

The Fraunhofer Diffraction

Mito Funatsu ^{*1}, Jinsheng Li ^{†1}, and Yiwei Yu ^{‡1}

¹Department of Physics, University of California, Berkeley

May 3, 2024

1 Abstract

We studied the diffraction patterns generated by 11 custom-made diffraction gratings of various shapes with a monochromatic laser. Using the Fraunhofer diffraction equation, we modeled the diffraction patterns of these diffraction gratings of different shapes. Inconsistencies between the theory and experiment are discussed, and agreements between the modeled and actual patterns are quantified.

Our results are largely consistent with the theory derived from the Fraunhofer diffraction equation. Among which, we shall discuss four regular patterns ("Cat", "Fence", "Double Triangle", and "Duck") and six fractal patterns in detail, focusing on the uncertainties introduced in modeling, experimental procedure, and data analysis.

2 Introduction

In the 17th and 18th centuries, many scientists supported Isaac Newton's "corpuscular theory of light" - the idea that light is made up of tiny particles - despite the experimental observations of Robert Hooke and the theoretical work of Christiaan Huygens. It was not until the interference and diffraction experiment of Thomas Young and the observation of the "Poisson spot" in a circular diffraction pattern that the scientific community largely discarded the corpuscular theory and accepted the wave theory of light.

This experiment probes the nature of light as electromagnetic waves by studying the diffraction patterns of monochromatic, coherent lasers. As such, light rays obey the superposition principle, where the amplitude of superposing light rays equals the sum of the amplitudes of each individual ray. Light rays that pass through a diffraction grating interfere and superpose at a plane of projection, forming a pattern which can be modeled by a mathematical expression.

^{*}mito.funatsu@berkeley.edu

[†]ljs233233@berkeley.edu

[‡]yuyiwei@berkeley.edu

3 Theory

3.1 The Fraunhofer Diffraction Equation

In the theory of optics, the Fraunhofer Diffraction Equation is used to model the far-field diffraction of plane waves incident on a diffracting object. The equation reads:

$$E(\vec{k}) = \iint_{\mathbb{R}^2} e^{-i\vec{k}\cdot\vec{r}} A(\vec{r}) E(\vec{r}) d\vec{r} \quad (1)$$

Which relates the electromagnetic wave in the position and reciprocal spaces; where $A(\vec{r})$ is the Aperture Function:

$$A(\vec{r}) = \begin{cases} 0, & \text{For when light is Transmitted} \\ 1, & \text{For when light is Obstructed} \end{cases} \quad (2)$$

We can see $E(\vec{k})$ is thus the Fourier Transformation of $A(\vec{r})E(\vec{r})$.

$$E(\vec{k}) = \mathcal{F}[A(\vec{r})E(\vec{r})] \quad (3)$$

The intensity of the light ray that we observe is given by the magnitude of the Poynting vector:

$$I = \langle S \rangle = \frac{1}{2} c \epsilon_0 E^2 \quad (4)$$

Where $c = 3.00 \times 10^8 \text{ ms}^{-1}$ is the speed of light, and $\epsilon_0 = 8.854 \times 10^{-12} \text{ Fm}^{-1}$ is the permittivity of free space.

3.2 Validity of The Fraunhofer Regime

Theoretically, the Fraunhofer Diffraction Equation only correctly models the pattern created at an infinitely far distance from the diffraction object. Nonetheless, as the diffraction phenomenon is governed by sinusoidal functions, we deemed that the Fraunhofer diffraction equation gives satisfactory approximations as long as we can make a small-angle approximation with negligible error.

The usual approach to simplify a single slit diffraction pattern is to approximate $\sin\theta$ by $\tan\theta$ for the angular size of the diffraction pattern as viewed from the aperture. The relative error of such approximation is:

$$\delta = \left| \frac{\tan\theta - \sin\theta}{\sin\theta} \right| \quad (5)$$

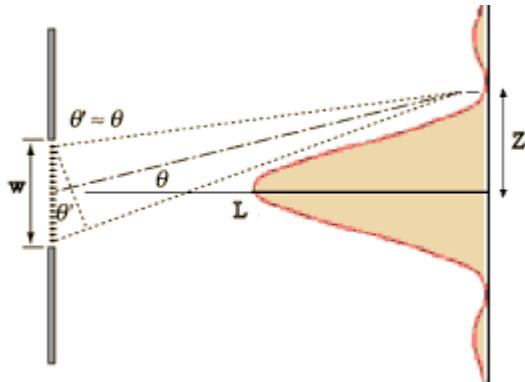


Figure 1: An illustration of the single slit diffraction pattern

An angle of $\theta = 0.14 \text{ rad} = 8^\circ$ would yield an error of $\delta = 1\%$. Under a conservative approximation, a diffraction pattern of radius 3 cm projected at a distance of 100 cm spans 0.03 radians of the view cone, far below the threshold of generating even a 1% error. We thus conclude that the errors from far-field approximation are negligible and do not cause a visible difference.

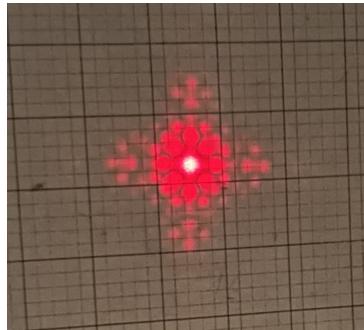


Figure 2: Sample diffraction pattern with underlying grid

4 Methods

4.1 Diffraction Pattern Preparation

To conduct the experiment, various 2D slit patterns were first prepared on transparent photo slides. The patterns were designed on Adobe Illustrator using vector graphics to create non-pixelated patterns (refer to Figures 3 and 4 for all of the patterns used for analysis). The white sections of the pattern were "printed" transparently, which became the 2D slits. Then, they were printed on 24mm x 36mm slides with a resolution of 2891 DPI.

4.1 Diffraction Pattern Preparation

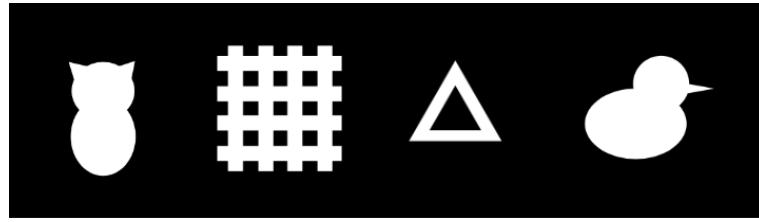


Figure 3: The patterns from the first round of experiment used for analysis. The exact measurements of the slit sizes are unknown, as the images were resized by the photo company to be in the range of 0.3~0.35mm. From the left, we will refer to them as "Cat," "Fence," "Double Triangle," and "Duck".

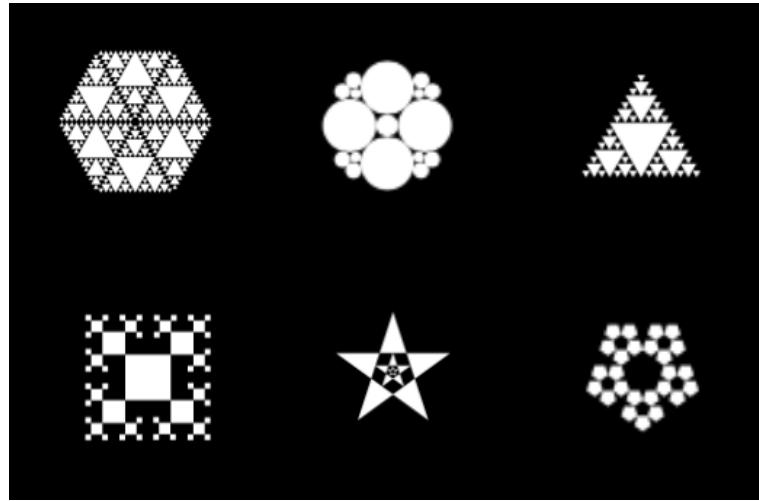


Figure 4: The fractal patterns from the second round of experiment used for analysis. The designs will be referred to as 1, 2, 3, 4, 5, 6, reading from the top left to top right, then bottom left to bottom right. The exact slit sizes are known for these patterns and are taken into account in the analysis.

There were two main concerns with the sizing of the slit: ensuring that its overall shape is smaller than the spot size of the laser, and designing the *smallest* slit to be *bigger* than the finite resolution of the printer so as not to have the design get "washed out." The prior part was ensured by taking a few initial measurements using an optical bench (1000 mm) with a screen and laser mounted on top. The laser was placed 5 ± 0.05 cm from the screen and shone directly on it. The spot size of the laser was then measured using a caliper a few times, with the resulting measurement of 0.15 ± 0.005 cm. This meant that the biggest slit size should be comparably smaller than this, given the diffraction pattern will be placed around where the screen is. Refer to Figure 5 for a sketch of this setup.

