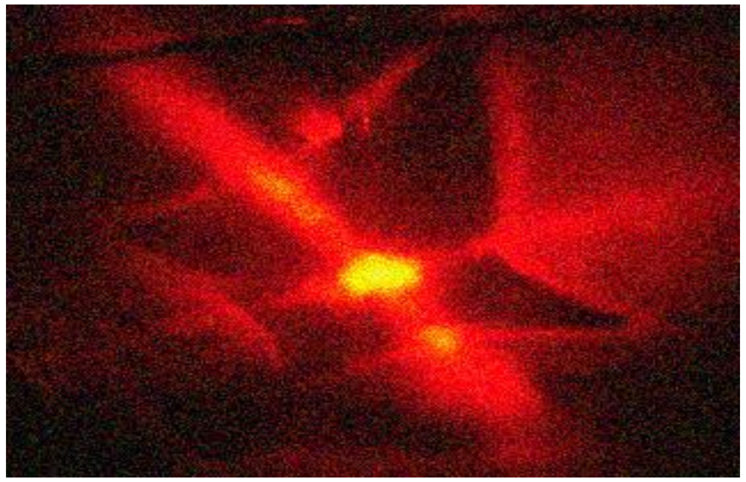
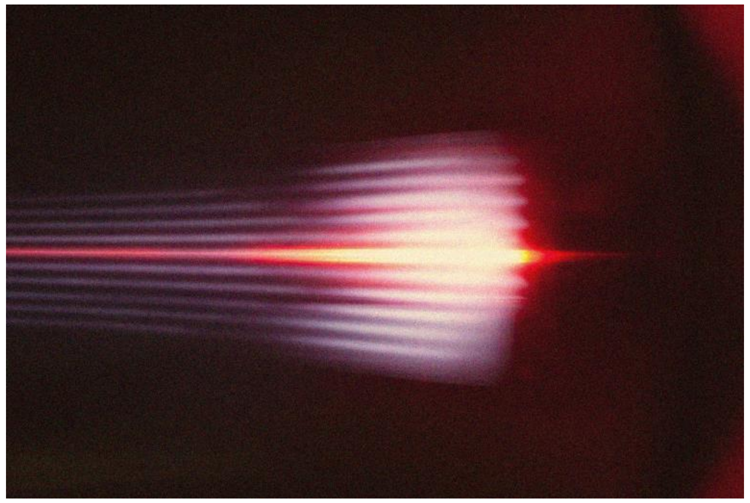
# PROPERTIES AND APPLICATIONS OF THE 2D FOURIER TRANSFORM

