12233. Validating if the results obtained are accurate within a confidence of roughly  $\pm 2\%$ .
- Robustness of the algorithm :
  - Enumerating a list of all possible parameters in the algorithm that could be influencing the quantity of interest i.e modulation transfer function (MTF).
  - Analyzing the sensitivity of the MTF to these parameters.
- Analyzing potential uncertainty in input variables through Monte Carlo simulation and other surrogate modeling approaches.
- Experimenting with alternate approaches aimed at improving the slanted-edge SFR algorithm.

---

### 3 Review of Literature

The review of literature for this work has been relatively straightforward because of one major reason being the ISO. The ISO (International Organization for Standardization) set up a technical committee (TC-42) dedicated towards formulating standards, definitions and methodologies for electronic and still picture imaging. This committee, since its inception in 1947, has been publishing test charts which imaging equipment manufacturers have used to test, calibrate and benchmark their products. The committee, which met in 1991, proposed to develop a standard for measurement methodologies related to spatial resolution. This paved the way to the publication of ISO 12233 in 2000 for the first time [7].

In earlier times, to compare the performance of two lenses or cameras, limiting resolution metrics were used. This enabled one to quantify spatial resolution but limited to one value per direction - either in Cartesian space (vertical and horizontal) or in polar space (radial and tangential). But it was later found that these metrics were misleading and were not congruent with perceived sharpness in the target. This led to the widespread adoption of metrics based on the modulation transfer function (MTF) [8]. The first version of a test chart that could be used to measure modulation was a sheet of film with multiple spatial frequencies (measured in cycles/mm). For the measurement, a microdensitometer was used to measure the modulation of every spatial resolution printed on the film [9]. As one of the pioneers in film and digital photography back then, Kodak created such a test chart to compare their cameras' performance. This is the reason behind the ISO12233 adopting the modulation transfer function (also known as optical transfer function) as a metric to quantify spatial resolution.

In the seminal work Reichenback et al. (1991) found that a single slightly slanted edge could be used as a robust target to measure the MTF [10]. This is because the slight tilt in the edge gave rise to the possibility of oversampling the ESF which greatly reduced bias in the measurement. This was recommended to the ISO and subsequently the ISO committee also adopted slanted edge SFR (spatial frequency response) as the primary measurement methodology.

This work is based on the 2017 version of ISO12233 as until recently, it was the latest version [11]. As the ISO committees meet every five years and discuss the possibility of potential updates to their standards based on latest advances in research, ISO12233 has also been subjected to numerous updates and revisions over the years with the latest one being in 2023. Dr. Peter D. Burns, one of the influential researchers in this domain and also a key contributor to the ISO's

---

digital photography team recently published a few possible updates to the 2017 version that could improve the performance of the slanted edge algorithm [12] and consequently, the standard has been revised in 2023. In addition to his publications, his open-source SFR analysis code written on Matlab *sfrmat* has been an industry benchmark and reference[13]. Two important revisions to the 2017 version present in the 2023 version is that the linear fit has been improved to a 5<sup>th</sup> order polynomial used to fit the slanted edge and in place of the Hamming filter used as the apodization window, a special case of Tukey window (with  $\alpha=1$ ) has been used.

The work by J.K.M Roland also brought to focus two key points - the sensitivity of MTF measurements and the inherent uncertainty in certain inputs that demanded them to be modeled as random variables [14]. This work led to the notion that the edge angle estimation process could have a significant impact on MTF. This work provided a basis that one could also work towards generating synthetic data to compensate for the inadequate real data available to us at the moment.

Synthetic data generation to substitute for real imaging data is tricky because we have not yet fully understood how exactly do several factors influence the spatial frequency response of cameras. The windshield on its own needs to be considered as a complex structure because of the multiple layers of material present and the lamination that binds together the layers. But the work by Carlson et al. (2018) on data augmentation techniques helped in understanding how different kinds of noises could be modelled and added to a synthetic image, and in what way one could parameterise these additive noise components [15]. This was the basis for my work on generating synthetic data. Another independent researcher's work which was insightful in understanding the role of the oversampling process and its influence on processing the ESF vector was Frans van den Bergh and his open-source implementation of the slanted-edge MTF methodology *MTF mapper* (refer to [16] and [17]). Moreover, his documentation of several topics related to the methodology was instrumental in understanding them in depth. Particularly, his focused analysis of certain critical edge angles and how such angles behaved in the edge-angle estimation process was referred to in my analyses ([18] and [16]). This understanding of the need and influence of oversampling led to some review of alternative approaches in oversampling, such as Dr.Kenichiro Masaoka's work on parameterizing the oversampling factor. Dr. Masaoka successfully attempted removing the independence of the oversampling factor (also referred to as binning ratio) from the algorithm [19]. The oversampling ratio (also referred to as the binning factor) was indirectly defined as a function of the initial angle estimate itself such that the most optimal oversampling factor for

---

a particular edge was chosen. This resulted in an improved ESF representation for different edges. The *OMNI-Sine* method developed by Dr. Masaoka projects every row's ESF onto one and groups the pixel intensity values based on a combination of bins of width  $\sin\theta$  and  $\cos\theta$  where  $\theta$  is the angle of the edge with respect to the vertical or horizontal depending on the orientation of the edge (discussed in detail in [19]).

The popular textbook by Smith et al. (2013) provided a good overview of multiple uncertainty quantification approaches and provided insight that goes into choosing the appropriate surrogate model for any particular problem (considering the dimensionality, complexity etc) [20]. The exhaustive study by Dr. Karniadakis and group on different families of polynomials used to solve differential problems and their error convergence while solving diverse stochastic differential equations greatly helped in understanding how surrogate models reduced the dimensionality and helped speed up computation [21]. It also aided in understanding the pseudo-spectral approach in solving polynomial chaos using Galerkin projection.

## 4 ISO:12233 - An Overview

In this section, a brief overview of the universal standard ISO:12233 is covered, with particular emphasis on the slanted edge SFR algorithm [11]. The methodology as defined by the standard is compared in depth with the in-house implementation at VW. ISO:12233 defines the complete slanted-edge SFR methodology through a sequential flowchart as illustrated by Fig. 4.1.

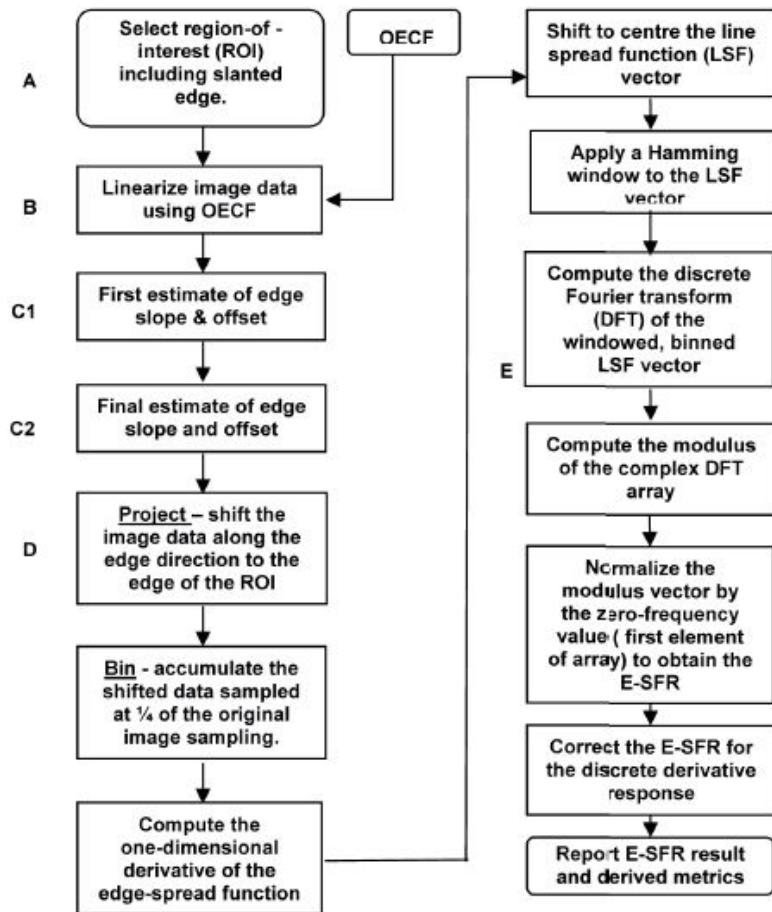


Figure 4.1: Slanted-edge SFR algorithm, as defined in [11]

### 4.1 Code

The process (see Fig. 4.1) has been decomposed into two sub-processes - for better understanding and comprehension. It is split into pre-processing and then, the actual e-SFR algorithm.

### 4.1.1 Pre-processing

In this section, we look into the functions that act on the raw image (imaged in the light tunnel) and in return produce the sub-images that are ready for the SFR algorithm. The notion is that these functions and their corresponding parameters do not contribute to or influence the quantity of interest (MTF) but they are critical to the subsequent steps in the methodology. The pre-processing group can be visualized better with Fig. 4.2. There are four functions constituting this group and are codified from 1-4.

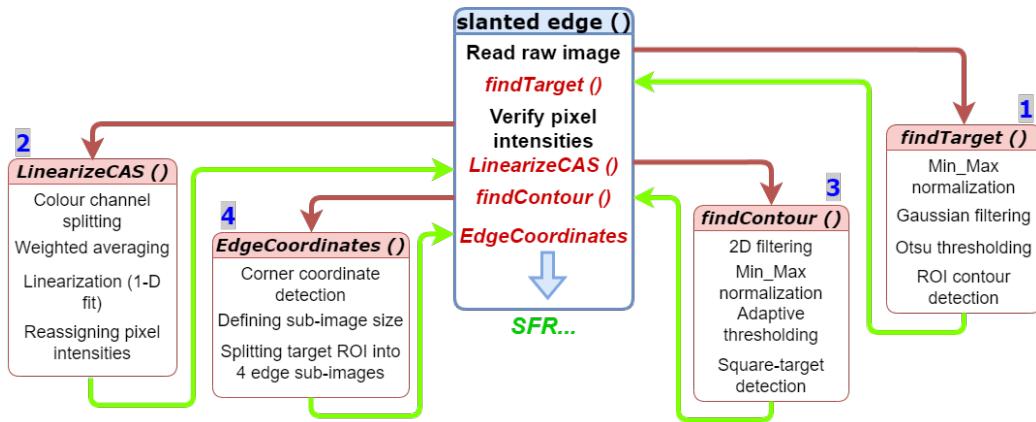


Figure 4.2: Pre-processing of the raw target image

Although there have been some arbitrary choices behind the parameters involved in these steps, they are still fully compliant with the standard and theory. These operations on the image do not artificially manipulate the pixel intensity values, except the linearization of pixel intensities which is also implemented as directed in the standard [11].

All parameters that systematically feature in pre-processing are tabulated in Tab. 4.1. These parameters are grouped according to their respective functions with information on what the value passed into the parameter signifies.