4.1 Diffraction Pattern Preparation

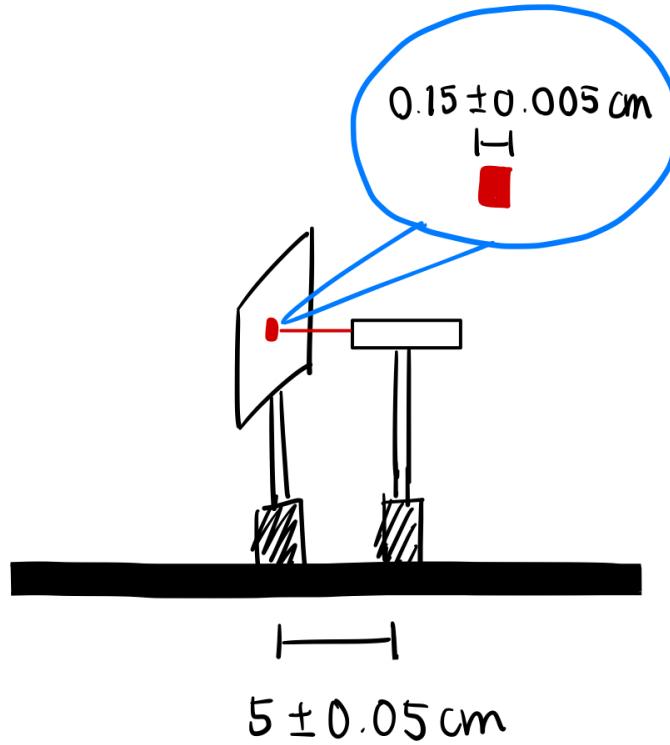


Figure 5: Initial measurement of the laser spot size.

For the latter concern, given a 2891 DPI, the resolution yielded 113.8 pixels/mm. This meant that each pixel is ~ 0.00879 mm in length, so all of our designs were made to be bigger than this. However, some patterns with intricate "sub-patterns" contain slits as small as 0.028 mm (namely the fractal patterns,) just about three times as big as the pixel limit. This, combined with the fact that many patterns don't follow along the vertical and horizontal lines of the pixels, inevitably created a more pronounced pixelated quality. As discussed in the "discussion" section later, this may have affected the results to some scale.

On another note, the slit size must be small enough to be in the Fraunhofer regime, namely:

$$D \gg \frac{a^2}{\lambda}$$

where D is the distance from the slit to the screen, a is the slit size, and λ is the wavelength of the laser. This was however not a major concern after the largest slit size across all slit patterns was chosen to be 0.35 mm, since the equation can be satisfied by varying the distance D as needed.

4.2 Modeling

For each diffraction pattern we prepared, we modeled the diffraction patterns that would result from a monochromatic and coherent laser using the formulae outlined in the Theory section. This was accomplished primarily with Python's NumPy library. Specifically, each diffraction pattern was read and stored in an array of 1 and 0, as defined by the aperture function.

The output of each diffraction pattern is generated using Numpy's Fast Fourier Transform (FFT) module, whose square is proportional to the intensity of the observable light ray. We then plotted each diffraction pattern next to its projection pattern predicted from the model, using the Matplotlib library.



Figure 6: Diffraction Grating of "Cat"

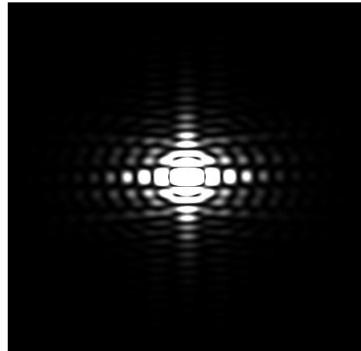


Figure 7: Diffraction Pattern of "Cat"

We must also note the inconsistencies of the NumPy FFT algorithm with the actual Fourier Transform function. For a 1-dimensional array of length n , FFT is defined such that positive frequency values are returned in indices $[n//2, n)$, and positive frequency values are returned in indices $[0, n//2)$. In other words, for a 1-D array, the left and right halves of the output array are inverted, folding the values at the edges into the center, and vice versa. Generalizing this pattern to 2-dimensional arrays,

the x and y coordinates of each quadrant of the FFT output are essentially inverted. To revert this effect and retrieve the results consistent with physical theory, we employed the following algorithm to unfold the FFT output and compute the intensity output.

The *FAST* Fourier Transform

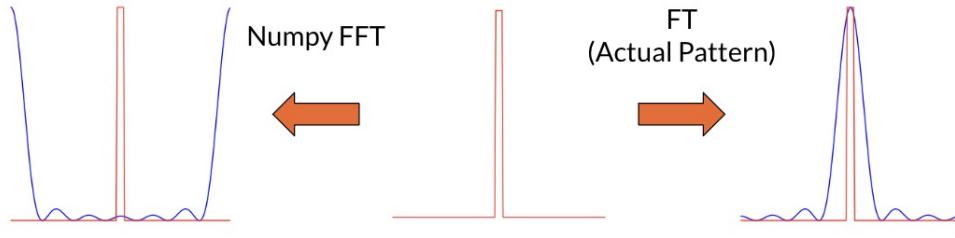


Figure 8: A 1-dimensional illustration of the inconsistencies of the NumPy Fast Fourier Transform (left) versus the actual Fourier Transform pattern (right).

Algorithm 1 Fraunhofer Diffraction Modeling

```

Input: Aperture                                ▷ The Aperture Function Array
Output: Intensity                            ▷ The Normalized Intensity Output
function Diffraction(Aperture)
    ft_real := abs(FFT2D(Aperture)) ** 2          ▷ Squared magnitude of the 2D Numpy FFT
    ft_real_norm := ft_real / sqrt(sum(ft_real ** 2)) ▷ Normalized ft_real
    Intensity := ZeroArray(shape = shape(ft_real_norm))
                                                ▷ Initialize array of zeroes, with dimension equal to ft_real_norm

    length := length(ft_real_norm)
    width := width(ft_real_norm)                   ▷ Length and Width of the output array
    for START i := 0; END i = length; STEP i += 1 do
        for START j := 0; END j = width; STEP j += 1 do
            Intensity[ i ][ j ] = ft_real_norm[ (i + length // 2) % length ][ (j + width // 2) % width ]
        end for
    end for                                     ▷ Unfolds the result of NumPy FFT; Writes into the Intensity array

return Intensity
end function

```

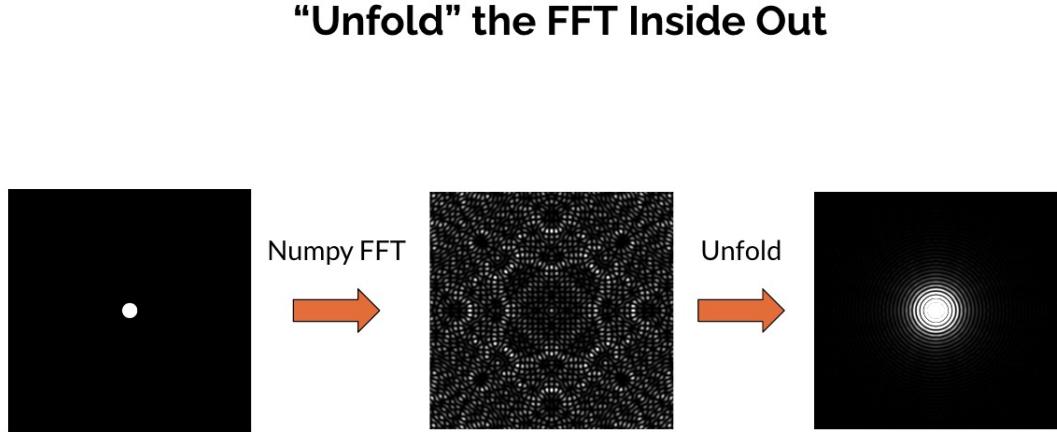


Figure 9: A 2-dimensional illustration of Algorithm 1, unfolding the NumPy Fast Fourier Transform (center) into the actual Fourier Transform pattern (right).

4.3 Experimental Procedure

The experiment was performed in a dark laboratory room setting in order to detect as much of the diffraction pattern as possible with the iPhone camera. For this experiment, the following materials are needed:

- Optical Bench, 1000 mm
- Slides holders (4)
- Red diode laser pointer ($\lambda = 634.6 \text{ nm}$)
- iPhone camera
- Phone holder (must fit in slide holder)
- White screen
- Slit patterns

First, mount the white screen and the laser onto opposite sides of the bench and measure the distance. For this particular experiment, the distance was $62 \pm 0.05 \text{ cm}$. Then, mount the slit pattern onto another slide holder, and place it just in front of the laser. The orientation of the pattern should be so that the pattern used for theoretical modeling faces the laser, and the horizontal edges are parallel to the table. Shine the laser through the 2D slit, and observe the pattern on the screen. Now, mount

4.3 Experimental Procedure

the iPhone onto a slide holder, and place it as close to the bench as possible, and as far away from the screen as possible. This minimizes the angle error (discussed in more detail in the next paragraph). Lock the focus at the pattern, and take at least 3 pictures. After every patterns are taken care of, take picture of a ruler lined up against the screen. This will determine the pixel to length conversion later. Refer to Figure 10 for a picture of the general setup.

