Diagnostics of laser light source illumination beyond industry-standard

Baron Gracias

Student Number: 189016690
Industry Partner: Malvern Panalytical
Academic Supervisor: Dr William Wadsworth
Industry Supervisor: Dr David Bryce

Department of Physics, University of Bath, UK, BA2 7AY
15^{th} October, 2024
Content-only page count: 17 pages

‡ §

1. Abstract

Malvern Panalytical produces analytical instrumentation, especially in particle sizing which utilises laser focusing technology. The main peak of a laser spot is characterised by the M-Squared parameter; the regions outside this, known as the "wings", provide valuable information. This research aims to develop a laser measurement system, enabling comparative light intensity measurements over 5 or more orders of magnitude including the main peak and wings. The development of a verification method was required, therefore, the well-known physics of the single slit diffraction experiment was used as a comparison. With a laser shone on the CMOS sensor of a DSLR, a combination algorithm was developed to stitch non-saturated regions of the image from various shutter speeds. This produced a high dynamic range image containing detail in the peak and wings. The algorithm proved successful, correctly replacing saturated regions with non-saturated regions; outputting an image with light intensities greater than 5 orders of magnitude from the peak. The intensity curve of the combined single slit diffraction pattern coincided with the theoretical curve, verifying the accuracy of this methodology. We find background heavily impacts the detail in the wings and consider algorithmic removal techniques.

‡ The use of the University of Bath logo is approved for student projects.
§ The use of the Malvern Panalytical logo was approved by Dr David Bryce.

Contents

1 Abstract **1**

Contents **2**

2 Introduction **4**
2.1 Malvern Panalytical . . . 4
2.2 Aims and Objectives . . . 4
2.3 Relevance to Malvern Panalytical . . . 4
2.4 Methodology Overview . . . 5
2.4.1 Choice of Method . . . 5
2.4.2 Laser Measurement System: Exposure Variation . . . 5
2.4.3 Measurement System Verification: Slit Diffraction . . . 5
2.5 Literature Review . . . 6

3 Detailed Theory **6**
3.1 Laser Physics . . . 6
3.1.1 Laser Fundamentals . . . 6
3.1.2 Existing Diagnostics . . . 6
3.1.3 Component Diffraction . . . 6
3.1.4 Neutral Density Filter . . . 7
3.1.5 Single Slit Physics . . . 7
3.2 Camera Optics . . . 7
3.2.1 Photon Detection . . . 7
3.2.2 Exposure Controls . . . 8
3.2.3 Bit Depth . . . 8
3.2.4 Image Artefacts . . . 8
3.2.5 Image Formatting . . . 9
3.2.6 Image Processing . . . 9

4 Methodology **10**
4.1 Image Collection . . . 10
4.2 Scale Factor Identification . . . 10
4.2.1 Purpose . . . 10
4.2.2 Experimental Set Up . . . 10
4.2.3 Measurements and Processing . . . 11
4.3 Exposure Variation . . . 11
4.4 Single Slit Diffraction . . . 11
4.5 Image Analysis . . . 12
4.5.1 Image Processing . . . 12
4.5.2 Image Combination . . . 12
4.5.3 Curve Fitting . . . 13
4.6 Image Artefact Identification . . . 13
4.6.1 Etaloning . . . 13
4.6.2 Dust Spots . . . 13
4.6.3 Burn-in . . . 13

5 Results **14**
5.1 Scale Factor and Background . . . 14
5.2 Exposure Variation . . . 14
5.3 Single Slit Diffraction . . . 14
5.4 Image Artefact Identification . . . 16
5.4.1 Etaloning . . . 16
5.4.2 Dust Spots . . . 16

6 Discussion **16**
6.1 Methodology Review . . . 17
6.1.1 Exposure Variation . . . 17
6.1.2 Single Slit Diffraction . . . 17
6.2 Combined Images and Algorithm Review . . . 17
6.2.1 Exposure Variation . . . 17
6.2.2 Single Slit Diffraction . . . 18
6.2.3 Combination Algorithm . . . 18
6.2.4 Background Analysis . . . 19
6.2.5 Algorithm Advancements . . . 19
6.3 Artefact Limitations . . . 19
6.4 Research Implications . . . 20

7 Conclusion **20**
7.1 Primary Findings . . . 20
7.2 Project Critique . . . 20
7.3 Continued Research . . . 20
7.4 Final Remarks . . . 20

Bibliography **21**

8 Appendix **22**
8.1 Sinc Squared Derivation . . . 22
8.2 Scale Factor Identification Code . . . 22
8.3 Background Analysis Code . . . 23
8.4 Exposure Variation Code . . . 24
8.4.1 Exposure Variation Slices . . . 24
8.4.2 Exposure Variation Combination Algorithm . . . 26
8.5 Single Slit Diffraction Code . . . 28
8.5.1 Single Slit Slices . . . 28
8.5.2 Single Slit Combination Algorithm . . . 30
8.6 Artefact Identification Code . . . 33
8.6.1 Etaloning Identification . . . 33
8.6.2 Dust Identification . . . 33

Acknowledgments

We would like to give special thanks to both of our industry and academic supervisors, Dr David Bryce and Dr William Wadsworth respectively, for their support throughout this project. As well as the laboratory technicians in supplying equipment and ensuring a suitable research environment, notably Isabel Wells and Dr Christopher Shearwood. Finally, the support of my project team in this research, thanks to Jessel Fernandes, Nat Jones and Harry Hill.

As project leader, my contributions included both experimentation and team management. I maintained the weekly logbook, produced Gantt charts on project delivery dates, produced a risk assessment chart, and organised meetings between the team, David and William. During experimentation, I formalised the etaloning experiment, and assisted in the exposure variation and slit diffraction experiments. I wrote the code for the scale factor, dust, etaloning identification, background analysis, and sinc squared curve fit algorithm. I assisted in developing the combination algorithm code, though I applaud Nat for his significant contributions to that code.

2. Introduction

2.1. Malvern Panalytical

Malvern Panalytical (abbreviated as Malvern in this report) is a leading analytical instrumentation company, notably in the advancement of particle sizing technology. Developing principle instruments designed to measure the size, shape and charge of particles. They help the world's leading companies, universities and researchers work on improving quality, optimizing production, or inventing entirely new medicines and materials[12]. Producing a wide range of products, such as particle size analysers, x-ray diffractometers and x-ray fluorescence spectrometers.

2.2. Aims and Objectives

We have partnered with Dr David Bryce from Malvern, to develop a methodology enabling *comparative light intensity measurements over 5 or more orders of magnitude*. With this as the primary goal, the following are also desired. The method should enable simultaneous comparison of the main peak and the wings of the laser. Avoid invasively affecting the measurements, such as the addition of too many components i.e. beam expansion optics. "Too many" is defined as introducing complex physics that does not allow simple derivation back to the original beam model. The method should provide good spatial resolution, be capable of measuring fluctuations at tens of kilohertz and capture more information than just intensity i.e. the phase of the light field.

The technique should be applicable across a range of technology and illumination types. It should be simple, straightforward and inventive, such that it could be placed within an existing instrument. Finally, there should be a verification to assess the accuracy of the method. To reiterate, the aim is to develop a *measurement system*, rather than making a measurement.

2.3. Relevance to Malvern Panalytical

Laser light sources are an essential component that can be found in almost any instrument, particularly for Malvern; high precision lasers are widely used in Malvern's particle size analysers. Characterising the laser beam and laser spot is vital in determining whether to tighten or relax system tolerances when analysing particles on the orders of micrometres.

The specification of beam intensity is widely identified by the M-Squared parameter, which determines how tightly the laser spot can be focused relative to the diffraction limit. This region is considered the "Main/Central Peak" of the laser spot and is defined by the FWHM, $1/e^2$, or D4σ (4 times

the distribution standard deviation)[17]. However, the "Wings" are the region outside the well-parameterized central spot. These wings can provide useful information on measurement sensitivity, impacting the upper or lower ranges of an instrument. Though, the intensity can be many orders of magnitude smaller than the centre, making the information difficult to analyse. Therefore, being able to produce a system to measure these wings accurately would be hugely beneficial to Malvern in improving reliability and precision.

2.4. Methodology Overview

2.4.1. Choice of Method

Reducing the objectives to simply a measurement system of light, it is clear that a photon intensity detection device is required. We can consider options such as a photodiode that linearly measures intensity, but we desire a device that provides multi-dimensional information from the light source (location and intensity). An Active-Pixel Sensor (APS) is the obvious answer, and in a readily available commercial device - a DSLR camera.

We want to avoid the invasive addition of components, so with the light source being a laser, the experimental set-up can be viewed as simply a laser shone directly onto the sensor of a DSLR camera. With the requirement that the measurement system works across a range of illumination types, the quality of the laser used is irrelevant. Developing a well working system with a poor beam quality laser is suitable, as it would more than likely work with a better quality laser too.

2.4.2. Laser Measurement System: Exposure Variation

With a DSLR, we can manually adjust photon detection parameters, particularly the ISO and Shutter Speed. The shutter speed is synonymous with photon exposure over time. Whereas, the ISO is the camera's sensitivity to light. Greater detail on these will be provided in Section 3.2.2.

Consider the large difference in intensity between the central laser spot and the wings. At low exposures, there is sufficient detail in the centre of the spot, but little detail in the wings. At higher exposures the opposite is true, with little detail in the centre due to the sensor pixels being "maxed out", and sufficient detail in the lower intensity wings. Therefore, we want to use multiple exposures (varying shutter speeds) to computationally build a continually detailed image of the spot, with detail in both the wings and peak. Developing an algorithm that determines the regions of an image which are over-exposed and low detail, replacing them with equivalent regions from another image at a lower exposure, hence higher detail. Comparative light intensity measurements up to 5 orders of magnitude are the desired dynamic range, so the output image should have suitable detail between the highest and lowest intensity at that difference in magnitude.

2.4.3. Measurement System Verification: Slit Diffraction

We want to verify the accuracy of this measurement system, therefore by using well-known physics, we can assess how our system fairs to expected values from the formulaic calculation. The single-slit experiment provides a great foundation. We can use the exposure variation method to produce a highly detailed laser diffraction image, with detail in the central peak and detail in the subsequent peaks. We can fit the well-known sinc-squared solution to the slit experiment to the combined diffraction data. How well the model fits the image will confirm how accurate the image combination method is. The number of peaks that are shown that match the theoretical data will be the gauge of accuracy. This is primarily a physical and real comparison to verify the measurement system.

2.5. Literature Review

Many techniques are available from laser instrumentation companies to characterise the main peak. These require the purchase of professional hardware and software, which can cost thousands of pounds e.g. Ophir Photonics laser beam profilers[13]. We find that the wings are desirably negated in beam calculations. Mentioned in an Ophir Photonics blog describing laser beam size calculations, stating that the D4σ value "can overly weight the wings if not using proper background subtraction"[11]. There is extensive research regarding the use of CCDs as laser image capturing devices. The paper *"Coupling quantum dots to optical fiber..."* demonstrates a 2D profile and 3D reconstruction of a laser spot, imaged using a CCD array system that had an impressive resolution of 1nm[3]. LabVIEW has been used to process laser images in *"Laser spot image acquisition and processing based on LabVIEW"*[18], though the spatial resolution and dynamic range in these images are far smaller than the relative magnitudes we seek. Artefacts seen in image capture using a CCD are discussed heavily in *"Typical Effects of Laser Dazzling CCD Camera"*[19]. Finally, background estimation and algorithmic removal techniques are investigated in *"CCD...Background Estimation Algorithm"*[15].