CODE	1		2		3		4	
NAME	findTarget		LinearizeCAS		findContour		EdgeCoordinates	
	Operation	Value	Operation	Value	Operation	Value	Operation	Value
1	Min_Max Normalization	Min = 0 Max = 255	Channel Weight	0.25	2D Filtering	(3, 3) Window	Aspect Ratio	0.3
2	Gaussian Filtering	(5, 5) Window			Min_Max Normalization	Min = 0 Max = 255	Offset	5 px
3	Otsu Thresholding	Min = 0 Max = 255			Adaptive Thresholding	Size = 11 Const = 2		
4	Contour W/H	Min = 0.8 Max = 1.2			A/R Tolerance	0.1		
5	Contour Area	12000 sq. px			Contour Area	25 sq. px		

Table 4.1: Parameters featuring in pre-processing

#### 4.1.2 e-SFR

This section covers the functions that work on the remaining steps in the algorithm i.e starting with the sub-images containing a single slanted edge and resulting in a MTF distribution for every edge.

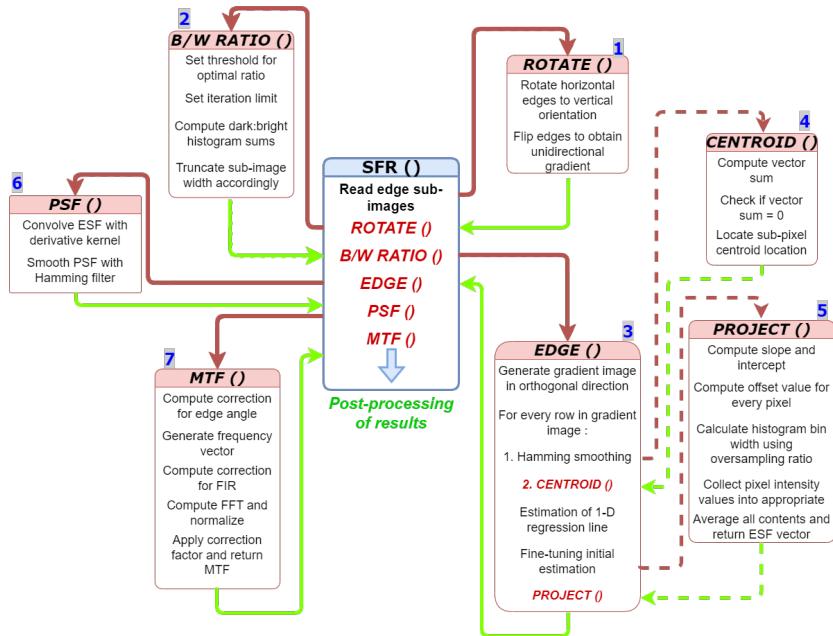


Figure 4.3: The actual e-SFR algorithm on the edge

The parameters featuring in this sub-process are grouped together depending on which function they belong to and are tabulated in Tab. 4.2.

## 4.1 Code

---

NAME	2		3		4		5		6		7	
	B/W Ratio		EDGE		CENTROID		PROJECT		PSF		MTF	
	Operation	Value	Operation	Value	Operation	Value	Operation	Value	Operation	Value	Operation	Value
1	Optimal-Ratio Threshold	0.03	Smoothing Window	Hamming (or) Tukey	Minimum Threshold	0.0001	Binning Factor	4	Derivative Trunc. Order	2	Corr. Factor Threshold	20
2	Iteration Limit	200	Window Size	ESF					Convolution Mode	'Same'	Savitzky Golay Filter Size	Length = 5 Order = 2
3									Smoothing Window	Hamming (or) Tukey	Polynomial Approximation	Cubic Spline

Table 4.2: Parameters featured in the actual e-SFR algorithm from Fig. 4.3

The most influential function of the whole analysis is the sequence of steps in the function numbered 3 in Fig. 4.3. This function is responsible for re-orienting the edge to a normal position, thereby computing an offset value for every pixel coordinate  $(p,r)$  [11]. Then all the row-edge spread functions are super-imposed on one another and binned together according to the binning factor into one resultant ESF vector which is representative of all the information present in the edge ROI. Hence, it is considered to be a critical set of steps.

The post-processing part of the analysis entails the plotting and saving of data as well as the calculation of certain metrics. Two important metrics that are prevalent in usage are  $\frac{Ny}{2}$  MTF and MTF50. They are derived from the spatial frequency response and are used extensively in literature to compare two edges and their MTF behaviours.

1.  $\frac{Ny}{2}$  MTF - The MTF value at half-Nyquist frequency. Usually expressed in percentage.
2. MTF50 - The spatial frequency at which 50 % MTF is attained.

---

## 5 Analysis

Understanding the working of the e-SFR methodology was of primary importance and this was covered in detail in Sec. 4. Consequently, in this section we focus on another domain - verifying and validating if the methodology is robust enough against variability in the parameters involved in the methodology and also variability in input variables. The robustness of the quantity of interest MTF is critical and needs to be ensured with satisfactory confidence because the imaging process is fundamental to the autonomous driving system and the MTF stands as the metric upon which several design and developmental decisions are built on.

The complete analysis is split into two parts -

1. Analyzing the sensitivity of MTF with respect to the parameters involved in the functions in Sec. 4.1.1 and Sec. 4.1.2. This is further decomposed into two sub-analyses.
  - (a) Monothetic analysis - One-Factor-At-a-Time (OFAT) studies focusing on the sensitivity of MTF due to variation in just one parameter at a time.
  - (b) Correlated inputs - Studies focusing on correlations of the second order ; Studying the sensitivity of MTF to varying two parameters in the algorithm simultaneously.
2. Modeling uncertain inputs as random variables and studying the uncertainty in the output (MTF).

### 5.1 Sensitivity

#### 5.1.1 Monothetic analysis

The parameters in their respective functions were varied first by defining a set of 'appropriate' values and sequentially iterating through the series. This is due to the discrete nature of certain parameters, for example, the window size of a filter - it can only be an integer value. And to check for repeatability and verify the initial results, parameters were again chosen at random order using a pseudo-random generator [22] from the series and applied. Then they were sorted removing repetitions and validated with respect to the earlier results. An instance of an 'appropriate' value would be to not use a Gaussian filter with a filter size of  $60 \times 60$  pixels because they would cover a majority of the image and thus there would be no scope left for the convolution operation, or that a standard deviation of the order of 100 would render the whole target image

irreversibly blurred.

The behaviour of one such parameter from this study is illustrated below in Fig. 5.1 . This parameter features in the third function *findContour()* (Code=3 in Fig. 4.2). This function operates on the linearized target ROI and removes noise from the image. It is aimed at obtaining an accurately positioned image of the target.

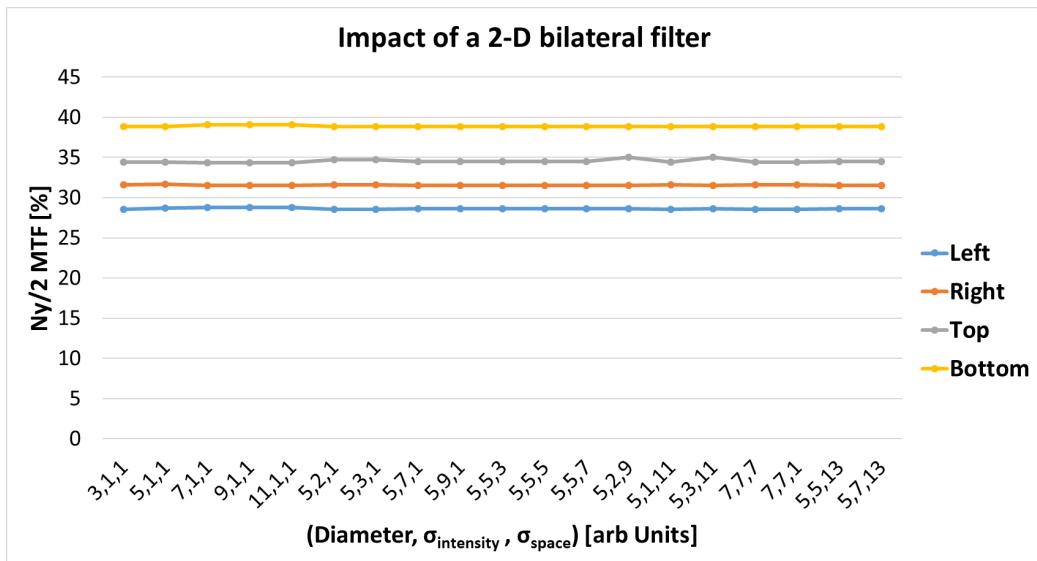


Figure 5.1: Impact of 2-D Filtering from function 3 in 4.2

Two parameters that demanded to be analyzed with a different approach were

- the weighting factor  $w_c$  used for the colour channels (Code=2 in Fig. 4.2). This is because the factors belonged to continuous distributions from which they could be potentially sampled from.
- the order of truncation error for the PSF (Code=6 in Fig. 4.3). This is because improving the order of truncation error had many implications - it meant that a new derivative filter with more neighborhood information had to be constructed and consequently, its correction factor was also required to be derived.

For this reason, these two parameters and their impact on the MTF is studied separately in the next section.

It is observed from these studies that except for the two aforementioned parameters, the independent impact of none of these parameters exceed  $\pm 0.25\%$ . These studies were performed on two images chosen arbitrarily from two different measurement campaigns using cameras, with internal ID: 080 and 147.

### 5.1.2 Correlated inputs - II order

Certain tasks in general image processing are always performed as a sequence so as to achieve desired results. One such example is the enhancement of critical features (for example, an edge) by Gaussian smoothing and thresholding. Generally, Gaussian filtering is performed as a form of low pass filtering to remove unwanted noise followed immediately by a form of binary thresholding. Different levels of Gaussian filtering were used along with the resultant Otsu thresholds with the parameters involved being correlated. It is to be noted that the Otsu threshold is an automatic binary thresholding technique separating the foreground and background by minimizing intra-class pixel intensity variance. Although the correlations are not quantified in detail, their combined impact on the quantity of interest is studied. Fig. 5.2 below illustrates this impact.