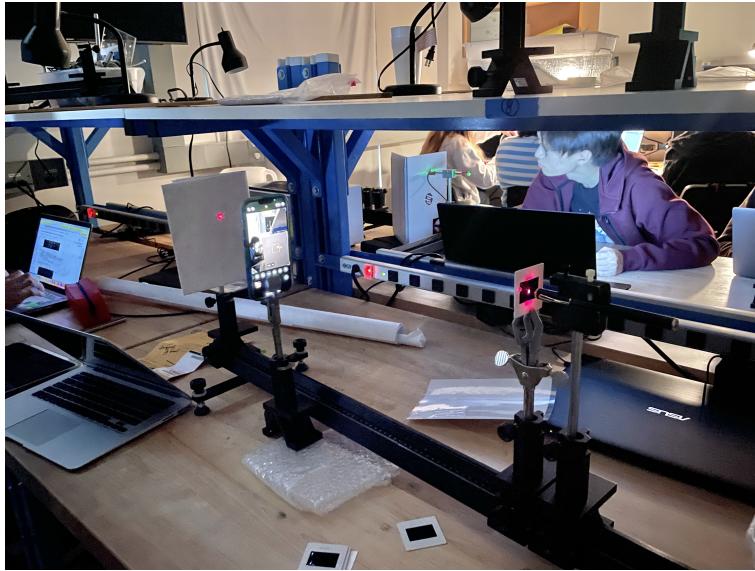


Figure 10: The experiment follows this same general setup. The diffraction pattern is placed so that its straight edges are parallel to the table. The iPhone is placed as close as possible to the bench to minimize the "tilt" error. The bubble wrap beneath the phone stand is there to correct the tilt in the phone holder and level the iPhone so that it is parallel to the screen.

When capturing the images with the phone camera, it is impossible to place the iPhone along the optical bench and capture the pattern head on, as it blocks the light from the laser. Therefore, it was inevitably placed at some angle from the pattern, albeit as close as possible to the path of the laser. In order to calculate the error produced from the angle of the iPhone camera, the "diagonal" iPhone to screen distance was measured, as well as the length from the screen to the position of the iPhone along the optical bench. Refer to Figure 11 for the exact measurements and a visual representation.

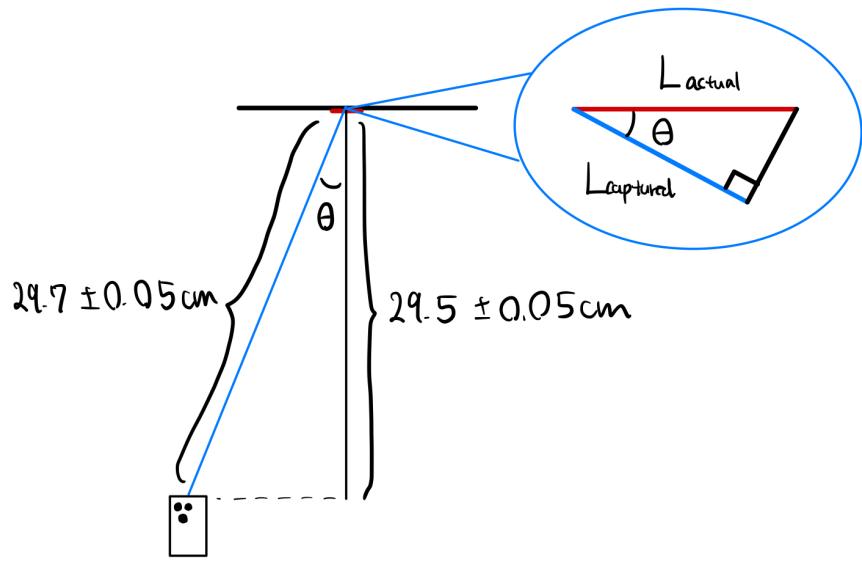


Figure 11: Here, the actual length (denoted as L_{actual}) can be calculated from the perceived length in the photo (denoted as $L_{captured}$) by $L_{actual} = \frac{L_{captured}}{\cos\theta}$, where θ is the angle between the optical bench (specifically along the line of the laser) and the imaginary line from the iPhone camera to the diffraction pattern. By geometry, the angle is identical to that of the zoomed in section.

From this geometric configuration, $\theta = \sec(\frac{L_{captured}}{L_{actual}}) = 0.116\text{rad}$, and $\cos(\theta) \approx 0.993$. Given this value, the ratio $\frac{L_{captured}}{L_{actual}} = 0.993 \approx 1$, yielding a less than 1% error. Therefore, the error in the position of the patterns from the angle were approximated to be negligible in the data analysis.

5 Analysis

5.1 Photo Processing

In order to perform an analysis on our image data, we need to process it to abstract our desired pattern. We first load the image with grayscale. The image is loaded as an array of grayscale values where each represent the intensity of the pattern.

5.1 Photo Processing

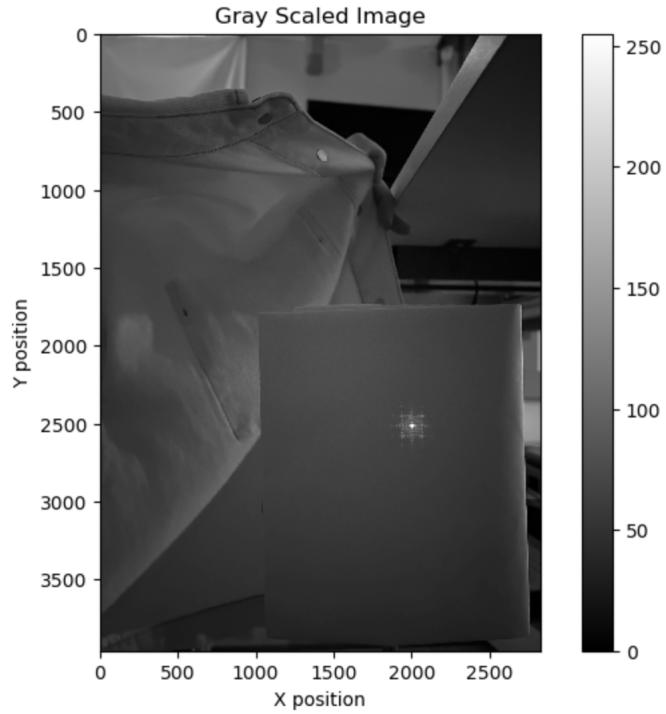


Figure 12: Example photo loaded with grayscale

Next, we need to crop out our region of interest that contains only our desired pattern and make sure it is centered at the central maximum of the pattern. We locate the central maximum by finding the pixel point with the highest intensity. If there are multiple points with the highest intensity, we locate the central maximum by finding the center of gravity among these points.

5.2 Aligning experimental image with theoretical patterns

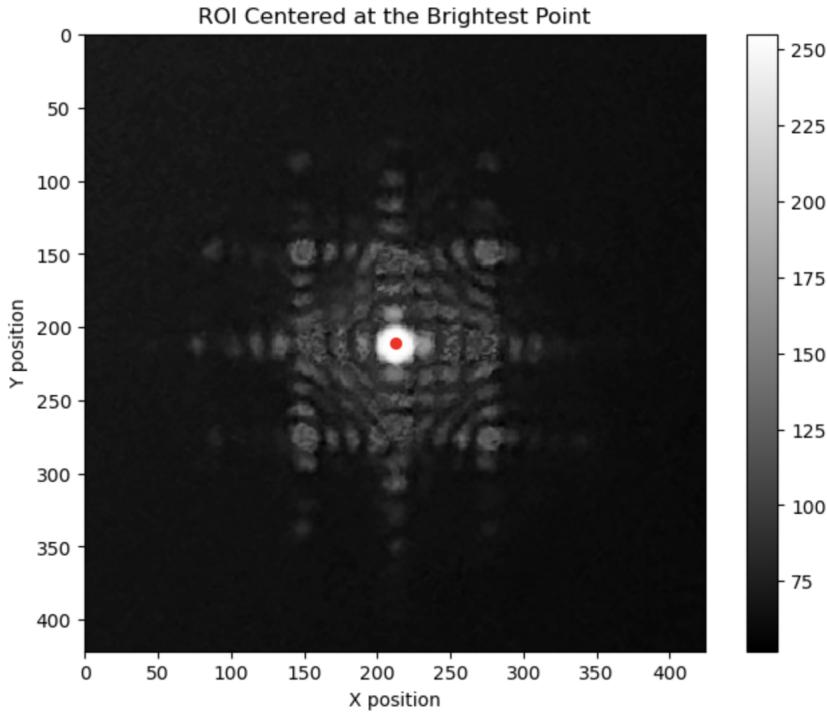


Figure 13: Example of cropped out and centered pattern

In our analysis, the intensity of the screen should be zero and all light on the screen is the ambient light from the environment. Hence, to get rid of the ambient light intensity from the environment, we subtract the intensity of the screen from the pattern, effectively setting the intensity of the screen to 0.

5.2 Aligning experimental image with theoretical patterns

In order to perform a fit, we need to align the experimental image with theoretical patterns. For our first batch of slits, we did not have the accurate measurement of the slit size, hence we identify a reference area and fit the image accordingly. The steps are described below:

1. Find a reference area for alignment.
2. Find the width and length ratio of the reference area relative to the pattern image generated by Fourier transform by counting pixels.
3. Reframe our camera image so that the reference area takes up the same ratio of pixels.
4. By doing this, each pixel on our image represents the same pixel on the generated pattern.