IMAGE AND VIDEO PROCESSING - MODULE 1



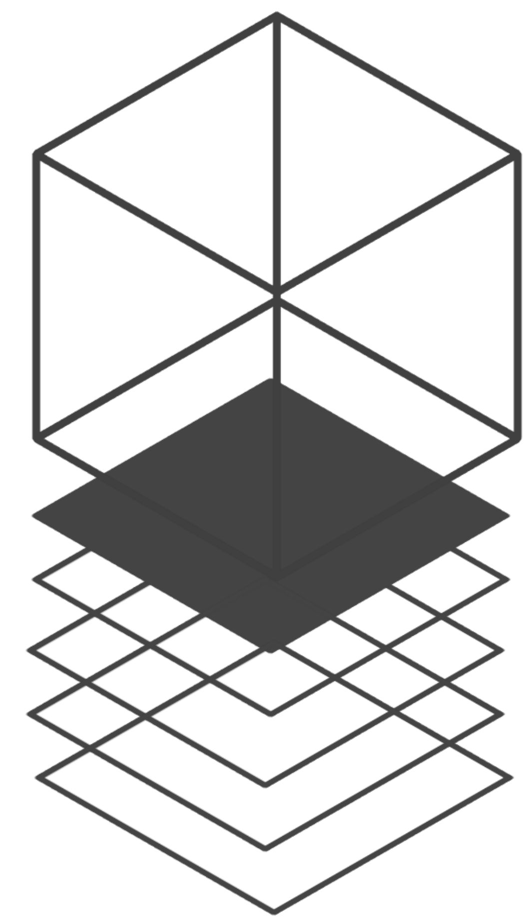




NINO PHILIP RAMONES | GITHUB
**2020 - 05616**
APRIL 21, 2023

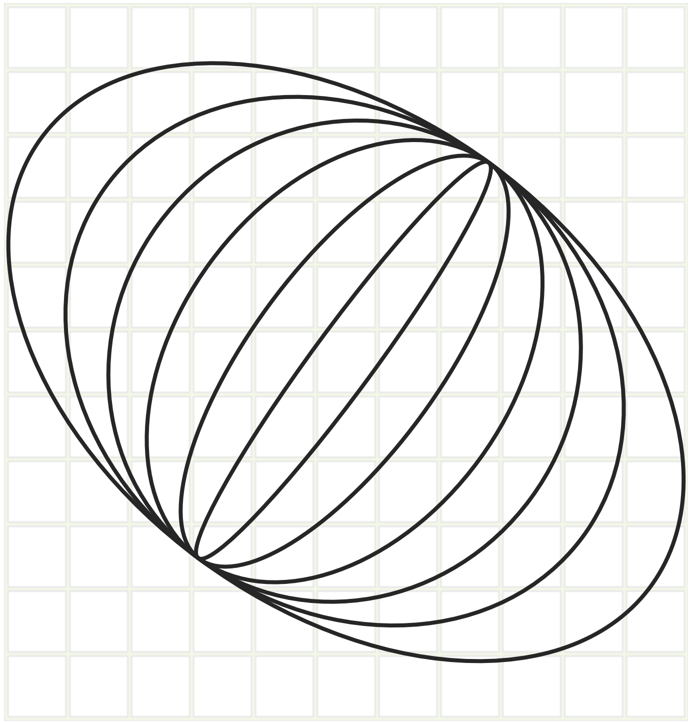
# OBJECTIVES

- Observe the rotation property of two-dimensional Fourier transform and examine how it comes into play in simulating patterns in images
- Explore some of the different applications of Fourier transform in filtering frequencies in Fourier space



# KEY TAKEAWAYS

- Similar to the previous activity, setting your **image resolution** to any powers of two will greatly speed up the calculation process when subjected to FFT algorithms and filtering
- Investigate the convolution theorem redux of some pair apertures
- The FFT algorithm comes into play when filtering unwanted frequencies or signals, which is useful in altering the appearance of images



# SOME PITFALLS

- The pattern that must be convolved must be of the **same size** as it processes both images by checking and comparing it pixel by pixel
- The choice of mask for filter affects the apperance of the resulting filtered image