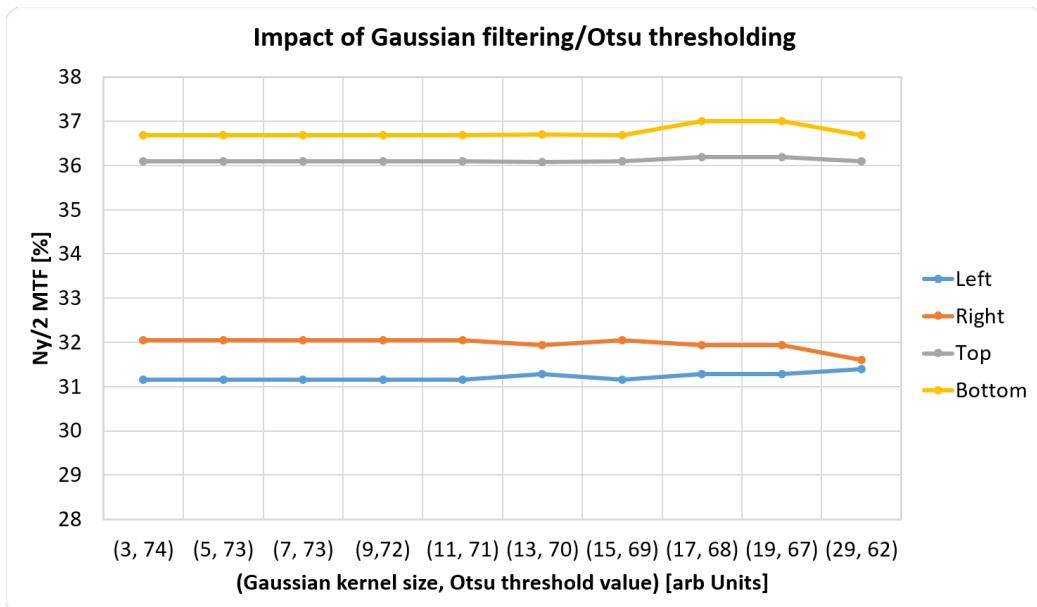


Figure 5.2: Impact of a combination of 2 parameters from function 1 in 4.2

It is observed from these studies that the impact of correlated inputs (upto the second order) of any of these parameters do not exceed  $\pm 0.25\%$ . These studies were also performed on the same two images chosen arbitrarily as earlier.

## 5.2 Demosaicing

According to the ISO standard in [11], the intensity values recorded by the CMOS sensor element in the raw image have to be linearized using the opto-electronic conversion function [23]. In addition to this, every channel in the image is

assigned a weighting factor and the resultant photopic intensity at every pixel position is calculated as a linear combination of the channel weights multiplied by the function.

As sufficient data about the weighting factors was not available in the specification document of the sensor used, all the weighting factors were assigned an equal value. The CMOS sensor used in the process is an advanced automotive sensor that is optimized for working in low-light and high dynamic range environments. The sensor array has the structure of RCCB - one red, one blue and two clear channels respectively. As all the channels were initially assumed to be contributing equally, they were assigned a weight of  $w_{ch} = 0.25$ . This was not an accurate way to calculate the weighted sum as the clear channels should have had a weight proportional to their larger contribution. The contribution of a channel of sensor elements can be expressed as :

$$\int_0^{\infty} QE(\lambda)S(\lambda)d\lambda, \quad (5.1)$$

where  $QE(\lambda)$  and  $S(\lambda)$  denote the quantum efficiency and spectrum of energy source respectively.

### 5.2.1 Quantum efficiency

In order to find the exact contribution of the channels, both the source (the television displaying the target) and the sensor capturing the light needs to be analyzed. An important metric used to measure sensor performance is the quantum efficiency (QE). It is defined as the number of electron-hole pairs created in a pixel by one incident photon, given by

$$QE_{sensor}(\lambda) = \frac{N_e}{N_\nu}, \quad (5.2)$$

where  $N_e$  and  $N_\nu$  denote the number of electrons produced and the number of photons absorbed, respectively [24].

The quantum efficiency spectrum (as it is a function of the incident light's wavelength) is analogous to the IPCE ratio of the sensor (Incident Photon to Converted Electron ratio). This data was available only in the form of a plot in the sensor's datasheet due to proprietary reasons. Thus, a plot digitizer tool was used to mask the three curves (for red, blue and clear channels) and then extract the data behind these curves by interpolating with sufficient points [25].

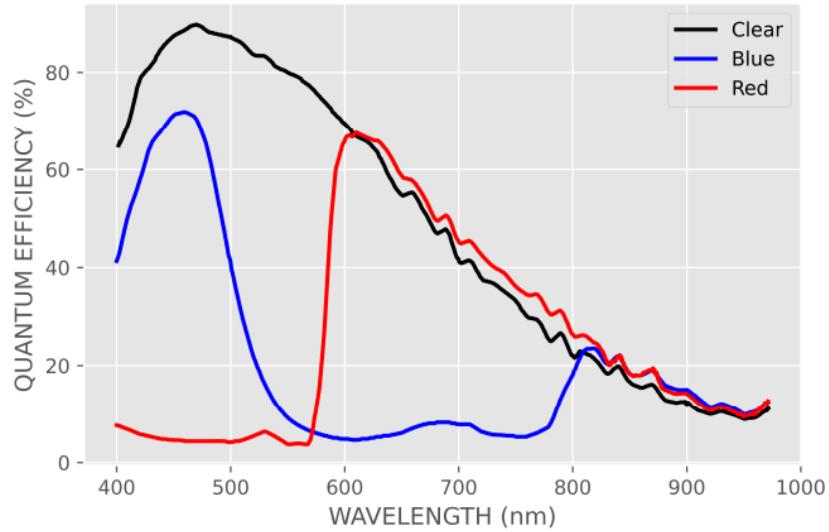


Figure 5.3: Quantum efficiency curves of the sensor used

The curves in Fig. 5.3 clearly indicate that the quantum efficiency is dependent on the wavelength and thus, the weighting factors should be proportionally different too. This is discussed next in Sec. 5.2.3.

### 5.2.2 Spectrum of TV

A spectrometer was used to record the spectrum of light emitted by the television, used as the source to display the square target. This is to calculate the energy emitted by the source for different wavelengths. Four points were used to measure the spectrum - two inside the dark target and two points on the left and right bright edges.

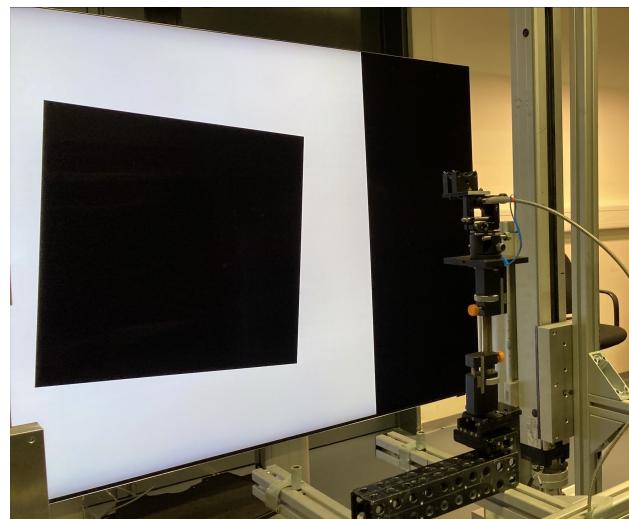


Figure 5.4: Setup to measure the spectrum of the TV

The mean of the intensities of the two bright points from the left and right side of the target were normalized between 0 and 100 to be used in calculating the integral. For equidistant sampling, the measurements were interpolated. 1000 values equally spaced between the minimum and maximum wavelength were considered and fit to the spectrum curve approximated using a cubic spline.

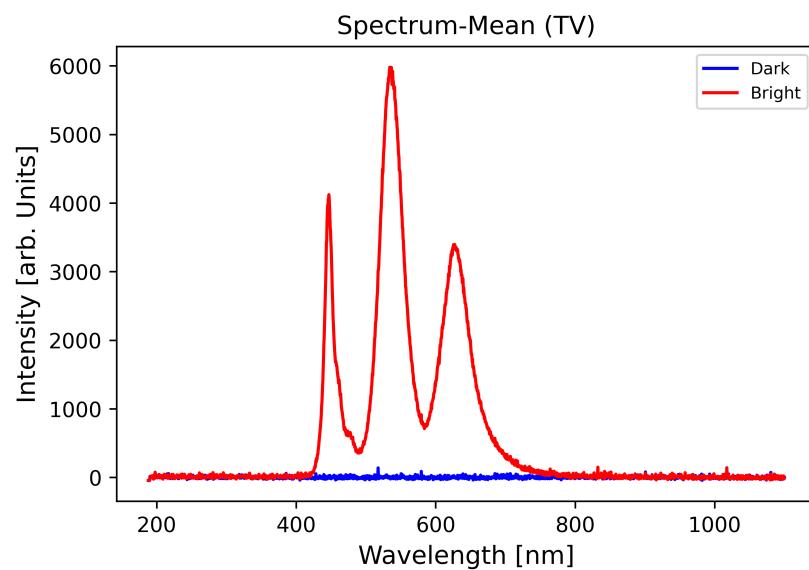


Figure 5.5: Spectrum of TV (source)

### 5.2.3 Results

After the quantum efficiency and energy spectrum is obtained, their product is calculated and numerically integrated using the Simpson's rule with an interval width of 0.5. Thus, we obtain the exact contribution of the two clear, one blue and red channels respectively. It is to be noted that the constraint of  $2 \times w_C + w_R + w_B = 1$  is satisfied for every set of samples.

Channel	Weighting factor
Clear	0.359
Blue	0.159
Red	0.124

Table 5.1: Colour channel weighting factors

To study the impact of the weighting factors on the MTF, normal distributions were generated with the computed factors being the mean ( $\mu = w_{ch}$ ) and standard deviation  $\sigma = 0.05$ , which is an estimate of maximum error obtained from professional expertise. The normal distributions were generated in a truncated fashion between  $10^{-3}$  and 1 to not consider null values as weights. The truncated normal distributions are depicted in Fig. 5.6.

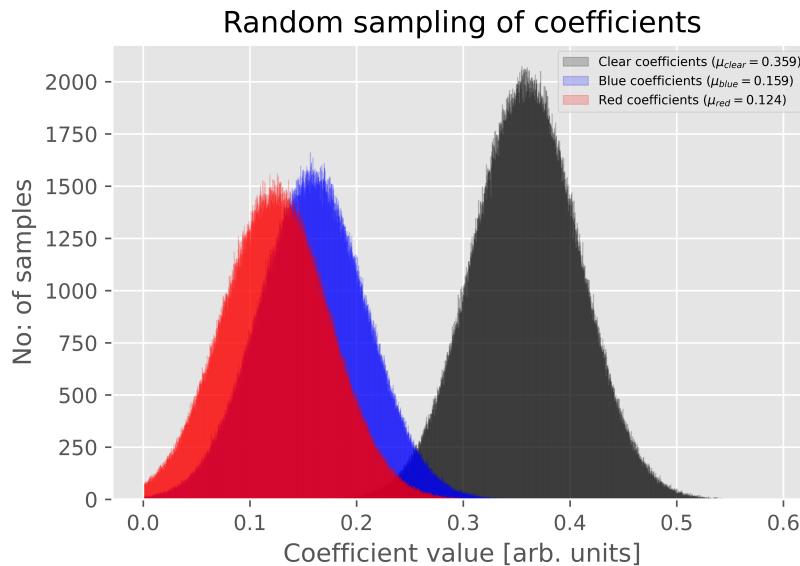


Figure 5.6: Distribution of weighting factors

Measurement 16 from the 26.04.2023 was chosen at random and analyzed with 10000 sets of weights sampled from their respective distributions based on one criterion - the sum of the sampled weights were controlled within  $1 \pm 0.005$ . The results are plotted in Fig. 5.7 in the form of histograms representing the percentage-deviation from the true  $\frac{N_y}{2} MTF$  value that was obtained when it was assumed that all channels were contributing equally i.e. all  $w_{ch} = 0.25$ .