3. Detailed Theory

3.1. Laser Physics

3.1.1. Laser Fundamentals

A laser is an acronym for Light Amplification by Stimulated Emission of Radiation. Unlike other light sources, a laser emits coherent light, meaning the light waves travel in phase with similar wavelengths. The feature of spatial coherence allows lasers to be used for many purposes because a high energy density can be focused on a tight spot, giving a high irradiance (a measure of radiant flux per unit area on a surface). The process of stimulated emission to achieve lasing will not be detailed as this knowledge is irrelevant to the development of the measurement system. The two primary lasers used in this research are a Helium-Neon (HeNe) gas laser and a fibre laser. Both lasers output a bright red spot, consistent with their wavelength specification.

3.1.2. Existing Diagnostics

An ideal laser beam will have an intensity profile like that of a Gaussian distribution. As defined in Section 2.3, the M-squared parameter is a beam quality factor. This parameter indicates how close the beam is to this perfect Gaussian profile, essentially comparing the divergence and beam waist size to a Gaussian Beam. Defined as,

$$M^2 = \frac{\pi w \theta}{\lambda} \tag{1}$$

where λ is the laser wavelength, w is the beam waist and θ is the divergence angle. A perfect Gaussian beam will have an M-squared value of 1, with larger values indicating the beam is less like a Gaussian[2]. With this value showing how tightly the laser beam can be focused, it provides information on the main peak of the laser, though no information regarding the wings.

3.1.3. Component Diffraction

A consideration to be made is diffraction caused by components interfering with the beam path i.e. introducing additional optics such as telescopic or focusing lenses, mirrors or filters. These components add additional interference effects, like diffraction, phase differences or intensity reductions. The most common effect is diffraction, this is especially noticed with the pin-hole output aperture of the fibre optic laser. The pin-hole is a lever-actuated iris diaphragm, meaning it has 14 spokes that retract or expand to change the size of the output. Though, the output is not perfectly circular, with 14 spikes

seen on high exposure images corresponding to the iris spokes. Not only do we see these spikes, but we also observe concentric rings of intensity peaks as the pin-hole causes circular diffraction.

3.1.4. Neutral Density Filter

In the Single Slit experiment, neutral density filters are used to reduce the fraction of light hitting the sensor[16]. These filters reduce all wavelengths of light equally by a fraction equivalent to their ND value, e.g. ND2 is 1/2 of the light intensity. Using these filters allows for the selection of exposure controls that would otherwise produce an overexposed image.

3.1.5. Single Slit Physics

When a collimated beam of light encounters an obstacle, it spreads out, resulting in a pattern containing bright and dark regions, i.e. diffraction. The diffraction intensity profile from a single slit for small angles is defined as,

$$I(\beta) = I_0 \frac{\sin^2(\beta)}{\beta^2} = \operatorname{sinc}^2(\beta) \tag{2}$$

where $I(\beta)$ is the intensity at value β and I_0 is the central peak intensity. β is a function defined as,

$$\beta = \beta(\theta) = \frac{\pi d \sin(\theta)}{\lambda} \tag{3}$$

where d is the slit width, λ is the wavelength, and θ is the angle from the central peak[14]. This maps a sinc squared intensity function, an example of this function is shown in Fig. 1a, with Fig. 1b showing the logarithm of this function.

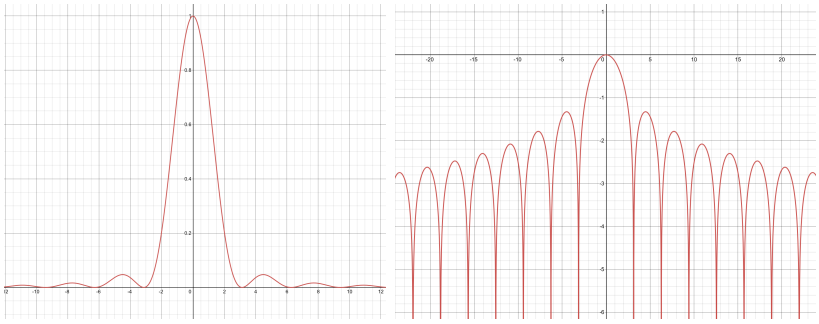

Figure 1. Left Fig. 1a, Right Fig. 1b
Fig. 1a shows a plot of $y(x) = \operatorname{sinc}^2(x)$ and Fig. 1b shows a plot of $y(x) = \log_{10}\left(\operatorname{sinc}^2(x)\right)$

3.2. Camera Optics

3.2.1. Photon Detection

We now consider how the DSLR (Digital Single-Lens Reflex) detects light. The camera has a mirror (shutter) that opens allowing light to hit the camera's sensor. A Canon EOS4000D uses an APS-C CMOS (Complimentary Metal-Oxide-Semiconductor) sensor, or simply, CMOS sensor. These are silicon blocks covered in a layer of glass for protection. Incoming photons are converted into an equivalent number of electrons via the photoelectric effect.

The CMOS consists of millions of these photodetector sensors, specifically detecting Red, Green and Blue wavelengths of light. Defined as RGB, and combined in an arrangement of blocks called

photosensors. These blocks are considered pixels and define the sensor resolution, commonly stated in Megapixels for the millions of pixels on the grid. The relative flux intensity transmitted then produces areas of light and dark across the grid. Green photosensors are considered luminance-sensitive elements, with red and blue as chrominance-sensitive. The human eye is most sensitive to green light, therefore twice as many green elements are used to achieve higher luminance resolution than chrominance[4]. Hence, the RGB colours are arranged in a Bayer Filter mosaic pattern, such that any pixel array has twice as many green detectors as red and blue - the RGB ratio is 1:2:1.

3.2.2. Exposure Controls

The camera has manual controls that alter how it detects photons, particularly the three exposure controls; Aperture, Shutter Speed and ISO. The aperture is the size of the camera opening and controls the amount of light entering the sensor; defined as f/x e.g. $f/1.2$. This changes the field depth, thereby, smaller apertures focus on both background and foreground, whereas larger apertures focus primarily on the foreground. The shutter speed controls the time the shutter is open, allowing photons to hit the CMOS; defined in seconds as $1/x$ e.g. as 1/100 or 15". Longer shutter speeds mean brighter images, as photons have been converted into intensity for a longer time, and conversely for shorter shutter speeds. The ISO is simply the CMOS's sensitivity to light, defined as $ISOx$ e.g. $ISO800$. Higher ISO's produce greater noise and a grainier image, hence a lower ISO is preferable.

3.2.3. Bit Depth

Understanding that the RGB pixels convert the photon energy to an equivalent electrical charge, we consider how this is quantified for the sensor. Bit depth, or pixel/colour depth, defines the number of colours that can be displayed by the sensor, i.e. the number of possible values detected per pixel. A greyscale image uses only one 'byte' per pixel, defined as 8 'bits'. A colour image utilises all three RGB values to create one full-colour pixel, each containing 8-bits, in a total of 24-bits for the three. A pixel with a bit depth of 1 has two (2^1) possible values, black or white. Subsequently, an 8-bit depth pixel has 256 (2^8) possible values and a 24-bit depth colour pixel has 16 million (2^{24}) possible values[8].

We can now consider the term saturation, which defines the intensity of the colour the pixel transmits, quantified by the RGB value. Over-saturation is when the pixel value is maxed out; in the case of an 8-bit depth image, it would read 256 (or 255 with 0 included) on all three RGB values even if the true value is far greater. This under-represents what the actual photon intensity is, and is a problem when seeking detail. Finally, the dynamic range of an image is the contrast ratio between the darkest and brightest colour tones, with higher bit depth sensors having higher dynamic ranges.

3.2.4. Image Artefacts

We will find certain artefacts that persist regardless of the experimental setup, these are all to be considered when analysing images. The most significant phenomenon is blooming (bleeding), which occurs when the maximum saturation of the CMOS is reached. When saturation occurs at a charge collection site, accumulation of additional photon charges results in an "overflow" or blooming, of the excess electrons into adjacent pixels[9] i.e. neighbouring pixels also read an over-saturated value. In the case of red laser light, the overflow of the Red sensor can bleed into the Green and Blue sensors, resulting in those intensities appearing to transmit from a red laser. Streaks of intensity from the source may also be found, commonly seen in CCTV footage of street lights or headlights.

Etaloning is an effect that occurs when light reflects between two flat parallel surfaces i.e. the light reflecting from the rear of the thin glass sheet covering the CMOS sensors. This produces interference fringes when the reflected waves interact with the incoming waves. Regions of light and dark intensity will be found across the image. If we consider the material thermodynamics, the glass will expand

due to the internal heating of the CMOS in natural operation. This increased width will result in a greater path length for the etaloning effect, identifiable by a movement of the interference fringes with respect to temperature and time[1].

Burn-in and dust spots are persistent artefacts that will be seen across all images taken. Dust spots are simply regions of darkness where dust is on the CMOS, this can be resolved by cleaning the glass, though inevitably one will find dust. Burn-in is due to the high photon energy density of the laser damaging CMOS pixels, causing these pixels to permanently transmit light regardless of incoming flux. This leaves specks of intensity transmitted throughout every image.

3.2.5. Image Formatting

The storage of the images taken is essential as the formats determine the compression and loss type of the data. Uncompressed formats retain all of the original CMOS captured data. Whereas compressed formats can be lossless, which retains original data but reduces file size, or lossy, which reduces file size and loses image quality. In either case, we want to avoid compressed formats to preserve maximum data. Formats like JPEG are often used to display images on the internet, applying lossy compression to reduce file size, dropping RGB data in the regions that the human eye is not sensitive to.

The images stored by the Canon DSLR used are stored in the CR2 RAW format, which stores lossless details without any processing, recording up to 14 bits per channel[6]. Though, these RAW images need to be processed into a format that can be analysed. TIFF is an ideal file type, saving uncompressed and lossless images whilst recording up to 16 bits per channel i.e. 65536 (or 65535 with 0 included) possible values. An observant reader will notice that this number provides almost 5 orders of magnitude, $\log_{10}(65536) \approx 4.8$; almost the dynamic range sought.

3.2.6. Image Processing

Before the RAW CR2 images are saved in TIFF format, the images need to be processed to represent true colour scales. With digital cameras, when twice the number of photons hit the sensor, it transmits twice the signal - a linear relationship that is stored in the RAW format. However, human eyes perceive twice the light as being a fraction brighter - a nonlinear/logarithmic relationship. Human eyes are far more sensitive to changes in dark tones than changes in bright tones, allowing our vision to operate over a broader range of luminance[5]; useful for perception in caves and sunlight.

Digital images are therefore, gamma encoded to translate the logarithmic sensitivity to the camera, making the images suitable for our eye's colour perception. The adjustment is defined as,

$$V_{out} = {V_{in}}^{\gamma} \tag{4}$$

where, V_{out} and V_{in} are the output and input/actual luminance values, and γ is the gamma adjustment value. A value of $\gamma < 1$ produces the logarithmic adjustment for our eyes, and a value of $\gamma > 1$ would be the inverse (quadratic-like). Image processing programs, like Canon Digital Photo Professional, automatically apply this adjustment. Since the gamma encoding redistributes tonal levels, fewer bits are needed to describe a given tonal range[5]; we want to maximise bits for suitable analysis. To restore the original tonal distribution, a linear factor is applied to the images. The linear factor is synonymous with an inverse gamma adjustment, such that the combination of gamma adjustment and inverse gamma adjustment produces a linear image. The figure below demonstrates this process.