The figure below shows an example of pattern with aligned image:

5.2 Aligning experimental image with theoretical patterns

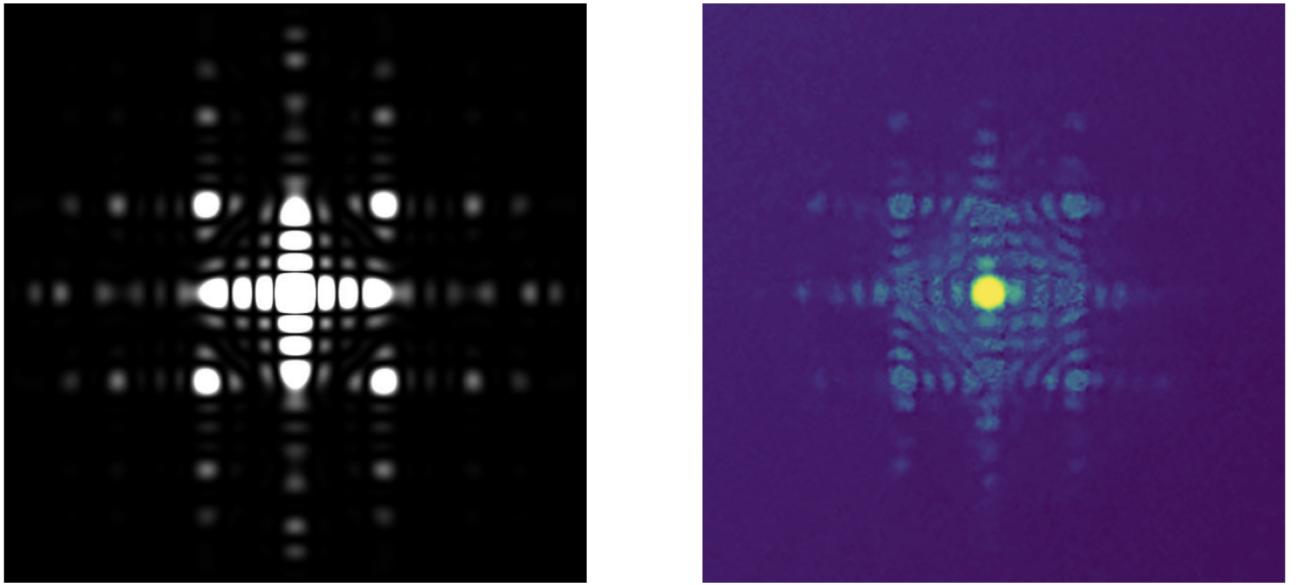


Figure 14: Example of pattern with aligned image (unknown slit size)

For fractal patterns we do have accurate measurements of the slit size, hence we perform the image alignment with the following steps:

1. Find the pixel to millimeter conversion rate r_1 in our slit picture.
2. Load the slit picture and perform fourier transform. Find the length and width of the resulting diffraction pattern in millimeter using pattern width/height = number of pixels $\times r_1$.
3. When take the photo of the diffraction image in experiment, we place the screen $d=62\text{cm}$ away from the slit. Hence, we need to find the magnification of the pattern using $M = \frac{\lambda d}{a}$, where a is the slit size.
4. By taking a reference picture using the same camera setting, we find out that the pixel to millimeter conversion rate for our camera images are $r_2 = 11.3 \text{ pixels/mm}$. Hence we use image width/height = pattern width/height $\times M \times r_2$ to find the relevant size of the camera image pattern in terms of pixels. We crop out the relevant region spanning from the center maximum of the pattern.
5. We then resize the theoretical pattern and our camera image to have the same size. By doing this, each pixel on our camera image should represent the same pixel on the generated pattern.

5.3 Agreement Test

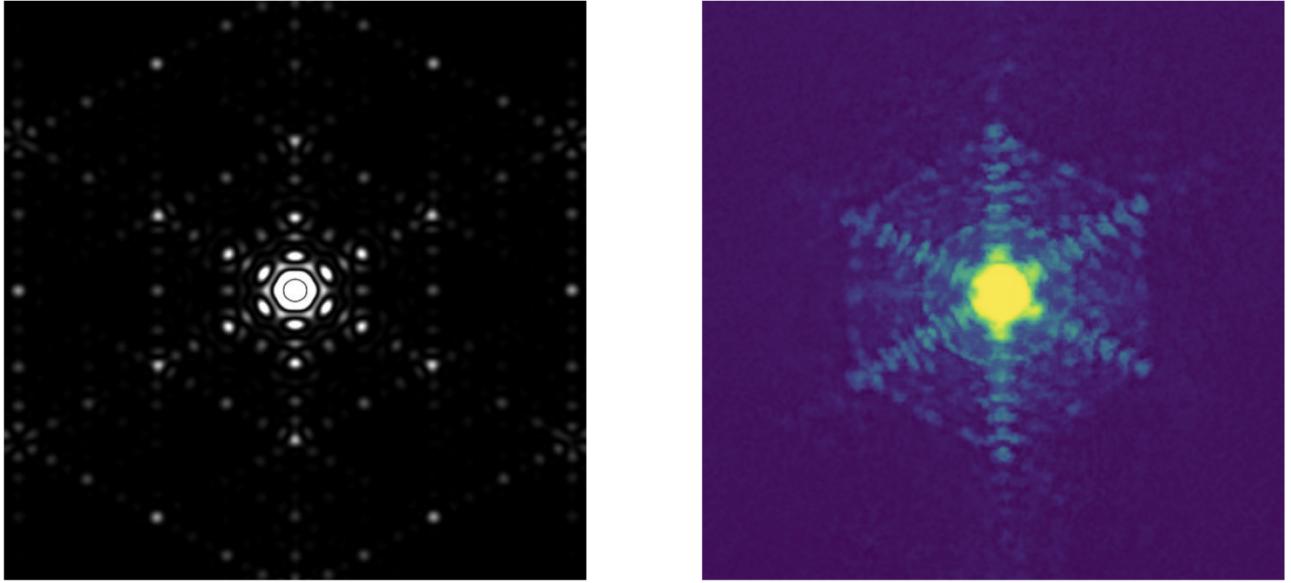


Figure 15: Example of pattern with aligned image (known slit size)

5.3 Agreement Test

In order to perform the agreement test, we first need to normalize our data to the same scale. We choose to normalize our graph so that the sum of all pixels points is equal to 1. This approach helps avoid situations where an excessively high intensity skews the scale.

$$I_{jk} = \frac{I_{jk}}{\sum I_{jk}}$$

, where I_{jk} is the normalized intensity at pixel point row j column k.

Next, We measure the divergence between the two graphs using the relative entropy

$$K = \sum_{jk} p_{jk} \log\left[\frac{p_{jk}}{q_{jk}}\right]$$

, where p_{jk} is the theoretic normalized intensity at pixel point row j column k, and q_{jk} is the empirical normalized intensity at pixel point row j column k. We perform the relative entropy test across all theoretical patterns and our images obtained.

5.4 Error Analysis

All of our measurements involved are related to the alignment of our theoretical pattern and experimental images. However, the propagation of error in misalignment is not directly translatable to the intensity values used on entropy comparison. Hence, to estimate the errors involved in the Relative

Entropy values, we take the standard errors of the intensity arrays and propagate them into the calculation of relative entropy. Since we have achieved relatively good alignment of the patterns, this method may overestimate the actual errors involved.

$$\delta\left(\frac{p}{q}\right) = \frac{p}{q} \sqrt{\left(\frac{\delta p}{p}\right)^2 + \left(\frac{\delta q}{q}\right)^2}$$

$$\sigma_{p \log(\frac{p}{q})}^2 = (\log(\frac{p}{q})\sigma_p)^2 + (p\delta(\log(\frac{p}{q})))^2$$

$$\sigma_{total}^2 = \sum_i \sigma_{p_i \log(\frac{p_i}{q_i})}^2$$

$$Error = \sqrt{\sigma_{total}^2}$$

6 Results

Tables and figures below show the relative entropy value with error for our selected patterns.
For simple slit patterns:

Image / Pattern	Cat Pattern	Double Triangle Pattern	Duck Pattern	Fence Pattern
Cat Image	1.129 ± 0.002	1.513 ± 0.002	1.344 ± 0.002	1.342 ± 0.002
Double Triangle Image	1.231 ± 0.002	1.439 ± 0.002	1.414 ± 0.002	1.391 ± 0.002
Duck Image	1.531 ± 0.002	1.892 ± 0.002	1.643 ± 0.002	1.683 ± 0.002
Fence Image	1.321 ± 0.002	1.697 ± 0.004	1.557 ± 0.003	1.22 ± 0.03

Table 1: Relative entropy result across all theoretical patterns and experimental images for simple slits