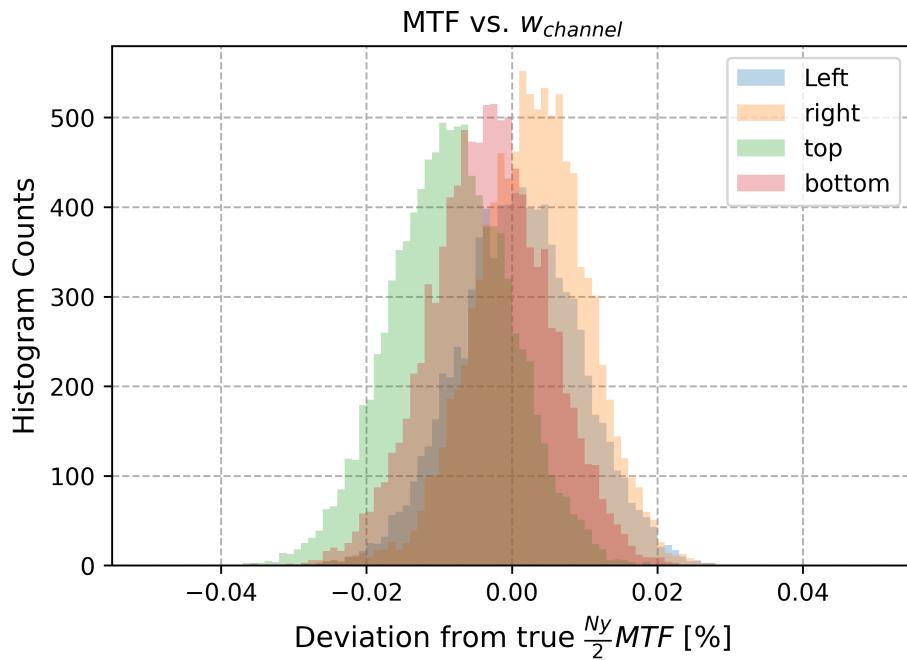


Figure 5.7: MTF vs. weighting factors

5.7 indicates that the combination of weights have negligible impact on the MTF. The same behaviour was observed repeatedly on multiple images with the deviation not exceeding  $\pm 0.1\%$ .

### 5.3 Derivative accuracy

Any function  $f$  could be expressed as a summation of an infinite power series given by

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(x)}{n!} (x - h)^n. \quad (5.3)$$

This, expanded is the well known Taylor series expansion of a function  $f(x)$  evaluated at the point  $x = h$ . This point  $h$ , being arbitrary provides immense flexibility to this approach as any real or complex function could be expressed

as a sum evaluated at any point as long as  $f$  is infinitely differentiable at that point.

### 5.3.1 Finite Differences

The derivative of a function  $f$  at the point  $x$  could be expressed as a finite difference quotient given by the limit  $f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h)-f(x)}{h}$ . The finite difference approach is a numerical differentiation method to approximate the derivative of a function when the analytical forms of the function and its derivative are unknown. The accuracy of the approximation, for time-independent problems such as ours, depends solely on the order of truncation (of the Taylor expansion). The first-order accurate forward and backward difference approximations of the first derivative  $f'(x)$  are given by

$$\begin{aligned} & \frac{f(x + \Delta x) - f(x)}{\Delta x}, \\ & \frac{f(x) - f(x - \Delta x)}{\Delta x}. \end{aligned} \tag{5.4}$$

depending on whether  $h$  is positive ( $h = \Delta x$ ) or negative ( $h = -\Delta x$ ) respectively. They are referred to as being first-order accurate because the truncated terms from the respective Taylor series expansions, when grouped together, start from the order  $O(\Delta x^1)$ .

### 5.3.2 Higher order of truncation error

A linear combination of Taylor Series expansions at different neighboring points around  $x$  could be used to compute derivatives with a higher order of accuracy i.e. higher order of the truncation error terms. The intuition is that every neighbor of  $x$  contains a specific non-zero contribution to  $f'(x)$  that could be considered in the summation to improve the accuracy. In our application, this would mean including more pixels from the neighborhood into the derivative-approximating process and thus, making the derivative less localized. For instance, subtracting the backward difference quotient from the forward difference quotient given in 5.4 leaves us with the second order accurate ( $O(\Delta x^2)$ ) central difference quotient  $\frac{f(x+\Delta x)-f(x-\Delta x)}{2\Delta x}$  which is also symmetrical about  $x$  and takes the two nearest neighboring terms on both sides  $x + \Delta x$  and  $x - \Delta x$ .

Such 'stencils' which are symmetrical about the centre i.e. about  $x = 0$  are the only meaningful approaches for us as the ESF represents an edge with a small ( $\approx 5^\circ$ ) edge angle and the PSF would not be accurate if the stencil favoured one direction over the other for neighborhood information. The tedious process of

finding the right linear combination could be simplified by the following approach. Taking the case of a fourth order approximation, we would require two neighbors from either sides of  $x$  for the approximation of  $f'(x)$  at  $x = x$ . This could be written in the form of a weighted sum for convenience as follows. Following previously stated intuition, the weights would help in finding the stencil-points' contribution to  $f'(x)$  :

$$\frac{d^4 f}{dx^4} \approx af(x - 2h) + bf(x - h) + cf(x) + df(x + h) + ef(x + 2h). \quad (5.5)$$

The term  $cf(x)$  at  $x$  does not require an expansion because it is about this point the expansion originates and eventually, while solving for the linear system of equations, the  $c$  term gets cancelled. Expanding each term in the neighborhood of  $x$  in 5.5 about  $x$ , with their respective coefficients multiplied, results in

$$af(x - 2\Delta x) = a.f(x) - 2a\Delta x f'(x) + 4a\Delta x^2 \frac{f''(x)}{2!} - 8a\Delta x^3 \frac{f'''(x)}{3!} + 16a\Delta x^4 \frac{f^{(4)}(x)}{4!} - \dots \quad (5.6)$$

$$bf(x - \Delta x) = b.f(x) - b\Delta x f'(x) + b\Delta x^2 \frac{f''(x)}{2!} - b\Delta x^3 \frac{f'''(x)}{3!} + b\Delta x^4 \frac{f^{(4)}(x)}{4!} - \dots \quad (5.7)$$

$$df(x + \Delta x) = d.f(x) + d\Delta x f'(x) + d\Delta x^2 \frac{f''(x)}{2!} + d\Delta x^3 \frac{f'''(x)}{3!} + d\Delta x^4 \frac{f^{(4)}(x)}{4!} + \dots \quad (5.8)$$

$$ef(x + 2\Delta x) = e.f(x) + 2e\Delta x f'(x) + 4e\Delta x^2 \frac{f''(x)}{2!} + 8e\Delta x^3 \frac{f'''(x)}{3!} + 16e\Delta x^4 \frac{f^{(4)}(x)}{4!} + \dots \quad (5.9)$$

Now, a system of linear equations is solved for the unique set of coefficient-values which when multiplied to their respective expansions leads to the disappearing of all higher order terms and leaves us with just  $f'(x)$ . Such a system could be concisely expressed with just the coefficients as :

$$\begin{aligned} a + b + c + d + e &= 0 && ; \text{ for } f(x) \\ -2a - b + d + 2e &= 1 && ; \text{ for } f'(x) \\ 4a + b + d + 4e &= 0 && ; \text{ for } f''(x) \\ -8a - b + d + 8e &= 0 && ; \text{ for } f'''(x) \\ 16a + b + d + 16e &= 0 && ; \text{ for } f^{(4)}(x). \end{aligned} \quad (5.10)$$

Solving this system of linear equations yields us with the coefficients

a	b	c	d	e
+1/12	-2/3	0	2/3	-1/12

Table 5.2: Coefficients for the 4<sup>th</sup>-order accurate derivative filter

It is to be noted that this is an unique solution to this system and the coefficients, when substituted into the constraint for  $f'(x)$  prove that there is no first-order derivative information lost through the process.

### 5.3.3 Correction Factors

Before applying the derivative filter with the coefficients from the previous step, one needs to investigate the bias in the Fourier transform due to the discrete derivative. As discussed in the previous section, the discrete derivative is only an approximation and this is bound to directly influence the frequency spectrum. From [11], we can write the frequency response of the edge spread function (ESF vector) as follows.

$$SFR_{edge}(k) = D(k) \cdot \left| \frac{\sum_{j=1}^N PSF_W(j)e^{-i2\pi kj/N}}{\sum_{j=1}^N PSF_W(j)} \right|, \quad (5.11)$$

where

1.  $k$  is the spatial frequency index ;  $k = 0, 1, 2, 3 \dots \frac{N}{2}$  for  $k$  being even or  $k = 0, 1, 2, 3 \dots \frac{N+1}{2}$  for  $k$  being odd.
2.  $PSF_W$  is the windowed (with Hamming filter) and oversampled point spread function of the edge.
3.  $D(k)$  is the correction factor introduced due to the discrete derivative.

The following derivation attempts to build the correction factor published in the ISO from the fundamentals. Given that the ESF is available in a vector form, the finite difference operator applied to the ESF results in an approximation to the PSF in (5.11),

$$P\hat{SF}(x) = \frac{ESF(x+h) - ESF(x-h)}{2 \cdot h}. \quad (5.12)$$

### 5.3 Derivative accuracy

---

Here  $h$  is by default the unit of discretization in the oversampled space. In the pixel space, this would be denoted as  $\frac{1}{N_{bins}}$  units, where  $N_{bins}$  is the binning factor.