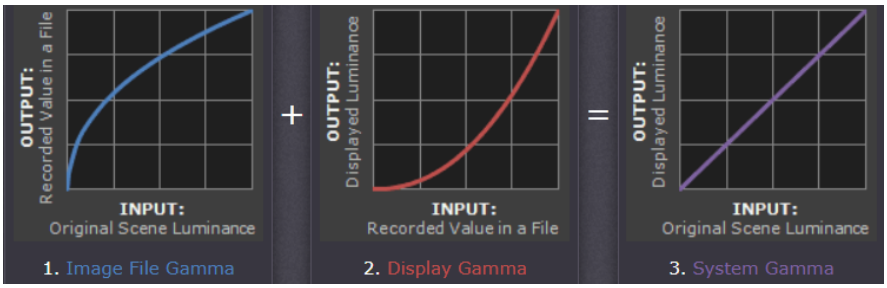

Figure 2. Graphic showing the input vs output image luminance.
Step 1 is the gamma adjusted logarithmic luminance relation, Step 2 is the Linear gamma factor relation applied to produce Step 3 - the final Linear luminance relation[5].

4. Methodology

4.1. Image Collection

The DSLR used was a lensless Canon EOS 4000D with an 18MP APS-C CMOS Sensor, connected directly to a computer. The Canon EOS Utility software allows real-time photo viewing and capture, with controls for shutter speed and ISO adjustment. The Canon Digital Photo Professional 4 (DPP4) allows for image processing and conversion. In all image collection experiments, the EOS Utility software was used to capture images and adjust the shutter speed and ISO as stated. Image shutter speeds and ISO will be presented as $name(SS, ISO)$, where $name$ is the image name, SS is the shutter speed and ISO is the ISO. The aperture was unchanged in all experiments, left as $f/1.4$ for maximum light entry. Background images were captured by placing the DSLR lens cap on the camera; taken at the shutter speed and ISO stated in the relevant experimental set-up.

4.2. Scale Factor Identification

4.2.1. Purpose

The aim is to make a laser measurement system that combines images of multiple exposures and variations of detail into a single high detail image. We must first determine whether the scale factor between different shutter speeds and ISOs is linear. So that the intensity of images at different exposures can be scaled relative to the base image combined. Two images at different shutter speeds and ISO's are taken and then divided by each other to confirm this scale factor. Doubling the shutter speed means double the light intensity, and half the ISO means half the light sensitivity, hence intensity. It is hypothesised that when the shutter speed is doubled and the ISO is halved between two images, the images should have the same intensity; thus a scale factor of 1 is assumed. This assumption means, with fixed ISO, halving the shutter speed results in half the intensity. e.g. the intensity of $(1/250, 200)$ is multiplied by $250/500 = 1/2$ when scaling to $(1/500, 200)$.

4.2.2. Experimental Set Up

The DSLR was placed at the beginning of an optical bench (0cm), the bench has a scale accuracy of 0.1cm. A Class 2 fibre optic laser, specified at 635nm, was placed 25cm further down the bench. The output of the fibre laser was attached to a lever-actuated iris diaphragm, with the output pinhole adjusted to the smallest size. The laser was pointed directly at the CMOS sensor, kept as perpendicular to the sensor as possible. All images were taken in quick succession in a dark room.

Table 1. The variation in shutter speed's and the corresponding scale factor values for Exposure Variation (EV), Single Slit Diffraction (SS) and Background (BG) images. ISO is locked at 200.

Image Name	Shutter Speed	Scale Factor
(EV,SS,BG)1	1/4000	4000/8
(EV,SS,BG)2	1/2000	2000/8
(EV,SS,BG)3	1/1000	1000/8
(EV,SS,BG)4	1/500	500/8
(EV,SS,BG)5	1/250	250/8
(EV,SS,BG)6	1/125	125/8
(EV,SS,BG)7	1/60	60/8
(EV,SS,BG)8	1/30	30/8
(EV,SS,BG)9	1/15	15/8
(EV,SS,BG)10	1/8	8/8

4.2.3. Measurements and Processing

Two images are taken, SF1(1/2000,400) and SF2(1/1000,200). Another two images of the background are taken, SFB1(1/2000,400) and SFB2(1/1000,200). The two images are imported into Python using the CV2 library. With the Red values isolated, the pixel location and intensity value are inserted into an array. The same is done with the background images, and these are negated from their corresponding scale factor image. The new SF1 array is divided by the SF2 array. A horizontal 1D slice is taken across the pixel axis. This can then be plotted against the intensity at that pixel to verify if the scale factor is on average 1 across the image.

4.3. Exposure Variation

This is the primary experiment; making the laser measurement system. Given the assumption that the scale factor is 1, we choose to vary the shutter speeds, whilst keeping the ISO locked at 200. This is due to the Canon DSLR having far more shutter speed options than ISO options. With multiple images at varying exposures and detail, we can combine these in processing to produce a high dynamic range laser spot.

Same set up as Scale Factor experiment, please see Section 4.2.2. With the ISO locked at 200, ten images are consecutively taken at the shutter speed shown in Table 1. The corresponding scale factors, relative to EV10 as the base image, are also displayed. Background images are then captured with the same shutter speeds, denoted as B1 through to B10.

4.4. Single Slit Diffraction

We must verify the accuracy of the laser measurement system by using known physics. By using the exposure variation method on the single slit diffraction experiment, we can verify the accuracy of the method. As we know what intensities to expect at each peak of the diffraction pattern, see Section 3.1.5, we can compare it to the combined exposure image. The sinc squared intensity pattern of the diffraction experiment should align with this combined intensity image, verifying the accuracy of the laser measurement system.

The DSLR was placed at the end of an optical bench that has a scale accuracy of 0.1cm. A 150μm diffraction grating (slit) was placed 75.69cm down the bench from the camera. A Class 2 HeNe laser, specified at 632.8nm, was placed at 80cm. Two neutral density filters, ND2, were placed between the

slit and the laser. The central peak of the diffraction pattern is aligned with the edge of the camera sensor, displaying as many intensity peaks as possible. The live view displayed this central spot on the left of the screen with the subsequent intensity peaks along the length of the sensor.

With the ISO locked at 200, ten images are taken at the shutter speed shown in Table 1. The corresponding scale factors, relative to SS10 as the base image, are also displayed.

4.5. Image Analysis

Please see Appendix for corresponding code in full, this section will describe what the code does.

4.5.1. Image Processing

The images captured were stored locally in DPP4. Within this software, a batch processing technique can be applied to multiple images simultaneously. The images are linearised by selecting the "Linear" checkbox and then converted and saved as 16 bit TIFF files on the computer. Please see Section 3.2.6.

Images from the Exposure Variation and Single Slit Diffraction experiment are loaded into Python using the image read function from the CV2 library. The convert colour function is then used to convert the BGR colour format to RGB. It then extracts the RGB intensity values into corresponding arrays. The R (Red) value is selected from the array as we care for only the red laser wavelengths detected. This also ignores the possible blooming into Green and Blue sensors.

We take a 1-dimensional slice across the image's horizontal pixel axis by determining the mean value of each row and using the row with the greatest mean. Therefore, in the case of the Single Slit images, the slice is taken across the diffraction peaks. This is the row used for all following slice images. All ten exposure slices are plotted on a logarithmic scale for the intensity, allowing us to compare the saturation at each exposure.

The corresponding background values are processed as before and then negated from the image arrays e.g. EV1−B1. N.B. all images stated from here will be background removed versions i.e. EV1−B1 is simply EV1. Similarly, with all single slit images are background negated. All exposures are then plotted to determine which shutter speeds to combine.

4.5.2. Image Combination

Currently, the algorithm works for only three images. EV1, EV5 and EV10 are selected as having suitable saturation ranges for the exposure variation combination. Similarly, SS1, SS5, SS10 for the single slit combination. The combination algorithm replaces pixels with values above or below the defined threshold with NaN's (not a number); removing them from further analysis. If no replacement is found, the existing value remains. The upper limit is defined as intensity values above 20000, and the lower limit as values below 100. For example, replacements are made for values below the lower limit in the EV1 exposure and values above the upper limit in the EV10 exposure. With EV5 as the middle exposure, a logical OR statement is used to replace values above the upper limit threshold OR below the lower limit threshold with NaN's.

EV1 and EV5 are then scaled according to EV10, refer to Table 1. The mean of the three arrays is taken as the combined slice. The mean is then used to normalise against the maximum value i.e. plotted with the maximum value as 1. This normalised mean array is now the final combined output; plotted as a 1D slice, then as a 2D contour plot and finally as a 3D plot. All of these plots are on a logarithmic scale to clearly indicate orders of magnitude. Similarly, for SS1, SS5 and SS10.

4.5.3. Curve Fitting

In the case of the Single Slit images, the combined arrays are compared to the real physics by overlaying the sinc squared diffraction solution. The physical pixel size of the sensor is used to convert the angular distance of the diffraction peaks into physical units. These can be inserted into the sinc squared function so that the curve is fitted with appropriate parameters. This curve is overlayed on the 1D combined slice plot for accuracy verification.

The following is the sinc squared function derived with real parameters[10], see Appendix Section 8.1 for derivation.

$$y\left(x\right) = \mathrm{sinc}^2\left(\frac{\pi ds}{\lambda D}x\right) \tag{5}$$

where $y\left(x\right)$ is the normalised intensity (y/y_0) at pixel location x, $d = 150\mu m$ is the diffraction grating width, $\lambda = 632.8nm$ is the laser wavelength, $D = 75.69cm$ is the grating to CMOS distance, and $s = 4.3\mu m$ is the CMOS pixel size[7]. This converts the intensity as a function of angle, to intensity as a function of the real pixel size on the CMOS. The range of values is then simply the width of the CMOS (5184 pixels) minus the midpoint pixel value of the central peak. N.B. this formula applies to any single slit experiment with any CMOS, given adjustments to λ,D,s and a.

4.6. Image Artefact Identification

4.6.1. Etaloning

We must verify that fringes on the laser spot images are an effect of etaloning. The same set-up as the Scale Factor experiment, please see Section 4.2.2. Now with a laser thermometer pointed at the CMOS of the DSLR. Two images are captured, the second image taken after leaving the setup and camera on for 30 minutes, with the temperature recorded at the start and end times. E1(1/4000,200) and E2(1/4000,200) were saved as CR2 files. The two images are then imported into python via the $Image.Open$ package from PIL, converted into arrays and displayed focused on the same pixel region showing the etaloning interference fringes.

4.6.2. Dust Spots

We must verify that dark spots across the laser spot are a result of dust on the CMOS lens. The same set-up as the Scale Factor experiment, please see Section 4.2.2. Two images are captured, the second image is captured with the laser shifted 1mm perpendicular to the length of the optical bench. D1(1/4000,200) and D2(1/4000,200) were saved as CR2 files. The two images are then imported into python via the $Image.Open$ package from PIL, converted into arrays and displayed focused on the same pixel region with the dust spot.

4.6.3. Burn-in

We must determine whether specks or streaks of high-intensity light are due to burn-in. A long shutter speed background image was investigated in DPP4 to find spots of abnormally high intensity. The longer shutter speeds will display these burn-in regions with greater contrast. A purely black background should have little to no intensity readings, but if specks of RGB values greater than 200 were found in DPP4, these were compared to other background or laser images to confirm whether they were found in the same pixel location.