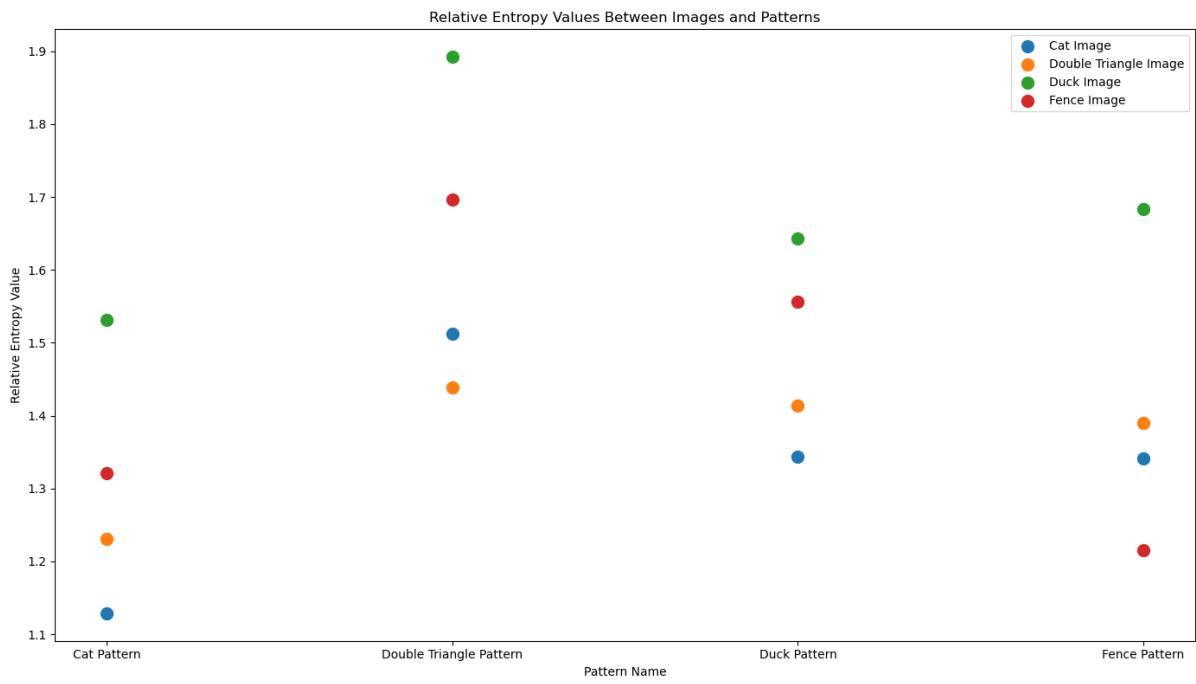


Figure 16: Relative entropy result across all theoretical patterns and experimental images for simple slits

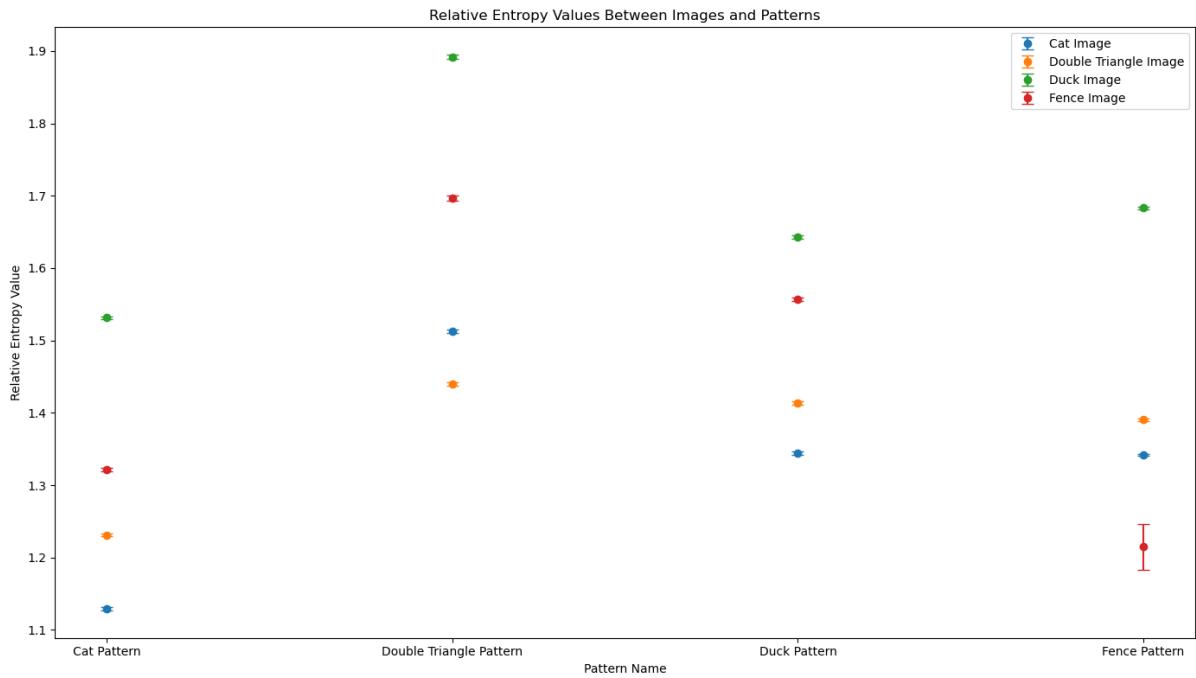


Figure 17: Relative entropy result across all theoretical patterns and experimental images for simple slits with error bar

For fractal patterns:

	Slit 1	Slit 2	Slit 3	Slit 4	Slit 5	Slit 6
Slit 1	1.209 ± 0.016	0.814 ± 0.005	0.618 ± 0.015	0.970 ± 0.040	0.633 ± 0.021	1.014 ± 0.005
Slit 2	1.451 ± 0.002	0.949 ± 0.008	0.659 ± 0.010	1.174 ± 0.032	0.925 ± 0.013	1.240 ± 0.021
Slit 3	1.445 ± 0.002	1.250 ± 0.017	0.597 ± 0.002	1.194 ± 0.001	1.024 ± 0.017	1.372 ± 0.004
Slit 4	1.428 ± 0.002	0.946 ± 0.002	0.670 ± 0.000	0.805 ± 0.002	0.725 ± 0.007	1.023 ± 0.002
Slit 5	1.387 ± 0.002	0.800 ± 0.020	0.692 ± 0.006	0.951 ± 0.025	0.566 ± 0.028	0.943 ± 0.012
Slit 6	1.494 ± 0.003	1.047 ± 0.002	0.654 ± 0.028	1.020 ± 0.016	0.768 ± 0.004	0.970 ± 0.059

Table 2: Relative entropy result across all theoretical patterns and experimental images for fractal slits

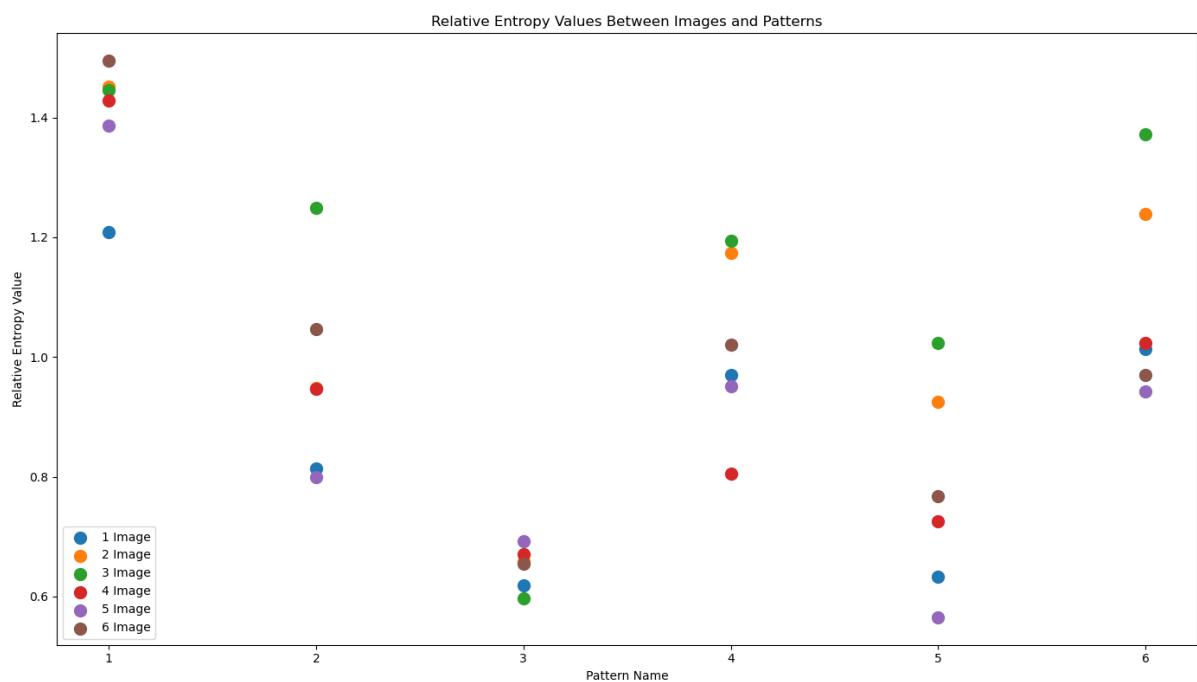


Figure 18: Relative entropy result across all theoretical patterns and experimental images for fractal slits