For convenience, if one denotes  $ESF(x)$  as  $f(x)$ , then for a spatial shift of  $\pm h$ , The Fourier transform can be written as :

$$FT(f(x \pm a)) = e^{\pm ih\omega} FT(f(x)). \quad (5.13)$$

Plugging into (5.13) expression from (5.12) and using Euler's representation of complex numbers, we obtain :

$$\begin{aligned} FT(P\hat{SF}(x)) &= FT\left[\frac{f(x+h) - f(x-h)}{2 \cdot h}\right] \\ \iff FT(P\hat{SF}(x)) &= \frac{FT(f(x+h)) - FT(f(x-h))}{2 \cdot h} \\ \iff FT(P\hat{SF}(x)) &= \frac{e^{+ih\omega} - e^{-ih\omega}}{2 \cdot h} \cdot FT(f(x)) \\ \iff FT(P\hat{SF}(x)) &= i \cdot \frac{e^{+ih\omega} - e^{-ih\omega}}{2i \cdot h} \cdot FT(f(x)) \\ \iff FT(P\hat{SF}(x)) &= i \cdot \frac{\sin(h\omega)}{h} \cdot FT(f(x)). \end{aligned} \quad (5.14)$$

The Fourier transform of the true analytical derivative of the ESF can be written as :

$$\begin{aligned} FT(PSF(x)) &= FT\left(\frac{d}{dx} \cdot ESF(x)\right) = FT\left(\frac{d}{dx} \cdot f(x)\right) \\ \iff FT(PSF(x)) &= i \cdot \omega \cdot FT(f(x)). \end{aligned} \quad (5.15)$$

The correction factor  $D$  from (5.11) is expressed as the ratio of equations (5.14) and (5.15) :

$$\begin{aligned} FT(PSF(x)) &= D \cdot FT(P\hat{SF}(x)) \\ \implies D &= \frac{i \cdot \omega \cdot \cancel{FT(f(x))}}{\cancel{i} \cdot \frac{\sin(h\omega)}{h} \cdot \cancel{FT(f(x))}} \\ \iff D &= \frac{h \cdot \omega}{\sin(h\omega)} = \frac{1}{\text{sinc}(h\omega)}. \end{aligned} \quad (5.16)$$

This approach to find  $D$  has been extended to the fourth-order accurate finite difference form, from (5.10).

$$\begin{aligned}
 FT(P\hat{S}F(x)) &= FT \left[ \frac{f(x-2h) - 8 \cdot f(x-h) + 8 \cdot f(x+h) - f(x+2h)}{12 \cdot h} \right] \\
 &= \frac{e^{-i2h\omega} - 8e^{-ih\omega} + 8e^{ih\omega} - e^{i2h\omega}}{12h} \cdot FT(f(x)) \\
 &= \left[ \frac{-i}{6h} \cdot \frac{e^{+i2h\omega} - e^{-i2h\omega}}{2 \cdot i} + \frac{8 \cdot i}{6 \cdot h} \cdot \frac{e^{+ih\omega} - e^{-ih\omega}}{2 \cdot i} \right] \cdot FT(f(x)) \\
 &= \left[ \frac{-i}{6h} \cdot \sin(2h\omega) + \frac{8 \cdot i}{6 \cdot h} \cdot \sin(h\omega) \right] \cdot FT(f(x)) \\
 \implies D &= \frac{\cancel{i} \cdot \omega \cdot \cancel{FT(f(x))}}{\cancel{\frac{i}{6h} \cdot [8\sin(h\omega) - \sin(2h\omega)] \cdot \cancel{FT(f(x))}}} \\
 \iff D &= \frac{6h\omega}{8\sin(h\omega) - \sin(2h\omega)}. \tag{5.17}
 \end{aligned}$$

Similarly, extending this approach to three points and four points on either side of  $x = 0$  i.e. a derivative with sixth and eighth order truncation error is presented in the following equation.

$$\begin{aligned}
 D &= \frac{30h\omega}{45\sin(h\omega) - 9\sin(2h\omega) + 3\sin(3h\omega)}; \\
 D &= \frac{420h\omega}{672\sin(h\omega) - 168\sin(2h\omega) + 32\sin(3h\omega) - 3\sin(4h\omega)}. \tag{5.18}
 \end{aligned}$$

#### 5.3.4 Implementation

In the existing version of code, *numpy's* 'convolve' function [22] was used to convolve a filter with the finite difference coefficient weights onto the ESF array to obtain the PSF. The existing version used a central difference approximation at all possible positions with the two end values of the array being manually assigned the closest 'valid' convolution value. Since this approach would have been tedious for higher order implementations, *scipy's* 'convolve1d' function [26] was adopted since it had a built-in mode to choose how the edges of the array would be computed. The higher order implementations required many neighboring points for which large systems of equations needed to be solved. Thus, a finite difference coefficient calculator [27] was used to compute the required coefficients and also to validate analytically derived values. The derivative filter were convolved in the PSF function and the correction factor was multiplied to the MTF vector inside the MTF function, as seen in 4.3

### 5.3.5 Results

In the light tunnel facility, the cameras are tested at various viewing positions in its field of view, with varying  $\phi$  (Azimuth angle) and  $\theta$  (Polar angle) respectively. An illustration of the  $\phi$  spanning the  $x - y$  plane and  $\theta$  spanning the  $x - z$  plane is in Fig. 5.8. According to Fig. 5.8, the optical axis in our measurements would be the x-axis, with the azimuth angle =  $\phi$  and polar angle =  $\frac{\pi}{2} - \theta$  respectively.

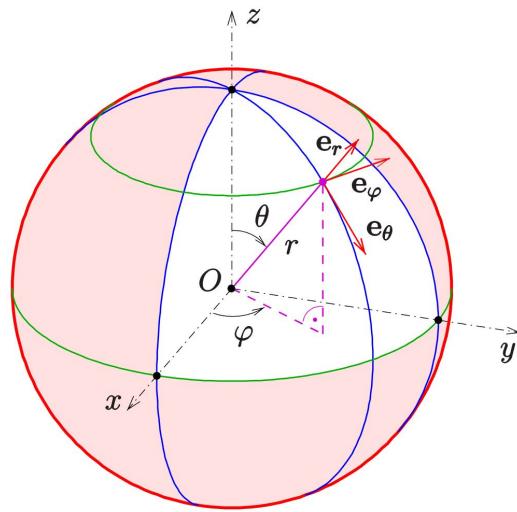


Figure 5.8:  $\phi$  and  $\theta$ , in the  $x - y$  and  $x - z$  planes respectively.

$\phi$  was varied between  $[-15^\circ, 15^\circ]$  approximately, with one measurement attempted at an interval of  $1^\circ$ . As the camera positions were manipulated manually, deviations from exact  $1^\circ$  measurements were expected and consequently noticed. Similarly,  $\theta$  was varied between  $[-7^\circ, 7^\circ]$ . The internal measurement campaign on 26.04.2023 was found to be yielding the best results with consistent camera performance from the three cameras used and thus their data is used as reference.

### 5.3 Derivative accuracy

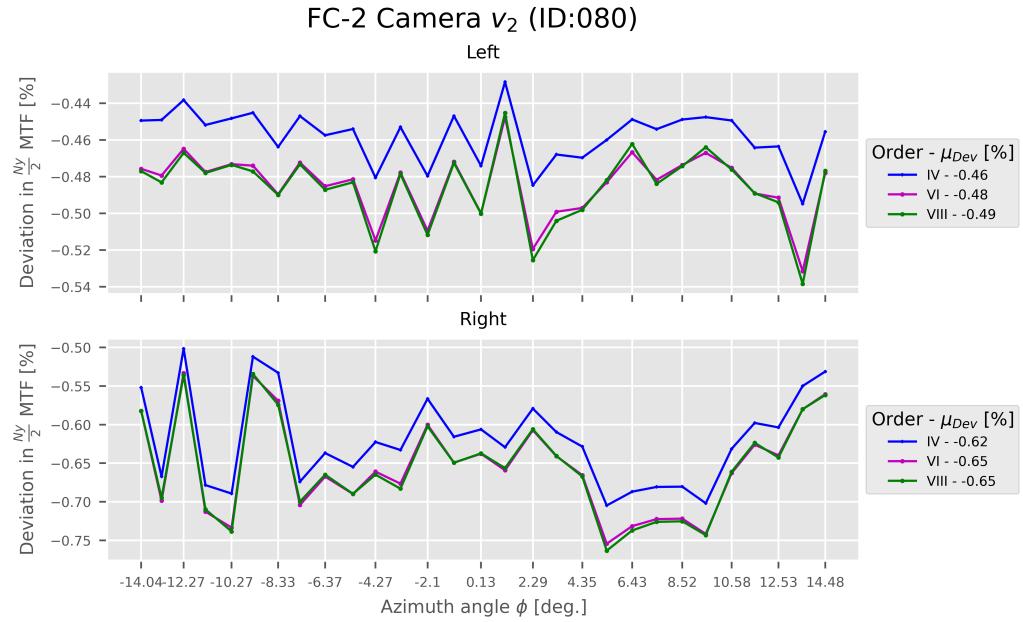


Figure 5.9:  $\frac{Ny}{2}$  MTF deviations vs.  $\phi$  for various orders.

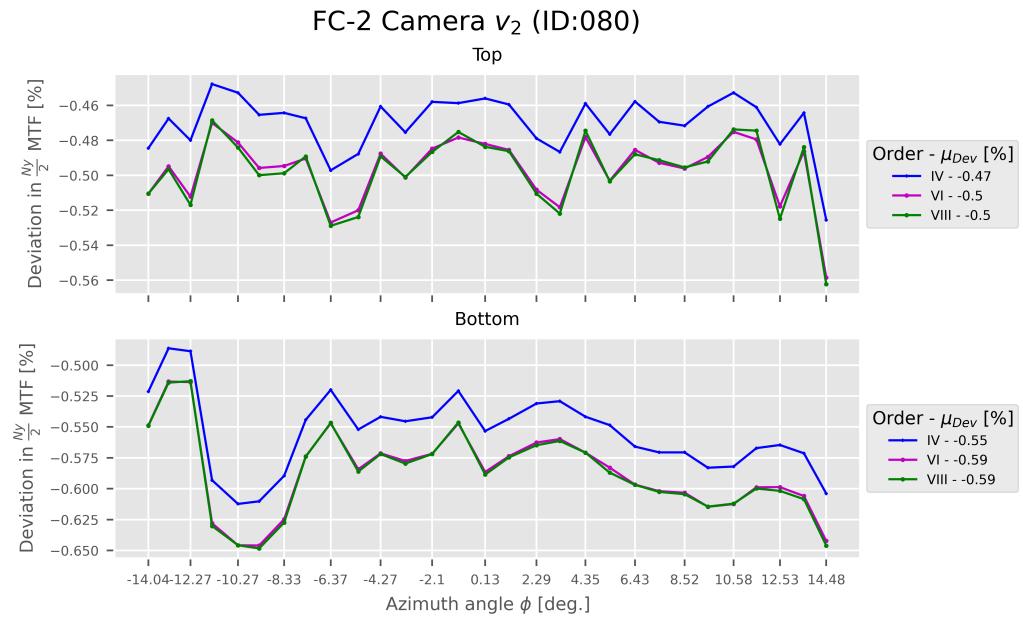


Figure 5.10:  $\frac{Ny}{2}$  MTF deviations vs.  $\phi$  for various orders.