5. Results

5.1. Scale Factor and Background

When the shutter speed is doubled and the ISO halved between two images, the images should have the same intensity; thus a scale factor of 1 is assumed. Fig. 3a shows the intensity of a slice taken across the ratio of the two images, SF1 and SF2. The intensities fluctuate across the slice but when an average is taken, we find it to be **1.05** to 3 s.f. This is approximately the hypothesised value of 1 with an error of 5%, confirming the scale factor used for varying shutter speed images.

Fig. 3b shows the distribution of intensities for a slice taken across BG1, BG5, BG10. A red horizontal line is plotted indicating the mean value for the three, we find it to be **11.7** to 3 s.f.

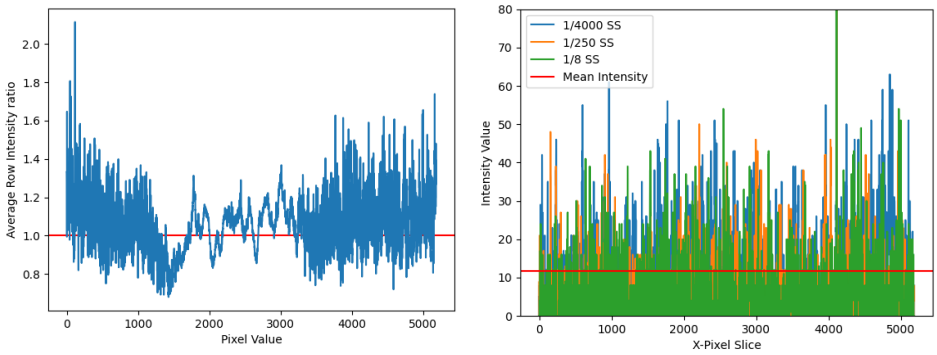

Figure 3. Left Fig. 3a, Right Fig. 3b.
Fig. 3a shows the ratio of intensities between SF1 and SF2 over a 1D slice. A horizontal red line is shown for the assumed linear scale factor.
Fig. 3b shows background slice intensities for BG1, BG5 and BG7. A horizontal red line indicates the mean value.

5.2. Exposure Variation

First we review the slices taken across the laser spot images at various exposures. Fig. 4a shows the horizontal slice taken across the image, with the pixel location on the x-axis and the logarithm of the normalised intensity on the y-axis. The various shutter speeds are indicated in the legend, all locked at ISO200. Fig. 4b shows the slices of EV1, EV5 and EV10 selected for the combination.

Next, we review the complete images of the three combined exposures from Fig. 4b. Fig. 5a shows a 3D plot of the combined exposure in both X-Y dimensions, with the pixel location on the x and y-axis and the logarithm of the normalised intensity on the z-axis. Fig. 5b shows a 2D contour plot of this combined exposure. We can see from the colour bar scale (or the z-axis) on Fig. 5a that we have detail of the laser spot across 5 orders of magnitude - from the top dark red region being 10^0, to the lower dark blue regions around 10^{-4} and 10^{-5}. There is no quantitative way to define what regions are no longer the laser due to the image artefacts, like blooming, therefore a minimum intensity cannot be accurately identified.

5.3. Single Slit Diffraction

All ten exposures and the three combined exposures have not been displayed as the image is synonymous with Fig. 4a & 4b. Instead, Fig. 6a shows the combined exposure from SS1, SS5 and SS10. The combined slice is shown in blue and the fitted sinc squared function is in orange, with almost 4 orders of magnitude. Fig. 6b then shows a 2D contour plot of this combined exposure. We notice vertical diffraction peaks originating from the central peak, repeated on the right of the image.

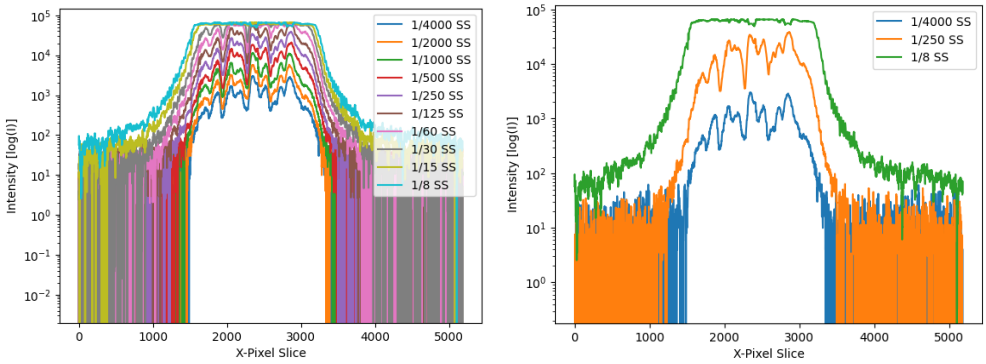

Figure 4. Laser Spot Normalised Log Intensity Slices: Left Fig. 4a, Right Fig. 4b.
Fig. 4a shows the slice intensity plots of all exposures according to Table 1, with the legend indicating the shutter speed.
Fig. 4b shows SS1, SS5 and SS10; the three exposures chosen to be combined.

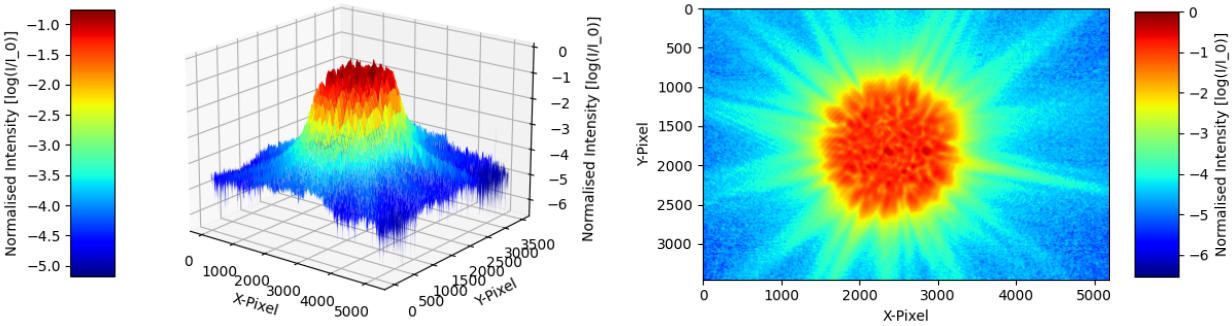

Figure 5. Laser Spot Normalised Log Intensity Plots: Left Fig. 5a, Right Fig. 5b.
Fig. 5a shows a 3D plot of the final combined exposure, with the normalised intensity shown in the colour bar.
Fig. 5b shows a 2D contour plot of the final combined exposure, with the normalised intensity shown in the colour bar.

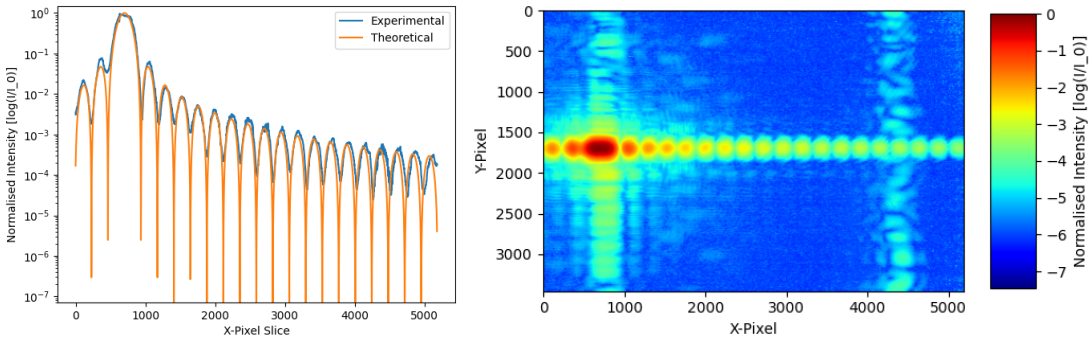

Figure 6. Single Slit Diffraction Combined Images: Left Fig. 6a, Right Fig. 6b.
Fig. 6a shows the final algorithm-combined slice in blue, overlayed with a fitted sinc squared function in orange.
Fig. 6b shows a 2D contour plot of the final combined exposure, with the normalised intensity shown in the colour bar.

5.4. Image Artefact Identification

5.4.1. Etaloning

With two images taken 30 minutes apart, the temperature measured on the CMOS increased by 12°C, from 14.4°C to 26.4°C. Fig. 7 shows the greyscale contrast of the interference fringes from one region of the laser spot, Fig. 7a as E1 and Fig. 7b as E2. We see that the fringes both shift and darken with increased temperature. A GIF switching between the two images demonstrates this shift clearly and confirms that these fringes are etaloning.

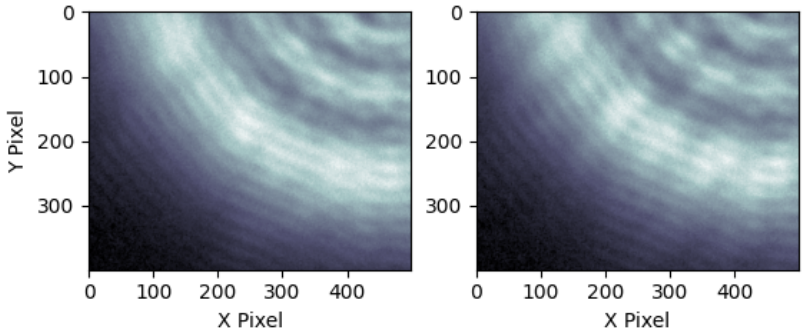

Figure 7. Etaloning interference fringes shift: Left Fig. 7a, Right Fig. 7b.
Fig. 7a shows the interference fringes at time, t=0mins, and temperature, T=14.4°C
Fig. 7b shows the interference fringes at time, t=30mins, and temperature, T=26.4°C

5.4.2. Dust Spots

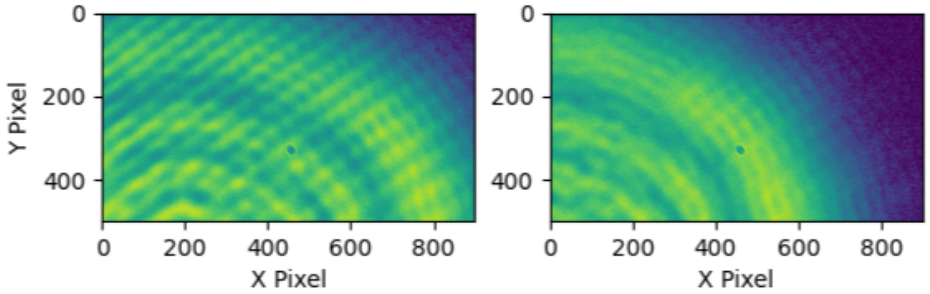

Figure 8. Dust spot shift: Left Fig. 8a, Right Fig. 8b.
Fig. 8a shows the initial dust spot on the laser image
Fig. 8b shows the final dust spot when the laser was moved horizontally by 1mm.

From Fig. 8a, we see a dark spot at around (450,360) on the X-Y pixel axis corresponding to D1. When the laser was shifted 1mm horizontally left, we see from Fig. 8b that the dark spot remains in the same place even though the laser spot has shifted left, corresponding to D2. Confirming that this dark spot is in fact dust on the CMOS sensor.

6. Discussion

The purpose of this research was to develop a laser measurement system and assess the characteristics of the output measurement. Therefore, the discussion will be split into two main sections; a Methodology Review and a Combined Image Review. The methodology review will be an assessment of the measurement system as a commercially viable device. The combined images review will assess the combination algorithm, output images, and background analysis.