# ROTATION PROPERTY OF FFT

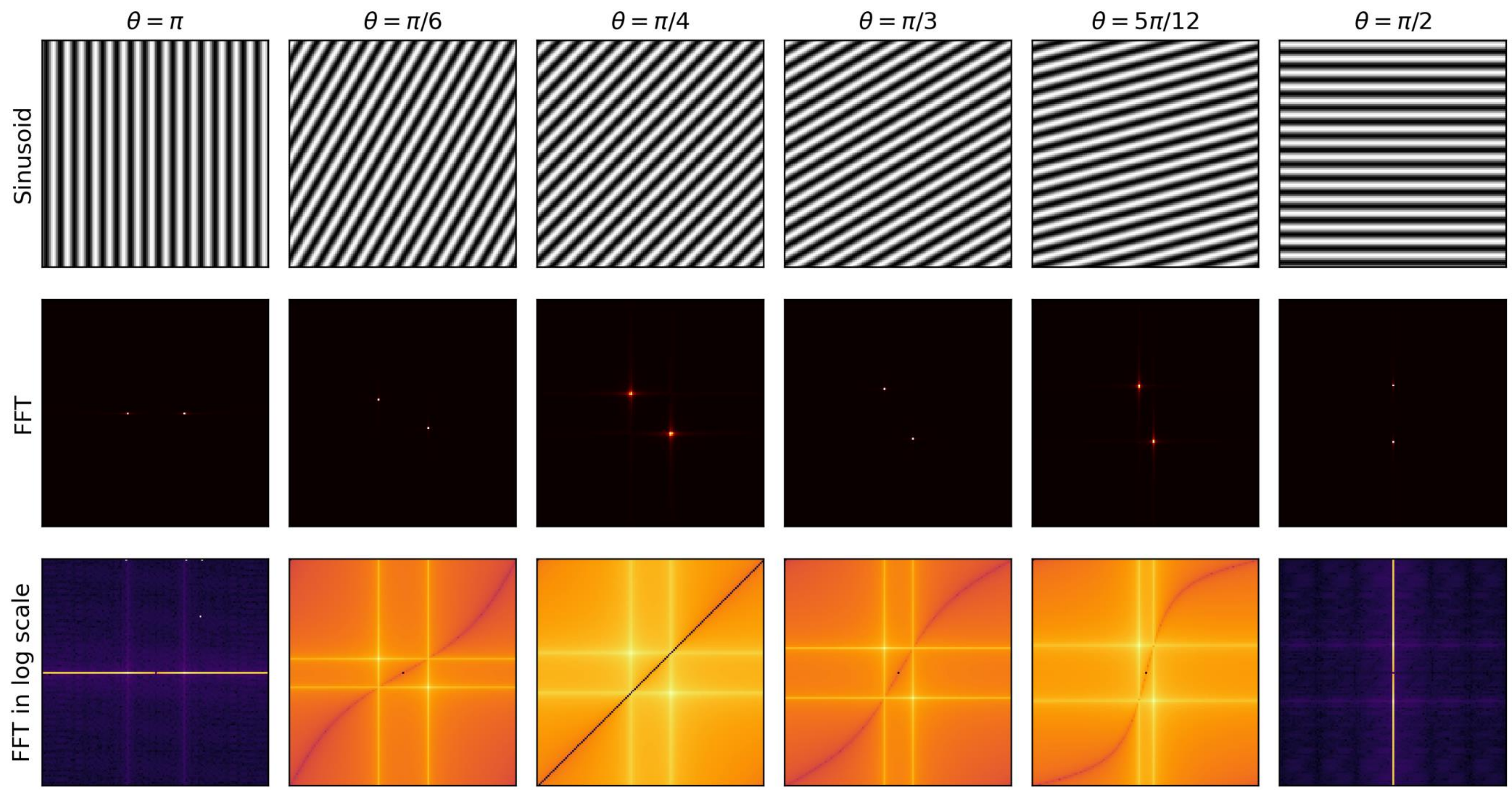



Figure 1. Rotation of sinusoids from 90 degrees to 180 degrees and their FFTs.

From the previous, the FFT a sinusoid results to a two dots or peaks in the Fourier space. The **rotation property** is unique to 2D Fourier transforms as rotating the sinusoids in the real space results to their FFTs being rotated as well. In general, the **FFTs are rotated by about 90 degrees from their orientations in the real space**, which will be of much use in the latter activities as it can be combined to simulate an imaging pattern seen in real life!