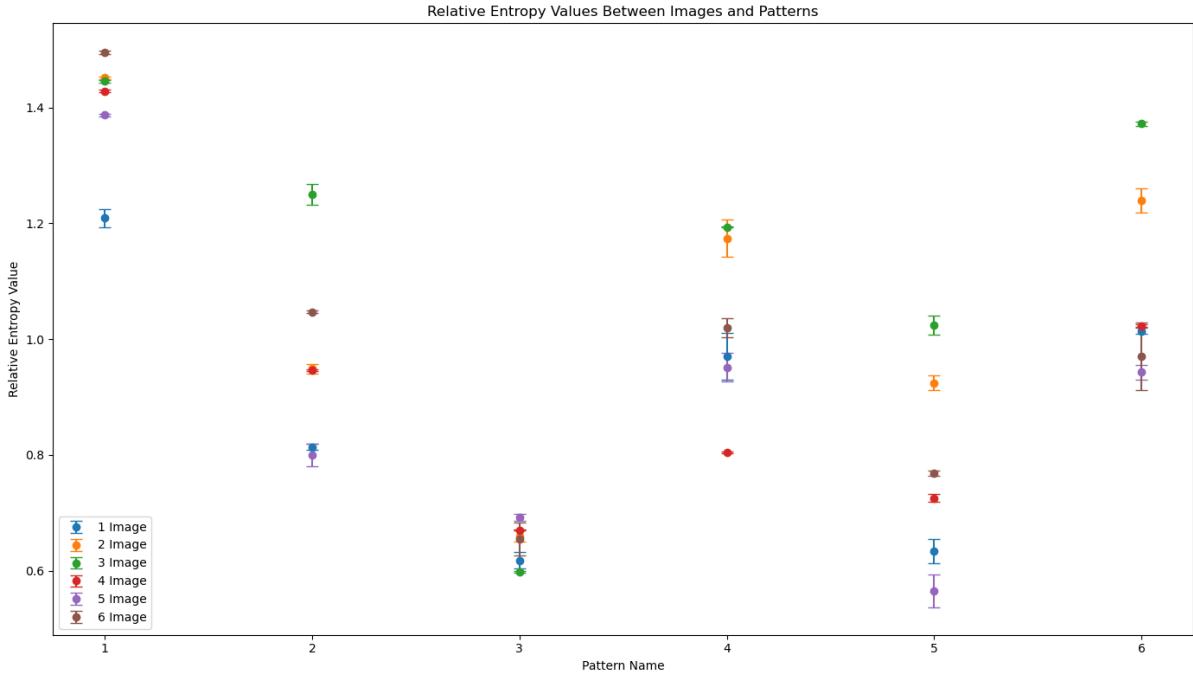


Figure 19: Relative entropy result across all theoretical patterns and experimental images for fractal slits with error bar

7 Discussion

Theoretically, the relative entropy calculated between the intensity values of an observed pattern and its theoretical pattern should yield the lowest value. In Figure 18, for each pattern labeled in the x axis, the smallest values (represented as dots in the y axis) correspond to their respective patterns except for the duck, which peculiarly has the largest relative entropy value with itself. In Figure 19, a similar trend is observed, where pattern 2 and 6 do not correspond with the pattern with which it yields the smallest relative entropy value. When we load the slit pattern for fourier transform, the slit patterns are pixelated and thus susceptible to aliasing and distortion. The fractal patterns have more fine-grained details and require better resolution, and fourier transform is sensitive to any small disturbances in input slit patterns. Hence, this could result in drastically different patterns when the loaded fractal slit images do not have high enough resolution. For the most part, the relative entropy test confirms that the observed patterns are modeled accurately by its slit pattern's Fourier Transform. At the same time, it raises questions about the experimental procedure as well as the aptness of this statistical test for our results, given a) 3 out of the 10 do not have the lowest value with their corresponding patterns and b) the lowest relative entropy value is not overwhelmingly low for the matching patterns, which would be expected.

We also noticed topics of potential interest that could be explored in future work. The grating's pixelation effects and the aperture function's edge effects were of great concern to us during the initial design and modeling process. However, during instrumentation, we failed to notice any major

issues that may have been caused by pixelation, and deemed it a less important matter. Furthermore, we also noticed that for most of our diffraction experiments, the intensity of the actual pattern falls off at a higher rate away from the central maxima than the theoretical model. It is unknown whether the former is a direct cause of the latter or whether other confounding factors are in play.

8 Conclusion

The goal of this experiment was to verify the Fraunhofer diffraction equation by modeling diffraction patterns from various 2D diffraction slits. The theoretical models were fitted to the observed diffraction patterns, which were experimentally created in the Fraunhofer regime using an optical bench setup with a red diode laser shone through the 2D slits onto a screen a distance away.

Overall, all of the diffraction patterns had overwhelming qualitative resemblance with the modeled Fourier Transform of the slit patterns, as can be seen from the alignment such as in Figures 14 and 15. Quantitatively, calculating the relative entropy of the normalized intensity readings of each pixels shows that for the most part, the respective diffraction patterns match the Fourier transform of the slit patterns. Specifically, in the first batch of patterns (cat, double triangle, duck and fence), the relative entropies were the lowest between each respective patterns except for the duck. For the fractal patterns, 4 out of the 6 patterns also had the lowest relative entropy values between each respective patterns, with design 2 and 6 being the outliers. Given that the relative entropy values are not profoundly small for the matching patterns, it points to possible error from the pixelation noise when printing slit patterns and loading images to Fourier Transform. In the future, other statistical tests could also be incorporated in order to further analyze the accuracy in the production of diffraction patterns. Additionally, an extension of this experiment includes "reverse engineering" slit patterns from various diffraction patterns by performing an inverse Fourier Transform on the diffraction patterns. This could be used to create diffraction slits that would theoretically produce a desired pattern.

9 Contributions

Theoretical Modeling: Jinsheng Li

Experimental Design: Mito Funatsu

Data Analysis: Yiwei Yu

Author Contributions:

Abstract: Jinsheng Li

Introduction: Jinsheng Li

Theory: Jinsheng Li

Methods: Mito Funatsu and Jinsheng Li

Analysis: Yiwei Yu

Results: Yiwei Yu

Discussion: Yiwei Yu, Mito Funatsu and Jinsheng Li

Conclusion: Mito Funatsu

All code used in this experiment can be found at: <https://github.com/ljs-233233/5cl-capstone>.

10 Appendix

Images for theoretical patterns and experimental result

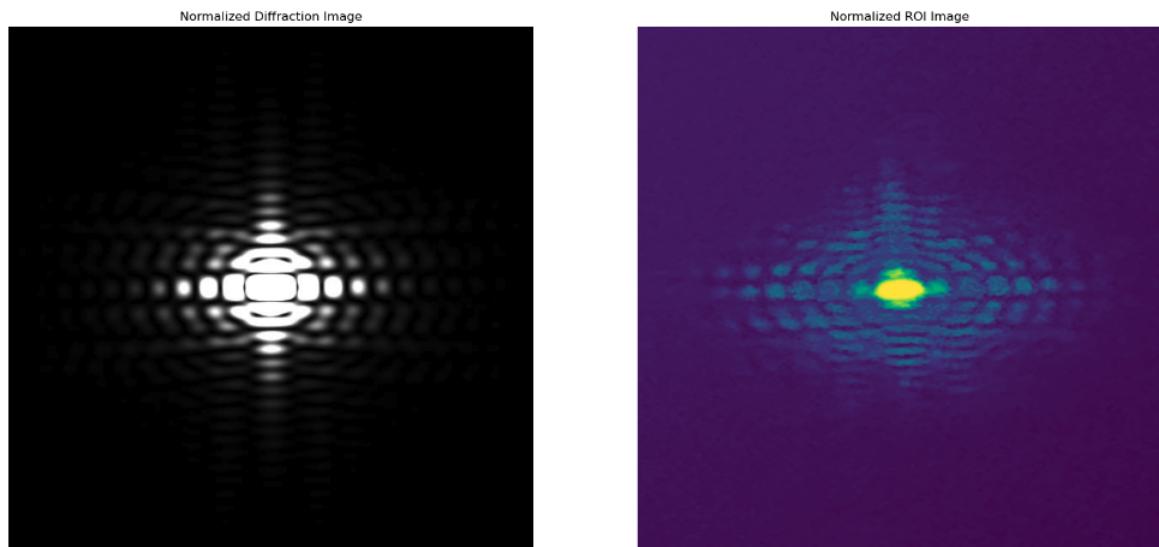


Figure 20: Simple Pattern Cat: Theoretical Patterns and Experimental Result

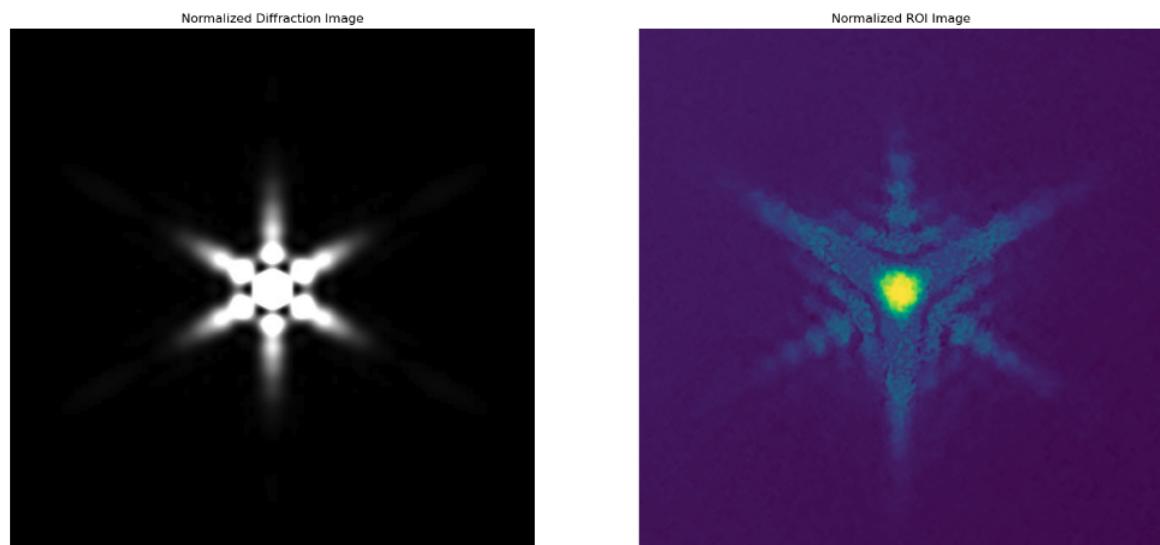


Figure 21: Simple Pattern Double Triangle: Theoretical Patterns and Experimental Result