6.1. Methodology Review

6.1.1. Exposure Variation

First, we consider the experimental set-up. To avoid invasively affecting measurements, the apparatus having simply a fibre optic laser shone on the CMOS of a DSLR camera is as minimal as possible. A measurement device and measurement source. Simply swapping the laser for other illumination sources would work accordingly. With the entirety of the measurement system being the two components set at an arbitrary distance from each other, this set-up could easily be implemented into an existing instrument for Malvern.

Although, beam optics were encountered with etaloning due to the glass cover of the CMOS, this is an unavoidable manufacture based issue. We did find beam optic effects with the iris diaphragm pinhole output of the fibre optic laser. The near-circular output produced circular diffraction of the laser spot, which was visible on all experimental images - notably Fig. 7 and Fig. 8. We also found 14 spikes of intensity beaming outwards from the central spot, corresponding to the 14 diaphragm spokes - seen in Fig. 5b. This could have been avoided by using a perfectly circular output to remove these spikes. Though, to remove the circular diffraction effects altogether, we avoid using a pinhole smaller than the diameter of the beamwidth. This was implemented by expanding the iris pinhole. We avoid dust spots by carefully cleaning the camera glass and the fibre optic laser output.

The process of converting and processing captured images was straightforward. The batch processing option in DPP4 sped this process up significantly, allowing the conversion to Linear and then TIFF to be very efficient. Finally, the large 5184x3456 sensor provided great spatial resolution.

6.1.2. Single Slit Diffraction

Now, we consider the verification method to assess the accuracy of the laser measurement system. Similar remarks to the Exposure Variation methodology review. Particularly, with etaloning, dust spots and image processing. The key methodology differences here were the use of a HeNe laser, the introduction of neutral density filters and a diffraction grating.

In Section 3.1.4, the role of neutral density filters is explained. The two filters reduced the measured intensity by 1/4, allowing for a greater number of diffraction peaks to be seen without over-saturation of the central peak. With the slit experiment being a verification method, the absolute intensities are irrelevant. We are comparing relative orders of magnitude to a fitted sinc squared function, therefore the absolute values here are unimportant. The thin layer of translucent glass may introduce backscatter and reflection of the incoming light, which can be reduced by using filters with anti-reflective coatings - a verification for future experiments.

The HeNe laser was introduced as a narrow beamwidth laser was desired to hit the diffraction grating. This was not possible with the fibre optic laser as narrowing the iris would induce significant circular diffraction and intensify the 14 light spikes. The HeNe was not suitable for the Exposure Variation experiment as the beam width spread significantly more in the far-field compared to the fibre optic laser (diffraction was induced in the near field for the slit experiment). This produced a largely grainy laser spot image. The choice between the two lasers was simply a leverage between pros and cons. Ultimately, worse quality lasers are preferable; if the methodology works for a qualitatively worse laser, it should work for a far better laser - especially those that Malvern uses.

6.2. Combined Images and Algorithm Review

6.2.1. Exposure Variation

The combination algorithm used EV1, EV5 and EV10, with the respective exposure slices shown in Fig. 4b. This selection was purely subjective, after reviewing the saturation regions of each exposure. Any combination of high, mid and low shutter speeds could be used and produce similar output. The combined output shown in Fig. 5a indicates at least 5 orders of magnitude between the normalised peak intensity to the lower regions. With the scale starting at 10^0 and ending at near 10^{-6}, the difference of five is certainly within this range. Detail is seen in both the central peak, shown in red, as well as detail in the wings from the yellow region outwards. The darker blue regions at magnitudes of -4 to -5, could be unaccounted background, but this is still the laser's wings. Fig. 5b shows this clearly, with the iris diaphragm spikes indicating laser light is in almost all regions - seen in light blue/cyan. Great detail in the central spot is clearly shown in this figure, although the detail in the wings could be improved. These definitions of detail are purely subjective, but with the overarching aim to show 5 orders of magnitude, we can see the algorithm works as needed.

6.2.2. Single Slit Diffraction

With SS1, SS5 and SS10 chosen as the exposure slices in Fig. 6a, the combined images in Fig. 6a and Fig. 6b indicate 3/4 orders of magnitude. This method did not achieve the 5 orders sought, though the importance is with the intensity peaks alignment. Fig. 6b shows 18 intensity peaks found to the right of the central peak. This coincides with the 18 intensity peaks shown by the theoretical fitted sinc squared function - shown in orange in Fig. 6a. The fact that the number of peaks of the sinc squared solution coincides with the combined output image indicates the accuracy of the method. The slice in this figure fitting to the theoretical curve indicates that the matching peaks are not simply a result of the fit, but rather an accurate image of the laser diffraction profile. We do see a slight deviation from the theoretical and experimental data towards the right of the image, this is likely due to the vertical peaks seen.

We consider these vertical intensity peaks originating from the central peaks and repeated on the far right of Fig. 6a. The peaks originating from the centre are due to the finite vertical length of the diffraction grating, acting as a tall rectangular aperture. This produces the well-known Fraunhofer diffraction with a rectangular aperture. Wider slits produce tighter intensity peaks, so the narrower vertical length scale corresponds to a larger length scale in real space. Since the laser has to be aligned with the CMOS perpendicularly, any misalignment from this would result in off-axis reflection i.e. to the left or right. Therefore, the vertical lines repeating on the far right of the image are likely due to reflection from the rear glass surface of the CMOS.

6.2.3. Combination Algorithm

In both exposure variation and single slit diffraction outputs, we see that the algorithm replaces the correct saturation regions with the non-saturated regions from a different exposure. We see in Fig. 4b that the central region of EV10 is completely saturated, this is replaced by the non-saturated central region of EV1. Similarly, for the wings and the joining slope in between the wings and main peak.

This method seems to work very well in capturing detail in the main peak, inevitably struggling at the wings. The wings have significantly less light in them and are easily washed out by the main peak. Even at higher shutter speeds, e.g. EV10, blooming from the main peak affects the detail in the wings. Consider Fig. 4b, the pixel by pixel detail in the central spot for EV1 (lowest shutter speed) is far greater than the detail in the wings for EV10. These higher shutter speed images *should* provide greater detail in the wings but the CMOS fails to capture them in as much detail, with significant image noise. This is an image capture issue than a fault in the combination algorithm. More images could be selected in the algorithm, though if the camera does not capture the detail in the wings, then this process is ultimately futile. Using a CMOS that can capture detail at high luminosity or testing the aperture controls may prove more effective.

6.2.4. Background Analysis

As discussed, the output images lose detail in the wings as they have significantly smaller intensity values than the central peak. Another factor that weighs into this lack of detail is how the background is accounted for. Since these low light intensity values are very close to, if not smaller than background intensity values. Hence, it is clear that removing background may inadvertently remove laser image data as well.

We consider the causes and effects of background light. We saw how image artefacts such as burn-in produce persistent high-intensity values - these are identified and acknowledged. We, otherwise, find two types of background contributions; background proportional to time and background that is a fixed value. Examples of fixed value backgrounds are the identified burn-in regions. As well as readout noise, which is a signal generated in the CMOS's amplifier, producing a fixed quantity transmitted during each reading. Time proportional background includes light leakage, cosmic rays and spontaneous electron tunnelling. These are simply possible causes of background and are inevitable, but hold as the primary reason to negate background from readings. The combination algorithm will incorrectly negate the fixed value backgrounds as they will be scaled according to the relevant image shutter speed. This can be remedied; in the case of burn-in, these regions can be removed by implementing an algorithm to determine extreme intensity values based on the local region.

Distribution analysis of the background over various shutter speeds seen in Fig. 3b indicates that the intensities do not increase with increased exposure time. This does not coincide with the expectation of time-dependent background and implies that background is incorrectly negated. This leads to inaccurate low-intensity regions i.e. the wings. We expect the distribution of background intensities to be normal about a mean, and peak in the same regions for different exposures. Since we do not see this, it suggests the background should be removed as a fixed average value for all images rather than a scaled quantity.

6.2.5. Algorithm Advancements

Programming a dynamic upper and lower limit for the combination algorithm may improve the method. The limits were subjectively defined as 20,000 and 100. Though, if limits could be defined according to the saturation levels of the image, we may find greater overall detail. This may form routes into machine learning, allowing for a software that identifies saturation regions to form a high dynamic range spot - limited by the detector's sensitivity. As described in Section 6.2.2; any combination of high, mid and low shutter speeds could be used and produce similar output. It would be interesting to code a looping algorithm that combines every combination of high, mid and low exposures. We could then determine which combination produces the highest detail output.

6.3. Artefact Limitations

The interference fringes due to etaloning affect certain results differently, such as the scale factor identification. Since we divide the two images, the peaks and troughs of the fringes should overlap and the average scale factor value is unaffected. Though, if the fringes shift through time-temperature dependence, the divided images would provide an incorrect average value since peaks may overlap with troughs. We also cannot assure that the glass cover is uniform in width. A non-uniform width cover would produce interference fringes of varying intensity, dependent on where the laser spot is on the sensor. This is prevalent in Fig. 8, where the laser spot shifting across the sensor appears to have deeper fringes in Fig. 8a than in Fig. 8b. If the glass was uniform, this can be avoided by capturing images consecutively, in short periods, to avoid the temperature increases significantly. We can avoid the non-uniform effects by keeping the laser spot in the same location as the sensor.

6.4. Research Implications

This research suggests that Malvern can effectively characterise the laser spot's main peak and wings. The simplicity of the method suggests commercial viability and relatively easy implementation into existing devices. This methodology is far more cost-effective than existing methods that require professional software and hardware, e.g. Ophir Photonics laser beam profilers[13]. The method is defined by the sensitivity of the detection device, which undoubtedly Malvern could invest in. The ability to simultaneously compare the wings of the laser spot at such orders of magnitude will allow Malvern to improve reliability and precision in their measurements. Specifically, determining whether to tighten or relax system tolerances when analysing microscopic particles.

7. Conclusion

7.1. Primary Findings

The exposure variation and combination algorithm methodology provided comparative light intensity measurements of the laser spot and wings at 5 orders of magnitude or more. The output images have a high dynamic range and good spatial resolution. The combination algorithm was inventive and unique; correctly replacing saturated regions with non-saturated regions from a different exposure. The method avoided the invasive addition of components, being at its core an illumination source shone on a CMOS. Subsequently, the technique was straightforward, such that it could easily be placed within an existing instrument with any light source. A verification method to assess the accuracy of the technique was devised and was successful. The experimental combined image matched the theoretical sinc squared curve, matching the 18 intensity peaks.

7.2. Project Critique

Although both the main peak and the wings were shown in the combined images, the detail in the wings was low relative to the main peak. This could be due to the lower limit set on the combination algorithm, incorrect background negation, or the CMOS's lack of low light resolution. This accuracy testing for this measurement system can be improved. The single slit diffraction method served as a "yes/no" verification for the combination algorithm from known physics. This provided little information on the saturation quality of the peaks, hence an expectation to compare the combined images would be desirable. Finally, the upper and lower limits of the algorithm were arbitrarily set, from subjective analysis of what over/under-saturated regions are defined as.

7.3. Continued Research

In Section 6.2.5, we discussed how the algorithm could be advanced. However, the aims we did not achieve were measuring fluctuations at tens of kilohertz and capturing more information than just intensity i.e. the phase of the light field. Fluctuations are time-dependent and would require a video rather than an image. This could be an advancement to the combination algorithm - stitching combined images at increasing time to show the fluctuations. A Michelson interferometer setup can be used to capture the phase information, though this would introduce many invasive components. Increased precision in the removal of background would greatly improve the detail in the wings. This could implement the algorithm techniques developed in *"CCD Camera Instrumental Background Estimation Algorithm"*[15].