# ROTATION PROPERTY OF FFT

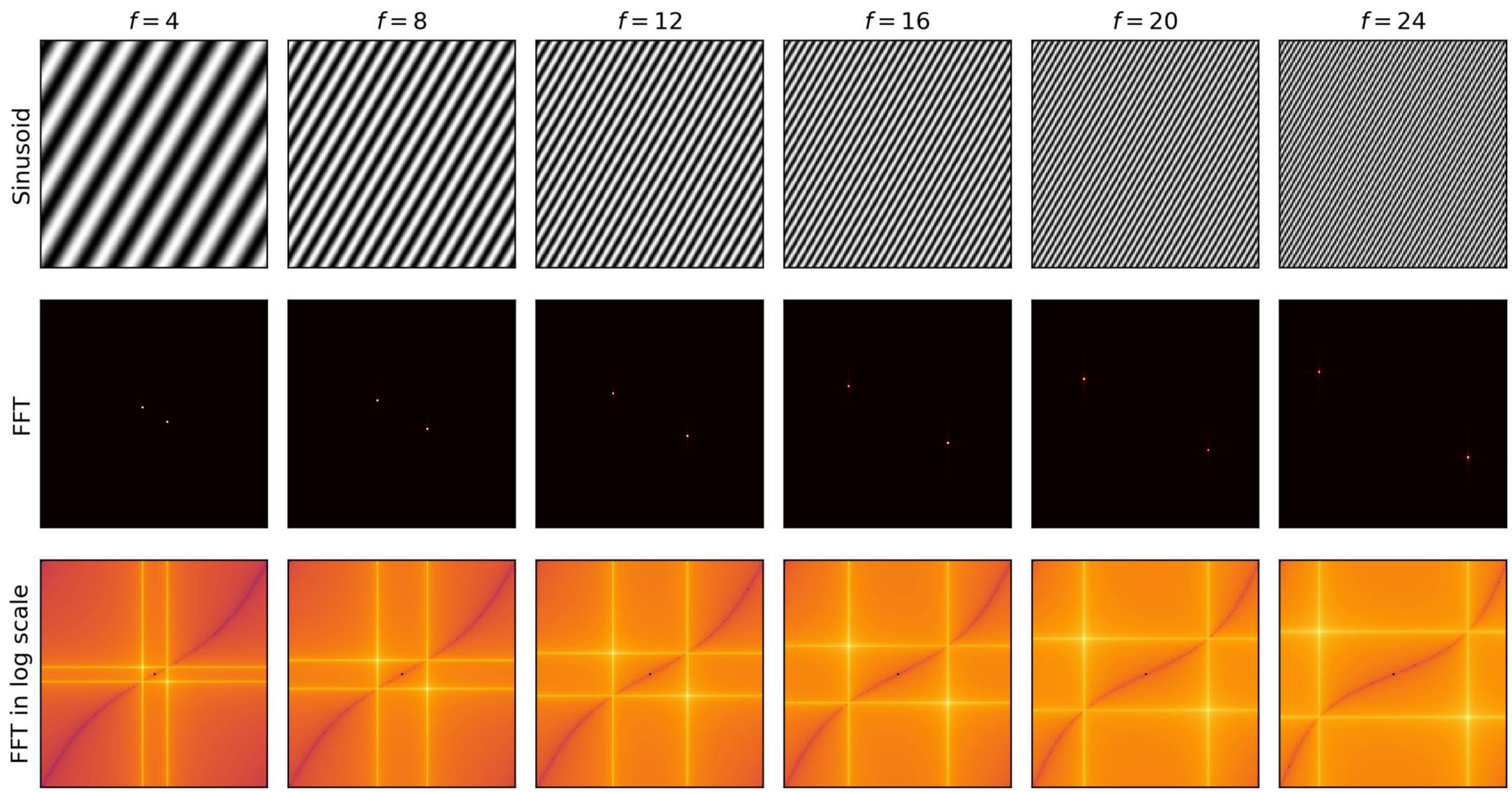



Figure 2. Sinusoids of varying frequencies and their FFTs.

Similarly, **increasing the frequencies** of a rotated sinusoid – which in this case is rotated by 45 degress – results to their FFTs being rotated as well with varying **dot spacings** in the Fourier space. It also be noted that the origin in Fourier domain represents the zero frequency by which is also symmetric on both axes. Thus, increasing the frequency in the real space will result to a **larger separation** between the two dots.