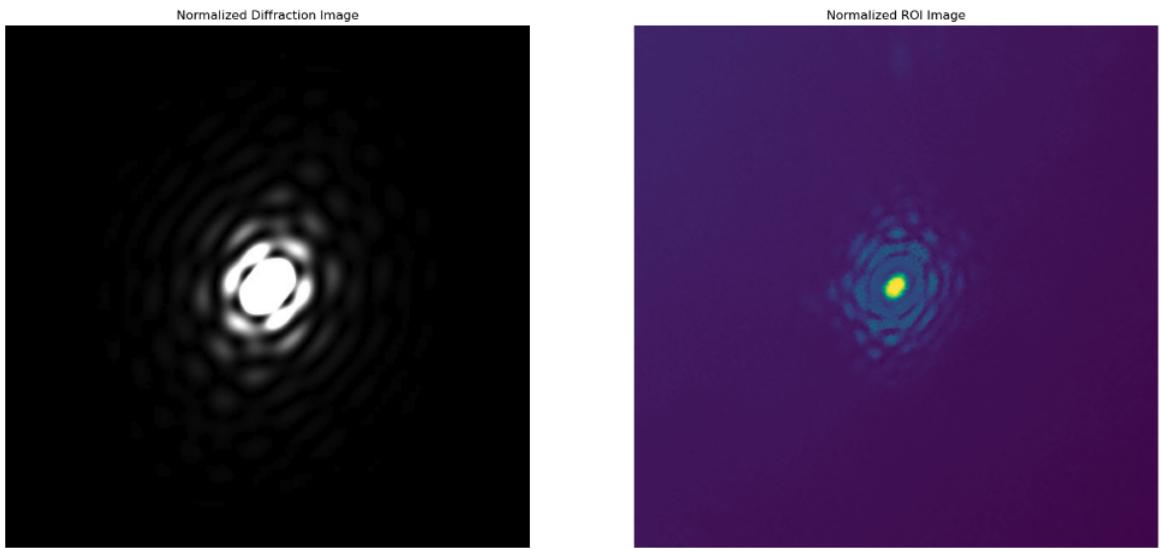


Figure 22: Simple Pattern Duck: Theoretical Patterns and Experimental Result

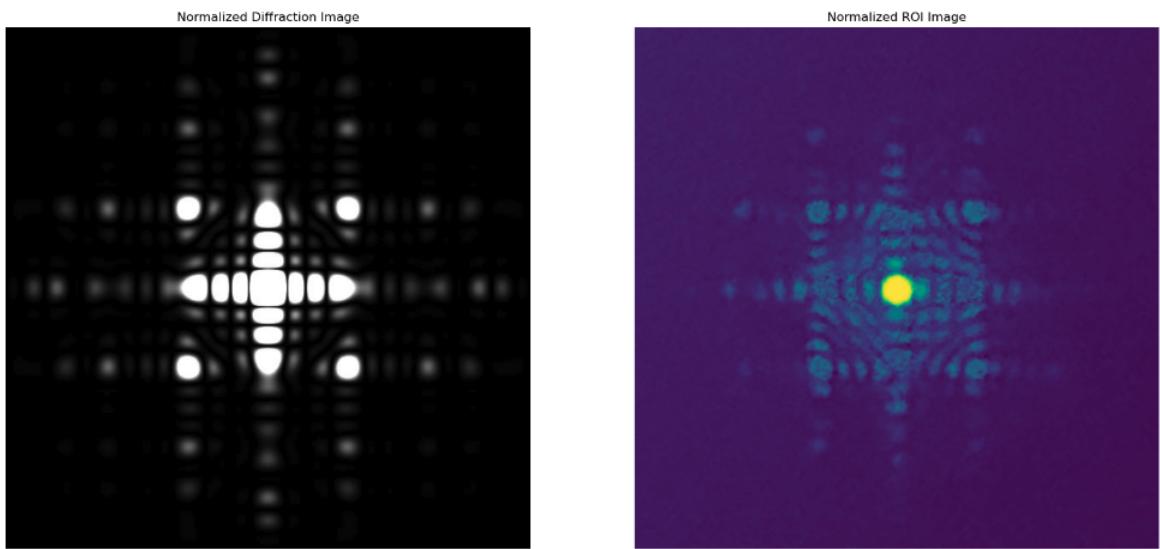


Figure 23: Simple Pattern Fence: Theoretical Patterns and Experimental Result

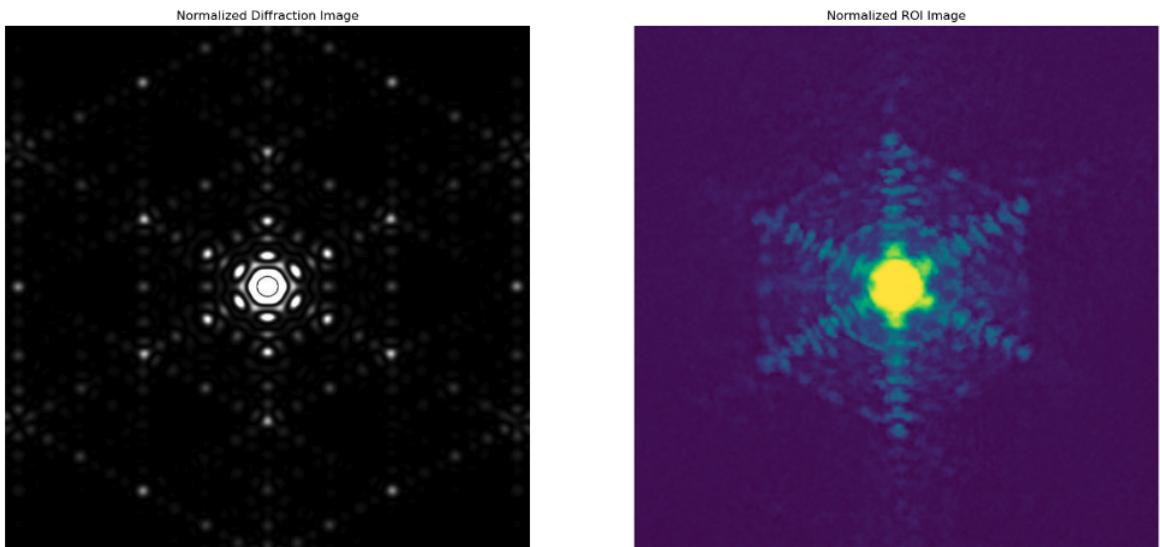


Figure 24: Fractal Pattern 1: Theoretical Patterns and Experimental Result

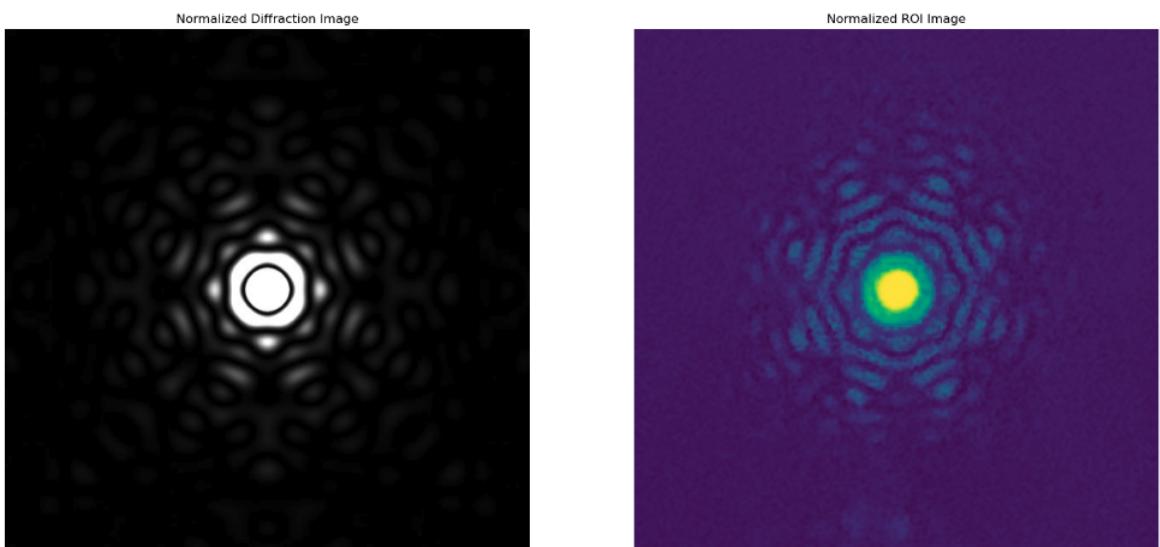


Figure 25: Fractal Pattern 2: Theoretical Patterns and Experimental Result