7.4. Final Remarks

In conclusion, this methodology proved successful with prosperous foundations for development and advancements. The applications of laser imaging technology are both scientifically and financially beneficial. Malvern Panalytical and many professionals needing high dynamic range laser imaging can utilise the simplicity of the combination algorithm concept.

Bibliography

[1] Andor. Optical etaloning in charge coupled devices (ccd). Last accessed 25 April 2022. Available from: https://andor.oxinst.com/learning/view/article/optical-etaloning-in-charge-coupled-devices.

[2] Argueta, V. M squared and designing laser beam optics. Last accessed 25 April 2022. Available from: https://www.opticsforhire.com/blog/laser-beam-quality.

[3] Cheng, H., Mironov, A., Ni, J., Yang, H., Chen, W., Dai, Z., Dragic, P., Dong, J., Park, S.J. and Eden, J., 2015. Coupling quantum dots to optical fiber: Low pump threshold laser in the red with a near top hat beam profile. *Applied physics letters*, 106(8), p.081106.

[4] Cheremkhin, P., Lesnichii, V. and Petrov, N., 2014. Use of spectral characteristics of dslr cameras with bayer filter sensors. *Journal of physics: Conference series*. IOP Publishing, vol. 536, pp.012–021.

[5] Colour, C. in. Understanding gamma correction. Last accessed 25 April 2022. Available from: https://www.cambridgeincolour.com/tutorials/gamma-correction.htm.

[6] Cr2 - canon digital camera raw image file. Last accessed 25 April 2022. Available from: https://www.online-convert.com/file-format/cr2.

[7] ePHOTOzine. Aps-c sensors. Last accessed 26 April 2022. Available from: https://www.ephotozine.com/article/complete-guide-to-image-sensor-pixel-size-29652.

[8] Evening, M., 2013. *Adobe photoshop cc for photographers: A professional image editor's guide to the creative use of photoshop for the macintosh and pc*. Routledge.

[9] Fellers, T.J. and Davidson, M.W. Ccd saturation and blooming. Last accessed 25 April 2022. Available from: https://hamamatsu.magnet.fsu.edu/articles/ccdsatandblooming.html.

[10] HyperPhysics. Fraunhofer single slit. Last accessed 30 April 2022. Available from: http://hyperphysics.phy-astr.gsu.edu/hbase/phyopt/sinslit.html#c1.

[11] OphirBlog. How to calculate laser beam size. Last accessed 29 April 2022. Available from: https://blog.ophiropt.com/calculate-laser-beam-size/.

[12] Panalytical, M., 2022. About malvern instruments. Last accessed 10 April 2022. Available from: https://www.malvernpanalytical.com/en/about-us/.

[13] Photonics, O. Laser beam profilers. Last accessed 25 April 2022. Available from: https://www.ophiropt.com/laser--measurement/beam-profilers.

[14] Samuel J. Ling, Jeff Sanny, W.M. *University physics volume 3*.

[15] Sankowski, D. and Fabijanska, A., 2007. Ccd camera instrumental background estimation algorithm. *2007 ieee instrumentation & measurement technology conference imtc 2007*. IEEE, pp.1–5.

[16] Shelby, J., 2005. Optical materials — color filter and absorption glasses. In: R.D. Guenther, ed. *Encyclopedia of modern optics*. Oxford: Elsevier, pp.440–446. Available from: https://doi.org/https://doi.org/10.1016/B0-12-369395-0/00871-X.

[17] Thériault, G. The beginner's guide on spot size of laser beam. Last accessed 25 April 2022. Available from: https://www.gentec-eo.com/blog/spot-size-of-laser-beam.

[18] Zhang, Y., Li, Y., Gu, X., Liu, H., Zhang, Y. and Hu, W., 2019. Laser spot image acquisition and processing based on labview. *Optik*, 185, pp.505–509.

[19] Zhang, Z., Zhang, J., Shao, B., Cheng, D., Ye, X. and Feng, G., 2015. Typical effects of laser dazzling ccd camera. *Third international symposium on laser interaction with matter*. SPIE, vol. 9543, pp.50–56.

8. Appendix

8.1. Sinc Squared Derivation

The original sinc squared function is in the form,

$$I(\theta) = \text{sinc}^2\left(\frac{\pi d \sin(\theta)}{\lambda}\right) \tag{6}$$

where the variables are defined in Section 3.1.5. If we consider the right angled triangle below showing the laser diffraction path on the CMOS sensor, we see that for small angles $\sin(\theta) \approx \theta$.

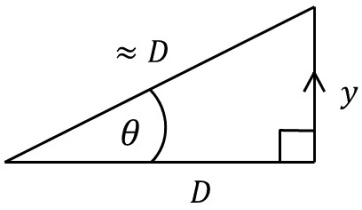

Figure 9. Diagram showing the right-angled triangle laser diffraction beam path. The hypotenuse is the diffraction path, the adjacent length to the angle is the distance from the diffraction grating to the sensor, and the opposite length is the distance from the central peak to the corresponding diffraction peak.

Here, $\theta = y/D$. If we divide through by the pixel size, s, we get $\theta = \frac{y/s}{D/s} = \frac{s}{D}x$. Where, $x = y/s$ is the conversion from angular y distance to the real distance on the sensor. Plugging this into Equation 6, we get the intensity function in terms of x pixels across the sensor

$$I(x) = \text{sinc}^2\left(\frac{\pi d s}{\lambda D}x\right). \tag{7}$$

8.2. Scale Factor Identification Code

```
import matplotlib.pyplot as plt
import numpy as np
import cv2

image1 = cv2.imread("SF_ISO400_SS1_2000.TIF",-1)
image2 = cv2.imread("SF_ISO200_SS1_1000.TIF",-1)
background_image = cv2.imread("BG_200_250.TIF", 1)

data1_ = cv2.cvtColor(image1,cv2.COLOR_BGR2RGB)
data2_ = cv2.cvtColor(image2,cv2.COLOR_BGR2RGB)

background_ = cv2.cvtColor(background_image,cv2.COLOR_BGR2RGB)
background = background_[:,:,0]

data1 = data1_[:,:,0]
data2 = data2_[:,:,0]

#Make 0s equal to nan

Nan_data1 = np.where(data1 == 0, np.nan, data1)
Nan_data2 = np.where(data2 == 0, np.nan, data2)

```

```
#Take the mean along the 1st index, ie collapse into a Nx1 array of means
means = np.mean(data1, 1)
#Now just get the index of the largest mean
idx = np.argmax(means)

# Divides the two image arrays
x = (Nan_data1/Nan_data2)
row_means = np.nanmean(x, 0)

# Average of the array
average = np.nanmean(x)
print(average)

# Slice data to find rolling division value
mean_data_ = row_means
plt.axhline(y=1, c='r',ls='-')
#plt.plot(data_slice)
plt.plot(mean_data_)
plt.xlabel("Pixel Value")
plt.ylabel("Average Row Intensity ratio")
plt.suptitle('Scale Factor Investigation', fontsize=16)
```

8.3. Background Analysis Code

```
import numpy as np
import matplotlib.pyplot as plt
import cv2

BG1 = cv2.imread("BG_200_4000.TIF", -1)
BG2 = cv2.imread("BG_200_250.TIF", -1)
BG7 = cv2.imread("BG_200_8.TIF", -1)

BGdata1_ = cv2.cvtColor(BG1, cv2.COLOR_BGR2RGB)
BGdata2_ = cv2.cvtColor(BG2, cv2.COLOR_BGR2RGB)
BGdata7_ = cv2.cvtColor(BG7, cv2.COLOR_BGR2RGB)

BGdata1 = BGdata1_[:, :, 0]
BGdata2 = BGdata2_[:, :, 0]
BGdata7 = BGdata7_[:, :, 0]

means = np.mean(BGdata1, 1)
idx = np.argmax(means)

BG_Slice1 = np.transpose(BGdata1[idx:idx + 1, :])
BG_Slice2 = np.transpose(BGdata2[idx:idx + 1, :])
BG_Slice7 = np.transpose(BGdata7[idx:idx + 1, :])

R_arr = [BG_Slice1,BG_Slice2,BG_Slice7]
R_mean = np.nanmean(R_arr)
print(R_mean)

fig1, ax1 = plt.subplots()
ax1.plot(BG_Slice1, label='1/4000 SS')
ax1.plot(BG_Slice2, label='1/250 SS')
ax1.plot(BG_Slice7, label='1/8 SS')
ax1.set_ylim(0,80)
```

```
ax1.set_xlabel("X-Pixel Slice")
ax1.set_ylabel("Intensity Value")
ax1.axhline(y=R_mean, c='r',ls='-', label='Mean Intensity')
ax1.legend(loc="upper left")
```