# COMBINATION OF SINUSOIDS

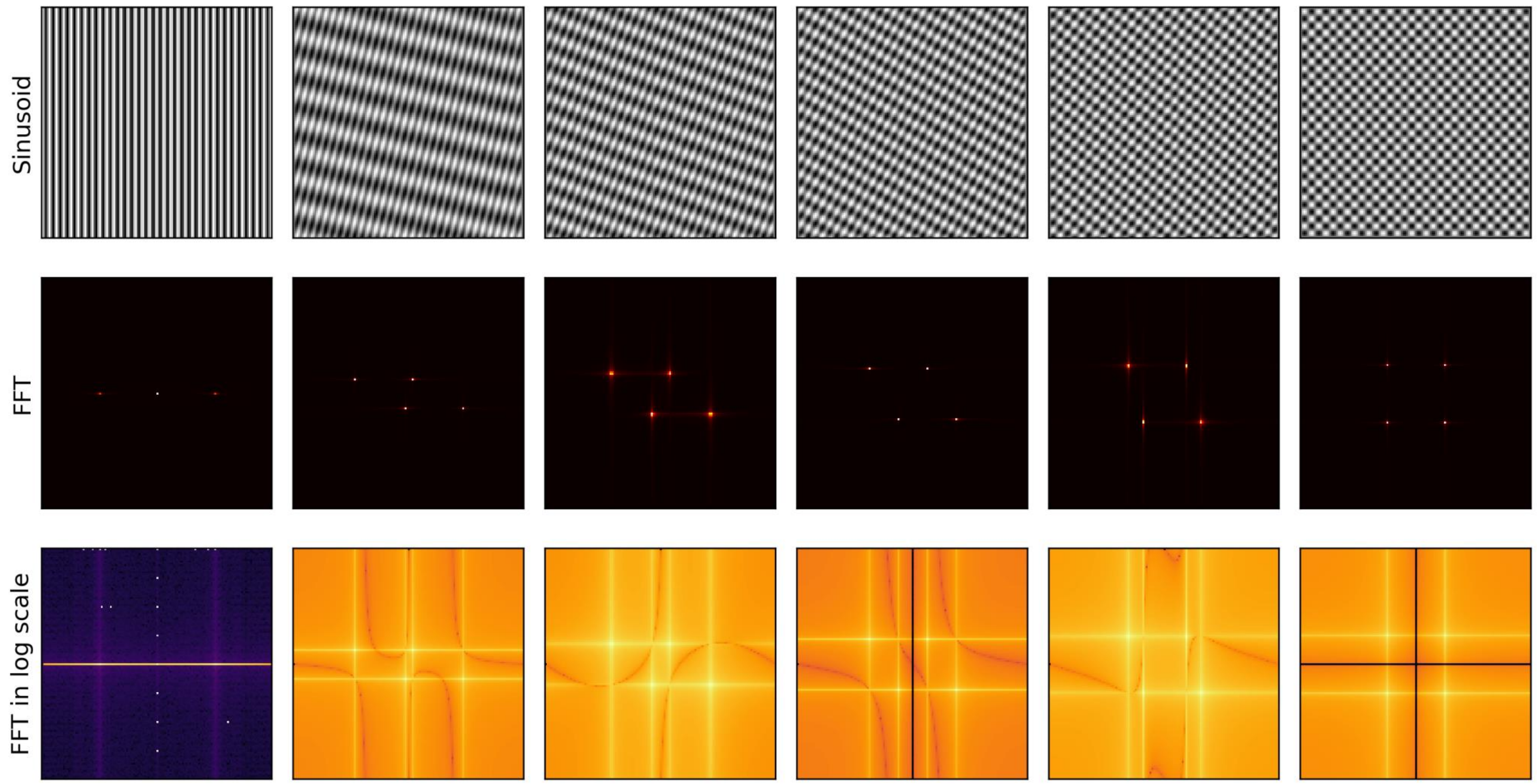



Figure 3. Product of two sinusoids at different rotation angles and their FFTs. Frequency of 4 were used for all sinusoids.