### 5.3 Derivative accuracy

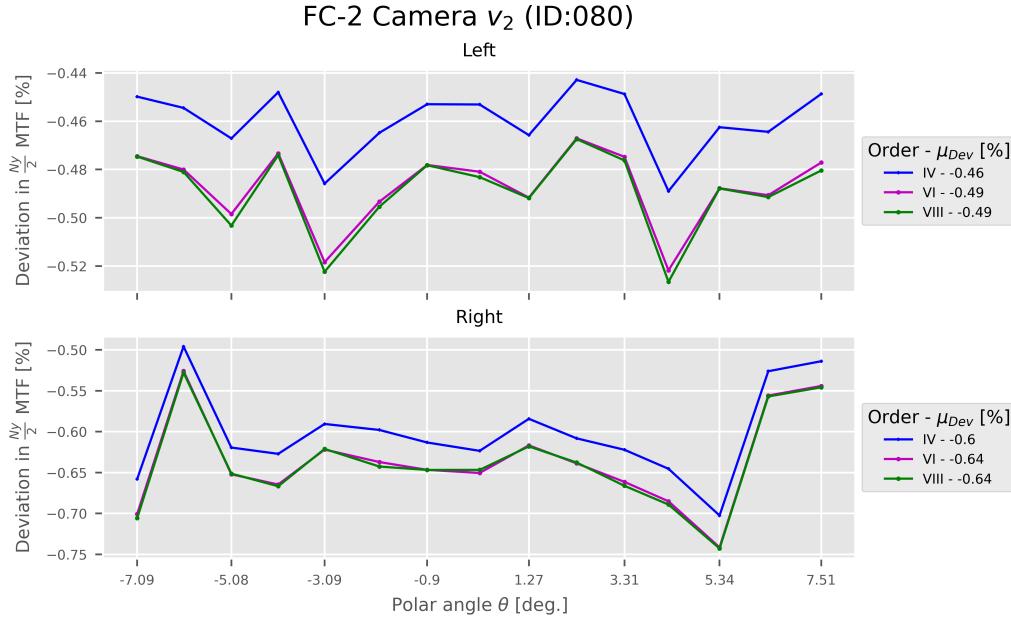


Figure 5.11:  $\frac{Ny}{2}$  MTF deviations vs.  $\theta$  for various orders.

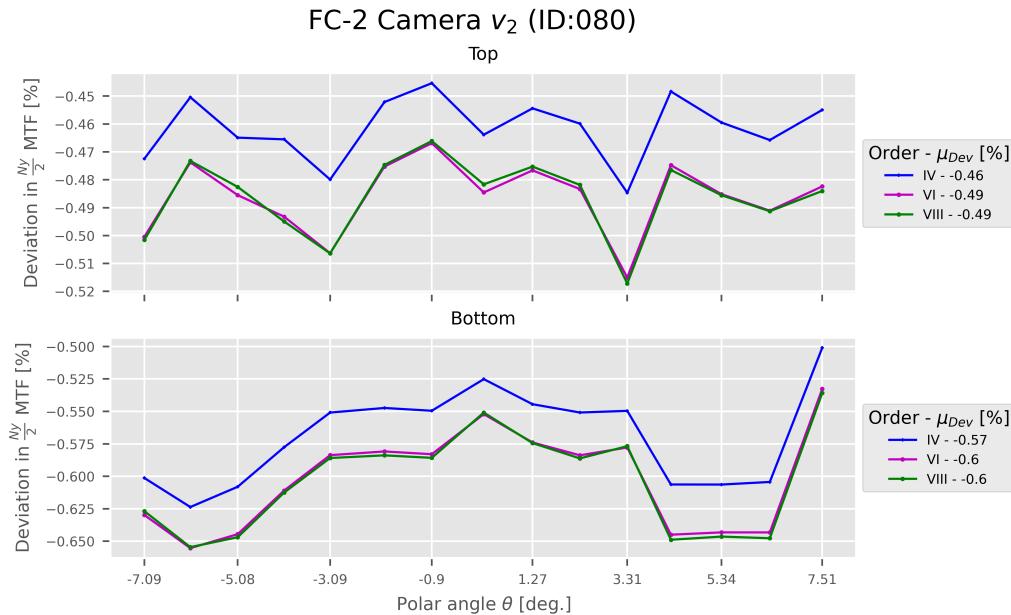


Figure 5.12:  $\frac{Ny}{2}$  MTF deviations vs.  $\theta$  for various orders.

- The relative differences between adjacent values in a plot are due to several other factors and are not attributable to the derivatives analysis.

### 5.3 Derivative accuracy

- The mean deviation of  $\frac{N_y}{2}$  MTF values for each plot is computed with respect to the original implementation recommended in [11] i.e. the central differences approach with second order truncation error.
- It can be noted that for the fourth order implementation, the reduction is  $\approx 0.5\%$  and for the sixth and eighth orders, the reduction increases to  $\approx 0.6\%$ . This also indicates the convergence of the analysis. Thus, even higher orders of accuracy are not analyzed.
- It can be concluded from this analysis that a second order finite differences approach to calculate the PSF, as recommended by [11] is an overestimation by roughly 0.5-0.6 %.

---

## 6 Input uncertainty

As introduced in Sec. 5, the sensitivity of MTF with respect to the parameters and with respect to input random variables have to be dealt separately as they require different approaches. The latter is covered in this section. The angle of the slanted edge  $\alpha$  was expected to possess some uncertainty from the empirical data obtained at the light tunnel. This belief was adequately supported by previous work on quantifying the uncertainty in slanted edge angles [14]. Thus, to begin with, the edge angle was assumed to be the only random variable in the slanted edge methodology.

### 6.1 Monte Carlo

As the problem at hand is fundamentally probabilistic in nature and also from what the law of large numbers suggest, the brute-force Monte Carlo simulation is an obvious choice to verify if the model assumptions are right and if the simulation is progressing in the right direction (towards convergence).

One arbitrarily sub-image from one target was chosen to be the base image upon which the Monte Carlo simulations were performed. The final estimate of the slope of the edge (direct result from function 3 in Fig. 4.3) was assumed to be the mean  $\mu$  and the standard deviation of the distribution was chosen to be  $\sigma = 1^\circ$  from empirical observations. In Fig. 6.1, a few such artificially sampled edge angles are pictorially represented (with the slope of the red lines being  $\tan(\alpha)$ )

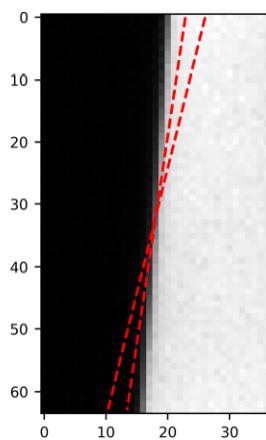


Figure 6.1: Monte Carlo sampling of edge angle  $\alpha$

To verify if this implementation was right, an image was chosen at random and the four sub-images were split. Then, the slope of one edge was chosen as the reference and was passed into the e-SFR algorithm for all the edges. The expected result was that, regardless of what happened to the other edges, the parent edge (whose angle was chosen as the reference) should return the true value of  $\frac{Ny}{2}$  MTF again. This was observed consistently with multiple images. This indicated that the method was working as intended and overriding the algorithm's edge angle estimation by passing in randomly sampled values was not adversely affecting the calculation of MTF in any manner.

Now that the implementation was working as intended, Monte Carlo analyses were performed on an arbitrarily chosen sub-image. Fig. 6.2 illustrates the distribution generated for the edge angle  $\alpha$  and the histograms for the distributions of  $\frac{Ny}{2}$  MTF and MTF50. Multiple Monte Carlo analyses were performed on the same image, with increasing sample size. The largest sample size was  $N_{samples} = 128000$  and larger simulations were not run owing to high computational cost, since every time, the entire e-SFR algorithm had to invo

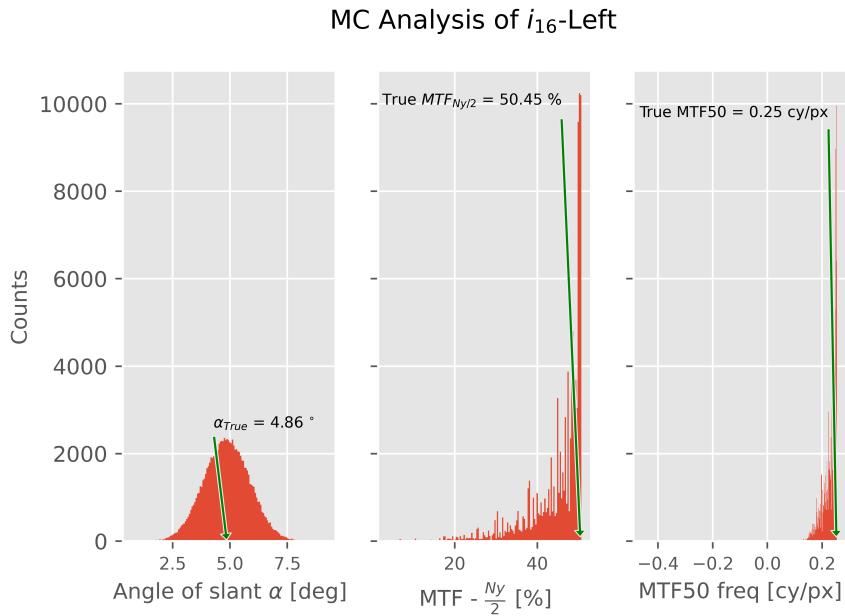


Figure 6.2: Monte Carlo with  $N=10^5$  with the original values estimated by e-SFR.

From Fig. 6.2, it is clear that there is one maximum value for  $\frac{Ny}{2}$  MTF and MTF50 beyond which there are absolutely no samples. This corroborates with the theory that regardless of how accurate the estimation of the edge angle is, the MTF is limited by the quality of the image itself.

In Fig. 6.3, the MTF metrics are plotted against the input distribution of  $\alpha$ . These are the response functions for the MTF metrics and give us insight into why the distributions from Fig. 6.2 are so narrow.

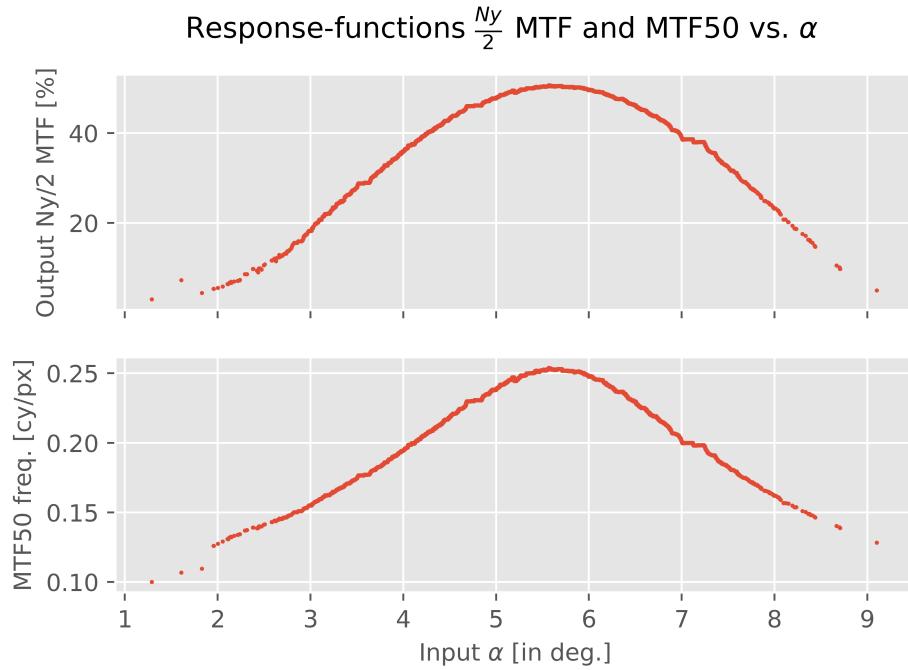


Figure 6.3:  $\frac{Ny}{2}$  MTF vs.  $\alpha$ , MTF50 vs.  $\alpha$

The response functions are highly nonlinear in nature and also behave quite differently in the extreme tails of the distributions. This is because for such extreme  $\alpha$  values, the offset calculated is so high that completely distorts the edge and renders it useless.