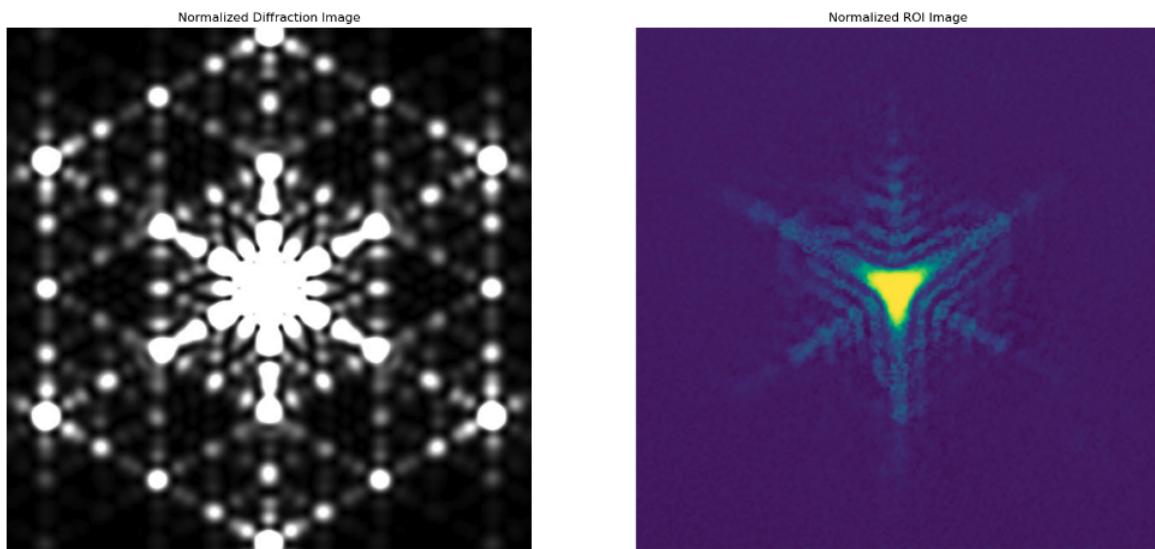


Figure 26: Fractal Pattern 3: Theoretical Patterns and Experimental Result

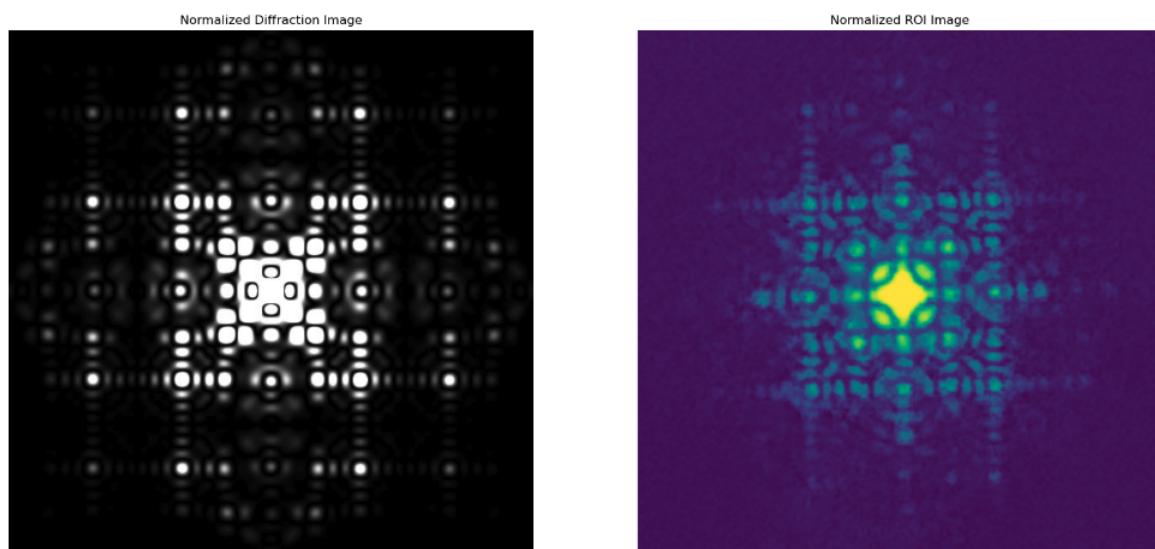


Figure 27: Fractal Pattern 4: Theoretical Patterns and Experimental Result

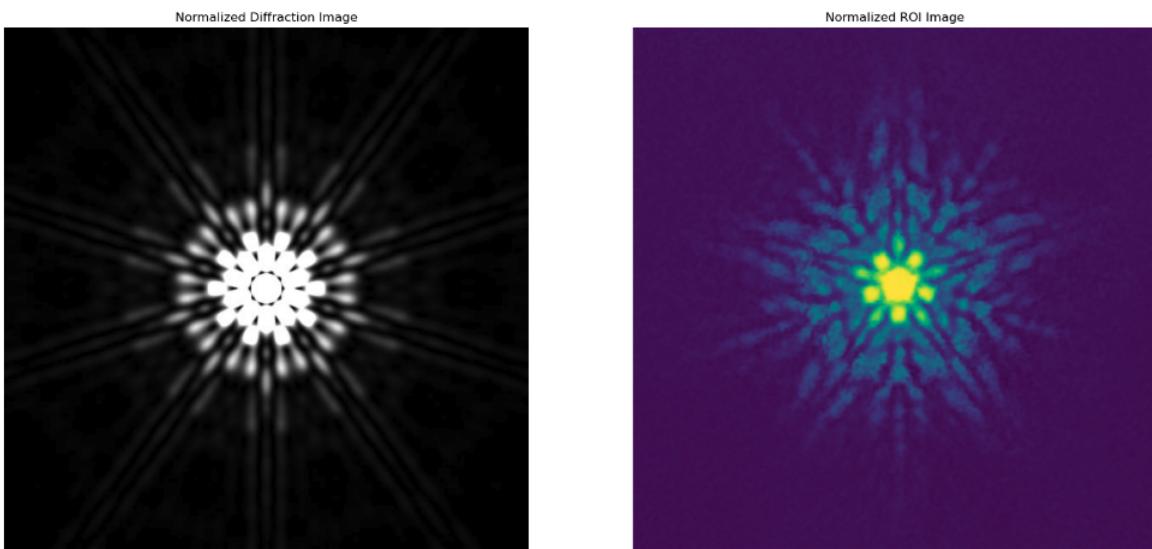


Figure 28: Fractal Pattern 5: Theoretical Patterns and Experimental Result

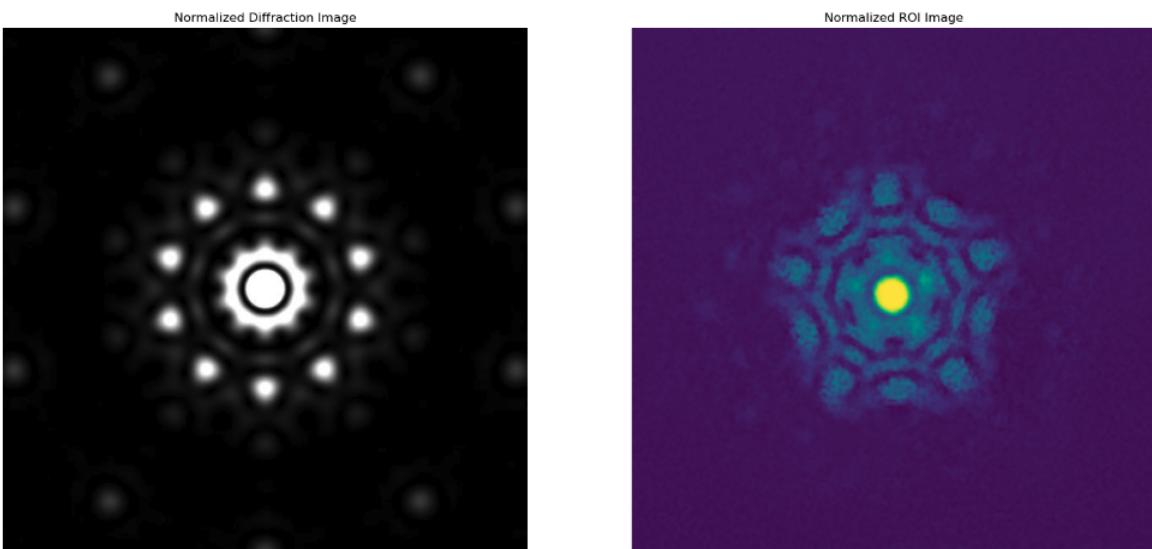


Figure 29: Fractal Pattern 6: Theoretical Patterns and Experimental Result