Taking the product of two sinusoids of the same frequency or taking their squares operation-wise resulted to an interesting property. Since the **product** of two sinusoids can be decomposed as **sums of sines or cosines**, the latter of which can be represented by a certain phase shift, these also deconstruct into their respective **frequencies** in the Fourier space. Hence, instead of just one pair of dots for a single sinusoid, two pairs were observed in the FFT as it represents the two sinusoids being added together.

# COMBINATION OF SINUSOIDS

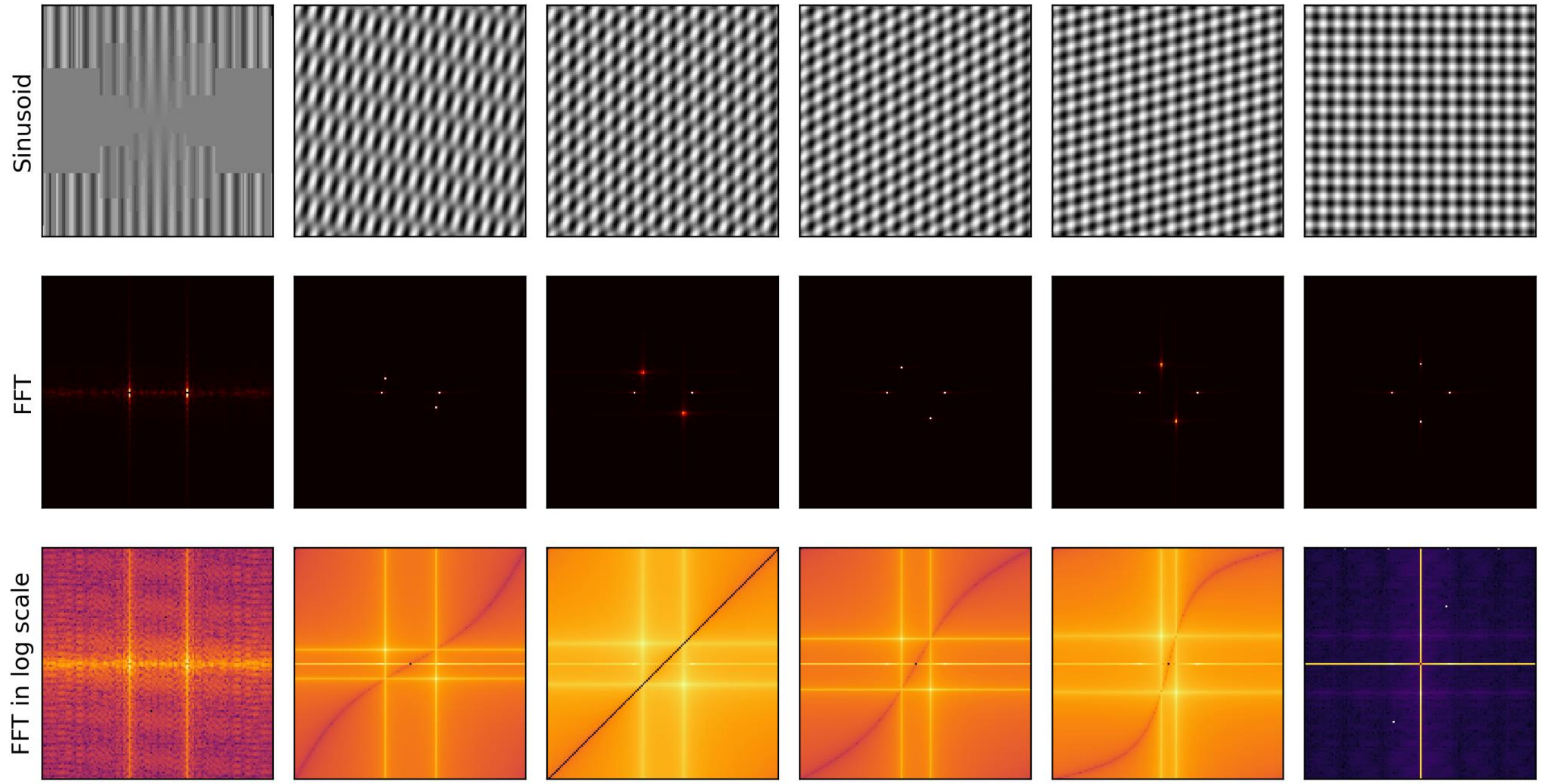



Figure 4. Sum of two sinusoids at different rotation angles and their FFTs. Frequency of 4 were used for all sinusoids.