For the convergence of the Monte Carlo study, the distribution with 128k samples is taken as the reference with the mean and variance from that run being assumed as the reference. The relative error of the variance with respect to the reference distribution is computed, whose magnitude is plotted in Fig. 6.4

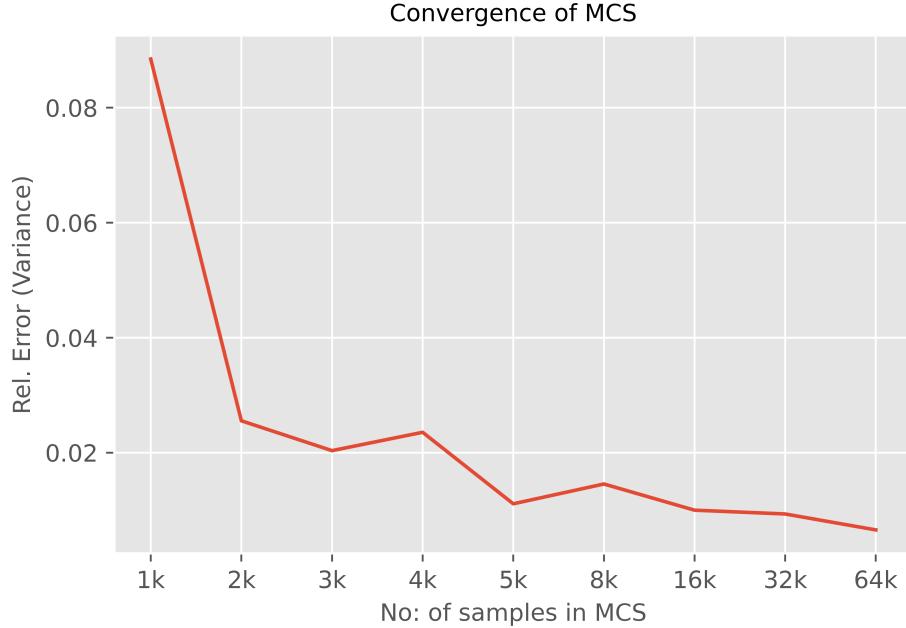


Figure 6.4: Convergence of Monte Carlo

It is a well known fact that brute force Monte Carlo is computationally draining and very slow at convergence. For instance, the reference distribution with 128k samples took roughly 47 minutes to complete which is extremely slow. For this reason, surrogate modeling approaches are beneficial as they reduce the variance in the quantity of interest and simultaneously improve the computational efficiency significantly.

## 6.2 Polynomial Chaos

As prior qualitative information behind the distribution of edge angle  $\alpha$  is not available, the edge angle is assumed to be normally distributed. Thus, the polynomials chosen were Hermite polynomials. Specifically, the probabilist's polynomials were chosen from *scipy* [26]. As the number of input random variables was 1, the number of samples required at each order of PCE was parameterized as  $N = 2 \times (p + 1)$ . Different orders of polynomial chaos were experimented with, from second order to seventh order. It was found that the second order PCE approximation fit the true response function the best and from the sixth order, the response surface started to oscillate rendering poor results (see Fig. 6.5 and Fig. 6.6).

## 6.2 Polynomial Chaos

---

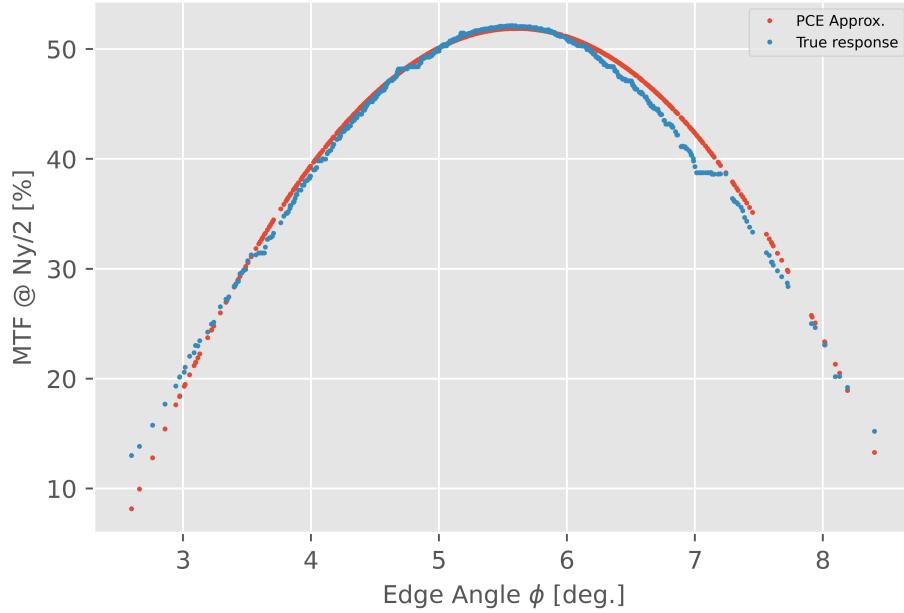


Figure 6.5: Best approximation by second order PCE

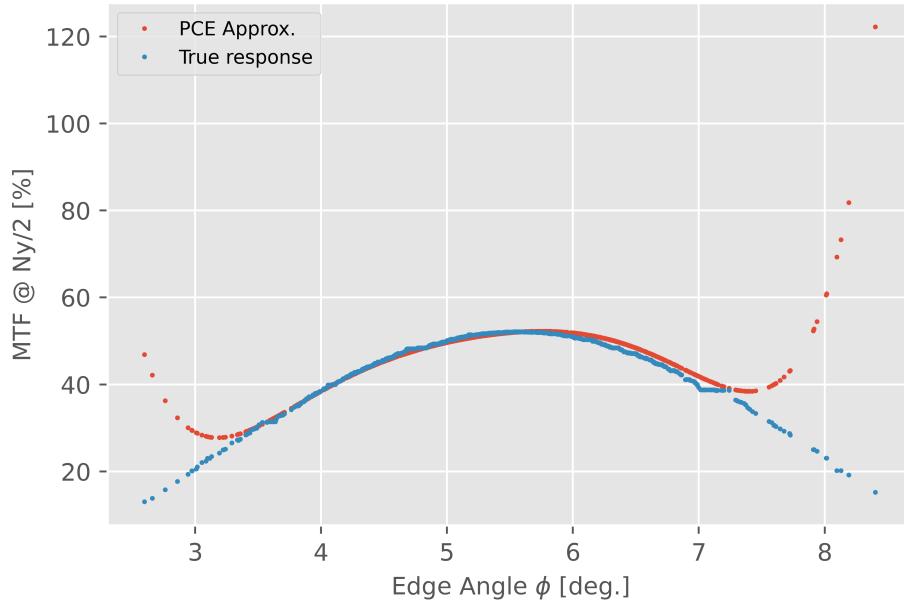


Figure 6.6: Poor approximation by seventh order PCE

The means and variances, for all the PCE orders are plotted along with the relative errors calculated with respect to the reference distribution's moments.

---

From this Fig. 6.7, it is confirmed that the PCE orders upto a certain order are consistently estimating the mean and variance after which the polynomials start oscillating. This indicates convergence. The huge computational gain behind using a surrogate model was that the seventh order estimation just required 16 samples and consumed 0.16 seconds.

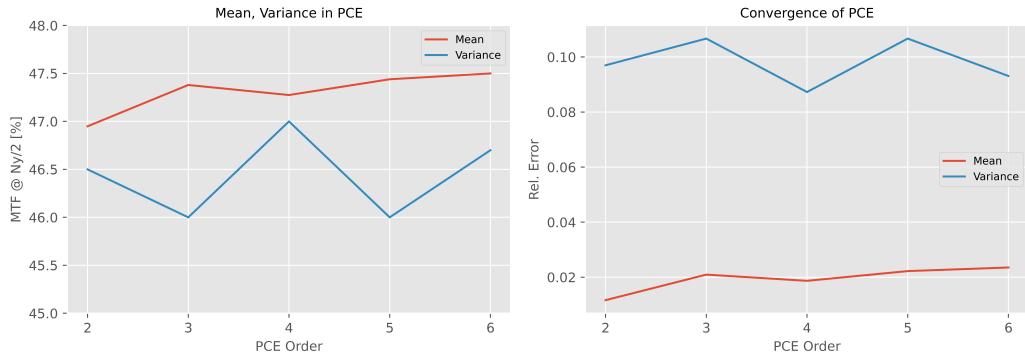


Figure 6.7: Convergence behaviour of  $\mu$  and  $\sigma$

## 7 Synthetic Data

Synthetic data, in this context refers to images that were not captured by an optical system but just generated by the computer by creating arrays of pixel intensities and adding noise components to it. The need for synthetic data arises because of two reasons -

1. Real data is scarce. It is extremely laborious to obtain real data because of the logistics and setup involved and also because of the availability of cameras. The cameras are supplied by an external supplier.
2. With synthetic data, the true edge angle in the image is known apriori. Since there are no non-uniform optical aberrations present to specifically target the edge, any huge deviation in the algorithm's angle estimation could be attributed to either the noise added or the error in the estimation process.

A function for a synthetic edge generator was written to take in a range of edge angles and different values for the Gaussian filter's size and standard deviation in each direction x and y. Image sizes were fixed at a standard  $100 \times 100$  pixels for uniformity. Since the edges, by design were built from the central point of the image, the requirement for optimizing the brightness/darkness ratio and

---

thereby truncating the width of the image is eliminated. For the additive noise, the Poisson-Gaussian noise model is adopted with the noise being expressed as

$$I_{\text{noise}}(x, y) = I(x, y) + \eta_{\text{poiss}}(I(x, y)) + \eta_{\text{gauss}}, \quad (7.1)$$

Here,  $\eta_{\text{poiss}}$  is the signal-dependent Poisson noise and  $\eta_{\text{gauss}}$  being the signal-independent Gaussian noise [15]. Multiple images are generated using this model with one such set of images illustrated in Fig. 7.1.