8.4. Exposure Variation Code

8.4.1. Exposure Variation Slices

```
import numpy as np
import matplotlib.pyplot as plt
import cv2

"""
Lines 10-76 are initial image importing and image readiness
"""

# Import images using openCV
image1 = cv2.imread("ISO200_SS1_4000.TIF", -1)
image2 = cv2.imread("ISO200_SS1_2000.TIF", -1)
image3 = cv2.imread("ISO200_SS1_1000.TIF", -1)
image4 = cv2.imread("ISO200_SS1_500.TIF", -1)
image5 = cv2.imread("ISO200_SS1_250.TIF", -1)
image6 = cv2.imread("ISO200_SS1_125.TIF", -1)
image7 = cv2.imread("ISO200_SS1_60.TIF", -1)
image8 = cv2.imread("ISO200_SS1_30.TIF", -1)
image9 = cv2.imread("ISO200_SS1_15.TIF", -1)
image10 = cv2.imread("ISO200_SS1_8.TIF", -1)

BG1 = cv2.imread("BG_200_4000.TIF", -1)
BG2 = cv2.imread("BG_200_2000.TIF", -1)
BG3 = cv2.imread("BG_200_1000.TIF", -1)
BG4 = cv2.imread("BG_200_500.TIF", -1)
BG5 = cv2.imread("BG_200_250.TIF", -1)
BG6 = cv2.imread("BG_200_125.TIF", -1)
BG7 = cv2.imread("BG_200_60.TIF", -1)
BG8 = cv2.imread("BG_200_30.TIF", -1)
BG9 = cv2.imread("BG_200_15.TIF", -1)
BG10 = cv2.imread("BG_200_8.TIF", -1)

# Convert to RGB
data1_ = cv2.cvtColor(image1, cv2.COLOR_BGR2RGB)
data2_ = cv2.cvtColor(image2, cv2.COLOR_BGR2RGB)
data3_ = cv2.cvtColor(image3, cv2.COLOR_BGR2RGB)
data4_ = cv2.cvtColor(image4, cv2.COLOR_BGR2RGB)
data5_ = cv2.cvtColor(image5, cv2.COLOR_BGR2RGB)
data6_ = cv2.cvtColor(image6, cv2.COLOR_BGR2RGB)
data7_ = cv2.cvtColor(image7, cv2.COLOR_BGR2RGB)
data8_ = cv2.cvtColor(image8, cv2.COLOR_BGR2RGB)
data9_ = cv2.cvtColor(image9, cv2.COLOR_BGR2RGB)
data10_ = cv2.cvtColor(image10, cv2.COLOR_BGR2RGB)

BGdata1_ = cv2.cvtColor(BG1, cv2.COLOR_BGR2RGB)
BGdata2_ = cv2.cvtColor(BG2, cv2.COLOR_BGR2RGB)
BGdata3_ = cv2.cvtColor(BG3, cv2.COLOR_BGR2RGB)
BGdata4_ = cv2.cvtColor(BG4, cv2.COLOR_BGR2RGB)
```

```
BGdata5_ = cv2.cvtColor(BG5, cv2.COLOR_BGR2RGB)
BGdata6_ = cv2.cvtColor(BG6, cv2.COLOR_BGR2RGB)
BGdata7_ = cv2.cvtColor(BG7, cv2.COLOR_BGR2RGB)
BGdata8_ = cv2.cvtColor(BG8, cv2.COLOR_BGR2RGB)
BGdata9_ = cv2.cvtColor(BG9, cv2.COLOR_BGR2RGB)
BGdata10_ = cv2.cvtColor(BG10, cv2.COLOR_BGR2RGB)

# Select the red values of the data arrays
data1 = data1_[:, :, 0]
data2 = data2_[:, :, 0]
data3 = data3_[:, :, 0]
data4 = data4_[:, :, 0]
data5 = data5_[:, :, 0]
data6 = data6_[:, :, 0]
data7 = data7_[:, :, 0]
data8 = data8_[:, :, 0]
data9 = data9_[:, :, 0]
data10 = data10_[:, :, 0]

BGdata1 = BGdata1_[:, :, 0]
BGdata2 = BGdata2_[:, :, 0]
BGdata3 = BGdata3_[:, :, 0]
BGdata4 = BGdata4_[:, :, 0]
BGdata5 = BGdata5_[:, :, 0]
BGdata6 = BGdata6_[:, :, 0]
BGdata7 = BGdata7_[:, :, 0]
BGdata8 = BGdata8_[:, :, 0]
BGdata9 = BGdata9_[:, :, 0]
BGdata10 = BGdata10_[:, :, 0]

"""
Lines 83-122 are for the multi-exposure slice images
"""

# Take the mean of each row and collapsing into Nx1 array
means = np.mean(data1, 1)
# Now just get the index of the largest mean
idx = np.argmax(means)

# Plot slices to show values are as expected, NB ~ no background
NB_data_slice1 = np.transpose(data1[idx:idx + 1, :] - BGdata1[idx:idx + 1, :])
NB_data_slice2 = np.transpose(data2[idx:idx + 1, :] - BGdata2[idx:idx + 1, :])
NB_data_slice3 = np.transpose(data3[idx:idx + 1, :] - BGdata3[idx:idx + 1, :])
NB_data_slice4 = np.transpose(data4[idx:idx + 1, :] - BGdata4[idx:idx + 1, :])
NB_data_slice5 = np.transpose(data5[idx:idx + 1, :] - BGdata5[idx:idx + 1, :])
NB_data_slice6 = np.transpose(data6[idx:idx + 1, :] - BGdata6[idx:idx + 1, :])
NB_data_slice7 = np.transpose(data7[idx:idx + 1, :] - BGdata7[idx:idx + 1, :])
NB_data_slice8 = np.transpose(data8[idx:idx + 1, :] - BGdata8[idx:idx + 1, :])
NB_data_slice9 = np.transpose(data9[idx:idx + 1, :] - BGdata9[idx:idx + 1, :])
NB_data_slice10= np.transpose(data10[idx:idx + 1,:] - BGdata10[idx:idx + 1, :])

fig1, ax1 = plt.subplots()
ax1.plot(NB_data_slice1, label='1/4000 SS')
ax1.plot(NB_data_slice2, label='1/2000 SS')
ax1.plot(NB_data_slice3, label='1/1000 SS')
ax1.plot(NB_data_slice4, label='1/500 SS')
ax1.plot(NB_data_slice5, label='1/250 SS')
```

```
ax1.plot(NB_data_slice6, label='1/125 SS')
ax1.plot(NB_data_slice7, label='1/60 SS')
ax1.plot(NB_data_slice8, label='1/30 SS')
ax1.plot(NB_data_slice9, label='1/15 SS')
ax1.plot(NB_data_slice10, label='1/8 SS')
ax1.legend(loc="upper right")
ax1.set_yscale('log')
ax1.set_xlabel("X-Pixel Slice")
ax1.set_ylabel("Intensity [log(I)]")

fig2, ax2 = plt.subplots()
ax2.plot(NB_data_slice1, label='1/4000 SS')
ax2.plot(NB_data_slice5, label='1/250 SS')
ax2.plot(NB_data_slice10, label='1/8 SS')
ax2.legend(loc="upper right")
ax2.set_yscale('log')
ax2.set_xlabel("X-Pixel Slice")
ax2.set_ylabel("Intensity [log(I)]")
```

8.4.2. Exposure Variation Combination Algorithm

```
import numpy as np
import matplotlib.pyplot as plt
import cv2
import time

t_start = time.time()

"""
Lines 12-43 are initial image importing and image readiness
"""

# Import images using openCV
image1 = cv2.imread("ISO200_SS1_4000.TIF", -1)
image5 = cv2.imread("ISO200_SS1_250.TIF", -1)
image10 = cv2.imread("ISO200_SS1_8.TIF", -1)

#Import background
b1 = cv2.imread("BG_200_4000.TIF", -1)
b5 = cv2.imread("BG_200_250.TIF", -1)
b10 = cv2.imread("BG_200_8.TIF", -1)

# Convert to RGB
data1_ = cv2.cvtColor(image1, cv2.COLOR_BGR2RGB)
data5_ = cv2.cvtColor(image5, cv2.COLOR_BGR2RGB)
data10_ = cv2.cvtColor(image10, cv2.COLOR_BGR2RGB)

b1_ = cv2.cvtColor(b1, cv2.COLOR_BGR2RGB)
b5_ = cv2.cvtColor(b5, cv2.COLOR_BGR2RGB)
b10_ = cv2.cvtColor(b10, cv2.COLOR_BGR2RGB)

# Select the red values of the data arrays
data1 = data1_[:, :, 0]
data5 = data5_[:, :, 0]
data10 = data10_[:, :, 0]

```

```
back1 = b1_[:, :, 0]
back5 = b5_[:, :, 0]
back10 = b10_[:, :, 0]

#Remove background
im1 = data1 - back1*1.0
im5 = data5 - back5*1.0
im10 = data10 - back10*1.0

"""
Lines 49-69 are the combination algorithm
"""

# Set array conditions
top_limit = 20000.0
bottom_limit = 100.0

lim10 = np.where(im10>(top_limit), np.nan, im10)
lim5 = np.where(np.logical_or(im5>(top_limit), im5<(bottom_limit)), np.nan, im5)
lim1 = np.where(im1<bottom_limit, np.nan, im1)

# Scale all of the data to highest exposure
fin1 = lim1 *(4000/8.0)
fin5 = lim5 *(250/8.0)
fin10 = lim10 *(8/8.0)

# Take mean of arrays
R_arr = [fin1,fin5,fin10]
R_mean = np.nanmean(R_arr, axis=0)

# Normalise answer
normalisation = np.nanmax(R_mean)
mean_= R_mean / normalisation
mean = np.where(np.logical_or(mean_<=0.0, mean_=='nan'),10.0e-6, mean_)

"""
Lines 75-99 are for the combination algorithm images
"""

# 2-D Contour plot of full data
fig1, ax1 = plt.subplots()
imag1 = ax1.imshow(np.log10(mean), cmap = 'jet')
ax1.set_xlabel('X-Pixel')
ax1.set_ylabel('Y-Pixel')
fig1.colorbar(imag1, shrink=0.7, aspect=6, label='Normalised Intensity [log(I/I_0)]')
plt.show()

# 3D log plot of the full data
X = np.arange(1,mean.shape[1]+1)
Y = np.arange(1, mean.shape[0]+1)
X, Y = np.meshgrid(X, Y)

fig2 = plt.figure()
ax2 = fig2.gca(projection='3d')
ax2.set_xlabel('X-Pixel')
ax2.set_ylabel('Y-Pixel')
ax2.set_zlabel('Normalised Intensity [log(I/I_0)]')
```

```
imag2 = ax2.plot_surface(X, Y, np.log10(mean), cmap = 'jet')

fig2.colorbar(imag2, shrink=0.7, aspect=6, location='left', label='Normalised
↪  Intensity [log(I/I_0)]')
plt.show()

t_end = time.time()
print(t_end-t_start, 'seconds')
```