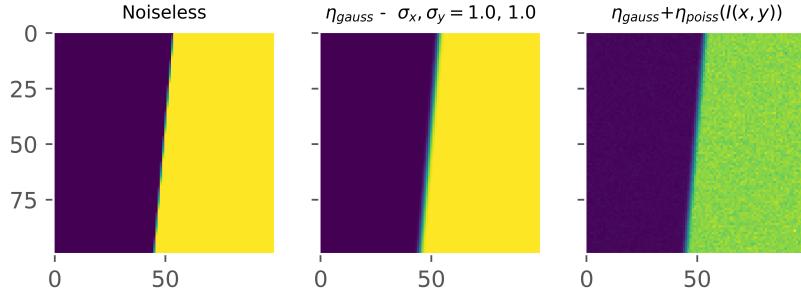


Figure 7.1: Synthetic images generated using the Poisson-Gaussian model

The MTF response for this synthetic image is as illustrated in Fig. 7.2.

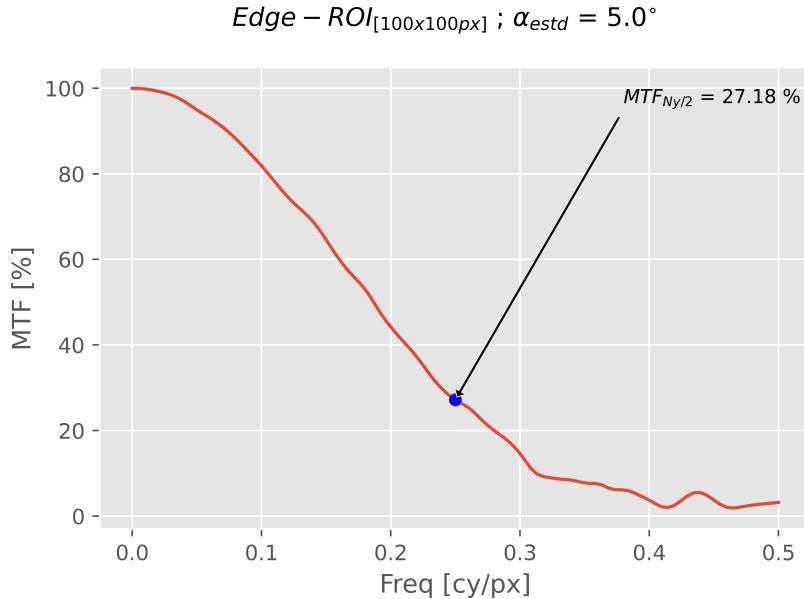


Figure 7.2: MTF curve of synthetic image (with noise).

One challenge with the analysis of synthetic images is that the functional relationship between additive noise and MTF is not known apriori. When there

---

is imaging equipment involved, it is known what is the expected SFR behaviour for it and in what range can MTF values be expected. But in the case of synthetic images, there is no expected behaviour. Thus, one can only compare the performance between different implementations of the e-SFR methodology [11]. One of Volkswagen's tier-1 suppliers work on projects in the same domain and the synthetic images generated from their end were tested by our implementation. Five images (with supposedly different optical qualities) were provided to us and the difference between the analyses are illustrated in Fig. 7.3.

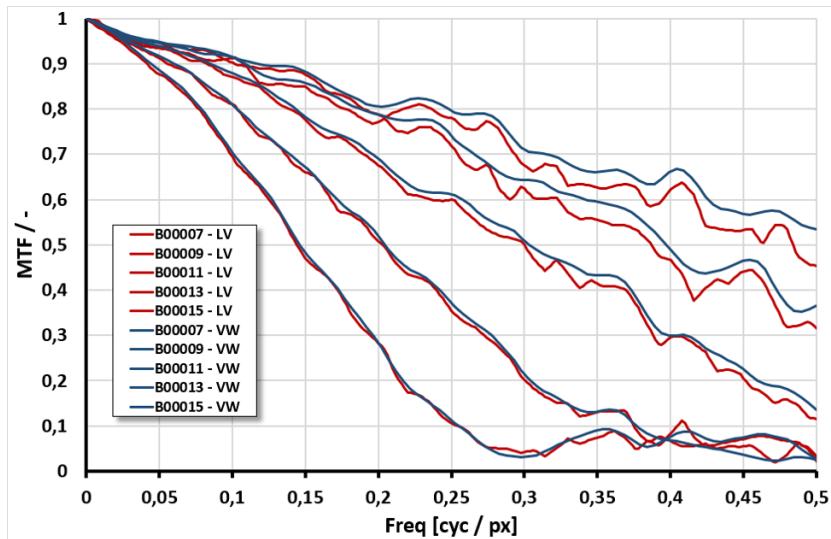


Figure 7.3: The tier-1 supplier's version of e-SFR underestimate our results.

In return, images generated by our edge generator (see Fig. 7.2) are currently being processed at their end and obtaining the results would help us further in analyzing how sensitive the MTF is to signal-dependent and independent noise.

## 8 Future Work

After my work here, I believe there is some more scope to improving the e-SFR algorithm's performance and robustness.

1. The impact of the oversampling ratio on the spatial frequency response has still not been understood completely and more importantly, not yet been quantified. This is because even within a single image, every row is being uniquely affected by optical aberrations and thus, no two rows have the same local intensity distribution. Further research on how aberrations

---

induce erroneous information into the image would be very beneficial in understanding the true behaviour of the MTF.

2. Within every bin in the oversampled ESF, an algebraic average is computed in [11]. But, other weighted averaging methods coupled with an error window for every pixel could potentially result in a more accurate representation of the edge.
3. Once the exact contribution of the projection process is understood, a global variance-based sensitivity index (like the Sobol index) could be developed, if not at a parameter level, but at least at a 'function' level.
4. Synthetic images could be studied in detail with the future goal of developing an advanced edge generator which models several optical aberrations close to how they are present and how they influence an image, in reality.

## 9 Conclusion

At the outset of this work, there were several open questions on whether the MTF values calculated by the in-house e-SFR algorithm were acceptable or not. Furthermore, there was little information available on whether the algorithm was sufficiently robust against different choices of parameters and more importantly, different images captured with constantly upgrading cameras. Through the course of this work, it was found and thus concluded that the ISO methodology and its implementation are robust enough (keeping in mind the precision limitation of roughly 2% MTF, see Sec. 1).

It was analytically found that the second order finite differences approach was overestimating the results and thus, for better accuracy, it is advised to use a derivative filter with fourth order truncation error. It was also found that the edge angle is one unique input which could be modelled probabilistically as a random variable. Synthetic images were found to be performing at a comparable level to the true images from the light tunnel, but sufficient insight is lacking on what to expect as results. Thus, a collaborative effort is required and this task is being worked on by us as well as a tier-1 supplier of ours, to improve understanding of how additive noise influences SFR.

## References

- [1] SAE International. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles.
- [2] Dimitris Milakis, Bart van Arem, and Bert van Wee. Policy and society related implications of automated driving: A review of literature and directions for future research. *Journal of Intelligent Transportation Systems*, 21(4):324–348, 2017.
- [3] Technical requirements for autonomous driving, howpublished = <https://www.iks.fraunhofer.de/en/topics/autonomous-driving.html>.
- [4] Christian Krebs, Patrick Müller, and Alexander Braun. Impact of wind-shield optical aberrations on visual range camera based classification tasks performed by cnns. *London Imaging Meeting*, 2(1):83–83, 2021.
- [5] Korbinian Weikl, Jeppe Revall Frisvad, Damien Schroeder, and Walter Stechele. Imaging through curved glass: Windshield optical impact on automotive cameras. In *ODS 2022: Industrial Optical Devices and Systems*, volume 12231, pages 55–64. SPIE, 2022.
- [6] Dominik Werner Wolf, Markus Ulrich, and Alexander Braun. Windscreen optical quality for ai algorithms: Refractive power and mtf not sufficient, 2023.
- [7] Kenneth Parulski, Peter D Burns, Dietmar Wueller, and Hideaki Yoshida. Creation and evolution of iso12233, the international standard for measuring digital camera resolution. *Image*, 347:2, 2022.
- [8] Jonathan B Phillips and Henrik Eliasson. *Camera image quality benchmarking*. John Wiley & Sons, 2018.
- [9] Ansi ph2.39 - method of measuring the photographic modulation transfer function of continuous-tone, black-and-white photographic films. Technical report.
- [10] Stephen Reichenbach, Stephen Park, and Ramkumar Narayanswamy. Characterizing digital image acquisition devices. *Optical Engineering*, 30:170–177, 02 1991.
- [11] Photography — Electronic still picture imaging — Resolution and spatial frequency responses. Standard, International Organization for Standardization, Geneva, CH, January 2017.

- [12] Peter D Burns, Kenichiro Masaoka, Kenneth Parulski, and Dietmar Wueller. Updated camera spatial frequency response for iso 12233. *Image*, 357:2, 2022.
- [13] Peter D Burns and Don Williams. Camera resolution and distortion: Advanced edge fitting. *Electronic Imaging*, 30:1–5, 2018.
- [14] Jackson KM Roland. A study of slanted-edge mtf stability and repeatability. In *Image quality and system performance XII*, volume 9396, pages 181–189. SPIE, 2015.
- [15] Alexandra Carlson, Katherine A Skinner, Ram Vasudevan, and Matthew Johnson-Roberson. Modeling camera effects to improve visual learning from synthetic data. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.
- [16] Frans Van den Bergh. Robust edge-spread function construction methods to counter poor sample spacing uniformity in the slanted-edge method. *JOSA A*, 36(7):1126–1136, 2019.
- [17] Kazuki Nishi. Does the slanted-edge method provide the true value of spatial frequency response? *J. Opt. Soc. Am. A*, 40(2):259–269, Feb 2023.
- [18] Critical angles, howpublished = <http://mtfmapper.blogspot.com/2015/06/critical-angles.html>, note = Accessed: 2015-06-15.
- [19] Kenichiro Masaoka. Edge-based modulation transfer function measurement method using a variable oversampling ratio: erratum. *Optics Express*, 30(1):179–179, 2022.
- [20] Ralph C Smith. *Uncertainty quantification: theory, implementation, and applications*, volume 12. Siam, 2013.
- [21] Dongbin Xiu and George Em Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2):619–644, 2002.
- [22] Charles R. Harris, K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'io, Mark Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy, September 2020.

- [23] Photography — Digital still cameras — Determination of exposure index, ISO speed ratings, standard output sensitivity, and recommended exposure index. Standard, International Organization for Standardization, Geneva, CH, February 2019.
- [24] Walter Schottky. On spontaneous current fluctuations in various electrical conductors. *Journal of Micro/Nanolithography, MEMS, and MOEMS*, 17(4):041001, 2018.
- [25] Ankit Rohatgi. Webplotdigitizer: Version 4.6, 2022.
- [26] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.
- [27] Cameron R. Taylor. Finite difference coefficients calculator. <https://web.media.mit.edu/~crtaylor/calculator.html>, 2016.