8.5. Single Slit Diffraction Code

8.5.1. Single Slit Slices

```
import numpy as np
import matplotlib.pyplot as plt
import cv2

"""
Lines 10-79 are initial image importing and image readiness
"""

# Import images using openCV
image1 = cv2.imread("Slit_ISO200_SS1_4000.TIF", -1)
image2 = cv2.imread("Slit_ISO200_SS1_2000.TIF", -1)
image3 = cv2.imread("Slit_ISO200_SS1_1000.TIF", -1)
image4 = cv2.imread("Slit_ISO200_SS1_500.TIF", -1)
image5 = cv2.imread("Slit_ISO200_SS1_250.TIF", -1)
image6 = cv2.imread("Slit_ISO200_SS1_125.TIF", -1)
image7 = cv2.imread("Slit_ISO200_SS1_60.TIF", -1)
image8 = cv2.imread("Slit_ISO200_SS1_30.TIF", -1)
image9 = cv2.imread("Slit_ISO200_SS1_15.TIF", -1)
image10 = cv2.imread("Slit_ISO200_SS1_8.TIF", -1)
image11 = cv2.imread("Slit_ISO200_SS1_4.TIF", -1)

BG1 = cv2.imread("BG_200_4000.TIF", -1)
BG2 = cv2.imread("BG_200_2000.TIF", -1)
BG3 = cv2.imread("BG_200_1000.TIF", -1)
BG4 = cv2.imread("BG_200_500.TIF", -1)
BG5 = cv2.imread("BG_200_250.TIF", -1)
BG6 = cv2.imread("BG_200_125.TIF", -1)
BG7 = cv2.imread("BG_200_60.TIF", -1)
BG8 = cv2.imread("BG_200_30.TIF", -1)
BG9 = cv2.imread("BG_200_15.TIF", -1)
BG10 = cv2.imread("BG_200_8.TIF", -1)

# Convert to RGB
data1_ = cv2.cvtColor(image1, cv2.COLOR_BGR2RGB)
data2_ = cv2.cvtColor(image2, cv2.COLOR_BGR2RGB)
data3_ = cv2.cvtColor(image3, cv2.COLOR_BGR2RGB)
data4_ = cv2.cvtColor(image4, cv2.COLOR_BGR2RGB)
data5_ = cv2.cvtColor(image5, cv2.COLOR_BGR2RGB)
data6_ = cv2.cvtColor(image6, cv2.COLOR_BGR2RGB)
data7_ = cv2.cvtColor(image7, cv2.COLOR_BGR2RGB)
data8_ = cv2.cvtColor(image8, cv2.COLOR_BGR2RGB)
data9_ = cv2.cvtColor(image9, cv2.COLOR_BGR2RGB)
data10_ = cv2.cvtColor(image10, cv2.COLOR_BGR2RGB)
```

```
data11_ = cv2.cvtColor(image11, cv2.COLOR_BGR2RGB)

BGdata1_ = cv2.cvtColor(BG1, cv2.COLOR_BGR2RGB)
BGdata2_ = cv2.cvtColor(BG2, cv2.COLOR_BGR2RGB)
BGdata3_ = cv2.cvtColor(BG3, cv2.COLOR_BGR2RGB)
BGdata4_ = cv2.cvtColor(BG4, cv2.COLOR_BGR2RGB)
BGdata5_ = cv2.cvtColor(BG5, cv2.COLOR_BGR2RGB)
BGdata6_ = cv2.cvtColor(BG6, cv2.COLOR_BGR2RGB)
BGdata7_ = cv2.cvtColor(BG7, cv2.COLOR_BGR2RGB)
BGdata8_ = cv2.cvtColor(BG8, cv2.COLOR_BGR2RGB)
BGdata9_ = cv2.cvtColor(BG9, cv2.COLOR_BGR2RGB)
BGdata10_ = cv2.cvtColor(BG10, cv2.COLOR_BGR2RGB)

# Select the red values of the data arrays
data1 = data1_[:, :, 0]
data2 = data2_[:, :, 0]
data3 = data3_[:, :, 0]
data4 = data4_[:, :, 0]
data5 = data5_[:, :, 0]
data6 = data6_[:, :, 0]
data7 = data7_[:, :, 0]
data8 = data8_[:, :, 0]
data9 = data9_[:, :, 0]
data10 = data10_[:, :, 0]
data11 = data11_[:, :, 0]

BGdata1 = BGdata1_[:, :, 0]
BGdata2 = BGdata2_[:, :, 0]
BGdata3 = BGdata3_[:, :, 0]
BGdata4 = BGdata4_[:, :, 0]
BGdata5 = BGdata5_[:, :, 0]
BGdata6 = BGdata6_[:, :, 0]
BGdata7 = BGdata7_[:, :, 0]
BGdata8 = BGdata8_[:, :, 0]
BGdata9 = BGdata9_[:, :, 0]
BGdata10 = BGdata10_[:, :, 0]

"""
Lines 85-125 are for the multi-exposure slice images
"""

# Take the mean of each row and collapsing into Nx1 array
means = np.mean(data1, 1)
# Now just get the index of the largest mean
idx = np.argmax(means)

# Plot slices to show values are as expected, NB ~ no background
NB_data_slice1 = np.transpose(data1[idx:idx + 1, :] - BGdata1[idx:idx + 1, :])
NB_data_slice2 = np.transpose(data2[idx:idx + 1, :] - BGdata2[idx:idx + 1, :])
NB_data_slice3 = np.transpose(data3[idx:idx + 1, :] - BGdata3[idx:idx + 1, :])
NB_data_slice4 = np.transpose(data4[idx:idx + 1, :] - BGdata4[idx:idx + 1, :])
NB_data_slice5 = np.transpose(data5[idx:idx + 1, :] - BGdata5[idx:idx + 1, :])
NB_data_slice6 = np.transpose(data6[idx:idx + 1, :] - BGdata6[idx:idx + 1, :])
NB_data_slice7 = np.transpose(data7[idx:idx + 1, :] - BGdata7[idx:idx + 1, :])
NB_data_slice8 = np.transpose(data8[idx:idx + 1, :] - BGdata8[idx:idx + 1, :])
NB_data_slice9 = np.transpose(data9[idx:idx + 1, :] - BGdata9[idx:idx + 1, :])
NB_data_slice10= np.transpose(data10[idx:idx + 1,:] - BGdata10[idx:idx + 1, :])
```

```

fig1, ax1 = plt.subplots()
ax1.plot(NB_data_slice1, label='1/4000 SS')
ax1.plot(NB_data_slice2, label='1/2000 SS')
ax1.plot(NB_data_slice3, label='1/1000 SS')
ax1.plot(NB_data_slice4, label='1/500 SS')
ax1.plot(NB_data_slice5, label='1/250 SS')
ax1.plot(NB_data_slice6, label='1/125 SS')
ax1.plot(NB_data_slice7, label='1/60 SS')
ax1.plot(NB_data_slice8, label='1/30 SS')
ax1.plot(NB_data_slice9, label='1/15 SS')
ax1.plot(NB_data_slice10, label='1/8 SS')
ax1.legend(loc="upper right")
ax1.set_yscale('log')
ax1.set_xlabel("X-Pixel Slice")
ax1.set_ylabel("Intensity [log(I)]")

fig2, ax2 = plt.subplots()
ax2.plot(NB_data_slice1, label='1/4000 SS')
ax2.plot(NB_data_slice5, label='1/250 SS')
ax2.plot(NB_data_slice10, label='1/8 SS')
ax2.legend(loc="upper right")
ax2.set_yscale('log')
ax2.set_xlabel("X-Pixel Slice")
ax2.set_ylabel("Intensity [log(I)]")
```

8.5.2. Single Slit Combination Algorithm

```
import numpy as np
import matplotlib.pyplot as plt
import cv2
import time

t_start = time.time()

"""
Lines 12-43 are initial image importing and image readiness
"""

# Import images using openCV
image1 = cv2.imread("Slit_ISO200_SS1_4000.TIF", -1)
image5 = cv2.imread("Slit_ISO200_SS1_250.TIF", -1)
image10 = cv2.imread("Slit_ISO200_SS1_8.TIF", -1)

#Import background
b1 = cv2.imread("BG_200_4000.TIF", -1)
b5 = cv2.imread("BG_200_250.TIF", -1)
b10 = cv2.imread("BG_200_8.TIF", -1)

# Convert to RGB
data1_ = cv2.cvtColor(image1, cv2.COLOR_BGR2RGB)
data5_ = cv2.cvtColor(image5, cv2.COLOR_BGR2RGB)
data10_ = cv2.cvtColor(image10, cv2.COLOR_BGR2RGB)

b1_ = cv2.cvtColor(b1, cv2.COLOR_BGR2RGB)
b5_ = cv2.cvtColor(b5, cv2.COLOR_BGR2RGB)
```

```
b10_ = cv2.cvtColor(b10, cv2.COLOR_BGR2RGB)

# Select the red values of the data arrays
data1 = data1_[:, :, 0]
data5 = data5_[:, :, 0]
data10 = data10_[:, :, 0]

back1 = b1_[:, :, 0]
back5 = b5_[:, :, 0]
back10 = b10_[:, :, 0]

#Remove background
im1 = data1 - back1*1.0
im5 = data5 - back5*1.0
im10 = data10 - back10*1.0

"""
Lines 49-69 are the combination algorithm
"""

# Set array conditions
top_limit = 20000.0
bottom_limit = 100.0

lim10 = np.where(im10>(top_limit), np.nan, im10)
lim5 = np.where(np.logical_or(im5>(top_limit), im5<(bottom_limit)), np.nan, im5)
lim1 = np.where(im1<bottom_limit, np.nan, im1)

# Scale all of the data to highest exposure
fin1 = lim1 *(4000/8.0)
fin5 = lim5 *(250/8.0)
fin10 = lim10 *(8/8.0)

# Take mean of arrays
R_arr = [fin1,fin5,fin10]
R_mean = np.nanmean(R_arr, axis=0)

# Normalise answer
normalisation = np.nanmax(R_mean)
mean_ = R_mean / normalisation
mean = np.where(np.logical_or(mean_<=0.0, mean_=='nan'), 10.0e-6, mean_)

"""
Lines 75-111 are the slit sinc squared curve fit
"""

mean2 = np.where(np.logical_or(mean_ <= 0.0, mean_ == 'nan'), 10.0e-6, mean_)

# Take the mean of each row and collapsing into Nx1 array
means = np.nanmean(mean2, 1)
# Now just get the index of the largest mean
idx = np.argmax(means, axis=0)

# Slice of data
mean_slice = np.transpose(mean2[idx:idx + 1, :])

# Theoretical Curve
```

```
a = 150E-6
lamba = 632.8E-9
dist = 75.69E-2
pixel_size = 4.3E-6
theta_conv = pixel_size / dist
k = np.pi * a / lamba

def sinc_func(x):
    k = np.pi*a/lamba
    theta_conv = pixel_size / dist
    y = (np.sinc(k*theta_conv*x))**2
    return y

old_x_pixel = np.arange(0,5183,1)
x_pixel = old_x_pixel - 700
sinc_ini = sinc_func(x_pixel)

# Plot Experimental and Theoretical curves
fig1, ax1 = plt.subplots()
ax1.plot(mean_slice, label='Experimental')
plt.plot(sinc_ini, label='Theoretical')
ax1.legend(loc='best')
ax1.set_yscale('log')
ax1.set_xlabel("X-Pixel Slice")
ax1.set_ylabel("Normalised Intensity [log(I/I_0)]")
plt.yscale('log')

"""
Lines 117-140 are for the combination algorithm images
"""

# 2-D Contour plot of full data
fig2, ax2 = plt.subplots()
im = ax2.imshow(np.log10(mean), cmap = 'jet')
ax2.set_xlabel('X-Pixel')
ax2.set_ylabel('Y-Pixel')
fig2.colorbar(im, shrink=0.7, aspect=6, label='Normalised Intensity [log(I/I_0)]')
plt.show()

# 3D log plot of the full data
X = np.arange(1,mean.shape[1]+1)
Y = np.arange(1, mean.shape[0]+1)
X, Y = np.meshgrid(X, Y)

fig3 = plt.figure()
ax3 = fig3.gca(projection='3d')
ax3.set_xlabel('X-Pixel')
ax3.set_ylabel('Y-Pixel')
ax3.set_zlabel('Normalised Intensity [log(I/I_0)]')
im = ax3.plot_surface(X , Y, np.log10(mean),cmap = 'jet')
fig3.colorbar(im, shrink=0.7, aspect=6, location='left', label='Normalised Intensity
↪  [log(I/I_0)]')
plt.show()

t_end = time.time()
print(t_end-t_start, 'seconds')
```

8.6. Artefact Identification Code

8.6.1. Etaloning Identification

```
import matplotlib.pyplot as plt
import cv2

# Opens images
image1=cv2.imread("Thermal_16_20_14_4.tiff",-1)
image2=cv2.imread("Thermal_16_50_26_4.tiff",-1)

# Converts images to arrays
data1_ = cv2.cvtColor(image1,cv2.COLOR_BGR2RGB)
data2_ = cv2.cvtColor(image2,cv2.COLOR_BGR2RGB)

# Isolates R values of RGB component
data1 = data1_[1500:1900,2600:3100,0]
data2 = data2_[1500:1900,2600:3100,0]

# Shows image with colorbar
fig, (ax1, ax2) = plt.subplots(1, 2)
fig.suptitle('Etaloning Investigation')
ax1.set_xlabel("X Pixel")
ax1.set_ylabel("Y Pixel")
ax1.imshow(data1, cmap = 'bone')
ax2.set_xlabel("X Pixel")
ax2.imshow(data2, cmap = 'bone')
```

8.6.2. Dust Identification

```
import matplotlib.pyplot as plt
import cv2

# Opens images
image1=cv2.imread("Dust_Spot1.tiff",-1)
image2=cv2.imread("Dust_Spot2.tiff",-1)

# Converts images to arrays
data1_ = cv2.cvtColor(image1,cv2.COLOR_BGR2RGB)
data2_ = cv2.cvtColor(image2,cv2.COLOR_BGR2RGB)

# Isolates R values of RGB component
data1 = data1_[700:1200,2600:3500,0]
data2 = data2_[700:1200,2600:3500,0]

# Shows image with colorbar
fig, (ax1, ax2) = plt.subplots(1, 2)
fig.suptitle('Dust Investigation')
ax1.set_xlabel("X Pixel")
ax1.set_ylabel("Y Pixel")
ax1.imshow(data1)
ax2.set_xlabel("X Pixel")
ax2.imshow(data2)
```