Similar two the product of two sinusoids, taking their sums also produce a pattern that follows a similar pattern to the previous. From the figure, notice that adding sinusoids in the spatial domain results into their respective **deconstructed frequencies** in the Fourier domain. As such, it can be observed that combining sinusoids together results to a pattern that resembles a **weaving pattern** or repeating microscopic patterns! This is where the applications of FFT comes in as a rich structure can be noted from the FFT of such patterns seen in the Figure 4.

# CANVAS WEAVE MODELLING

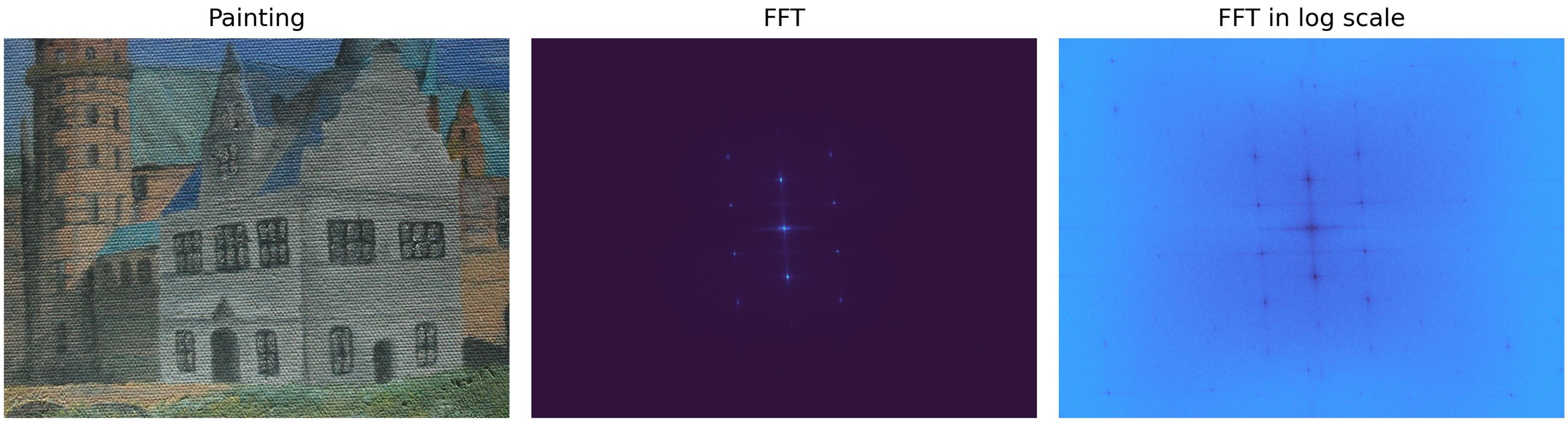



Figure 5. The original painting of interest and its FFT. Image used is the Painting by Daria for reference.

For this part of the activity, we begin by converting the painting into **grayscale** and taking its **mean value**. This was done to remove the DC bias of the image which can be ruled to affect the analysis later on. The corresponding FFT of the mean grayscale painting was carried out and we can see the peaks much more in the **logarithmic scale**. These peaks can be interpreted as the sinusoids that are combined to resemble the **weaving pattern** of the canvas itself. Since we want to remove the canvas weave and obtain a cleaner image, we can remove these unwanted patterns in the canvas by “masking” the peak frequencies in the Fourier domain.

# CANVAS WEAVE MODELLING

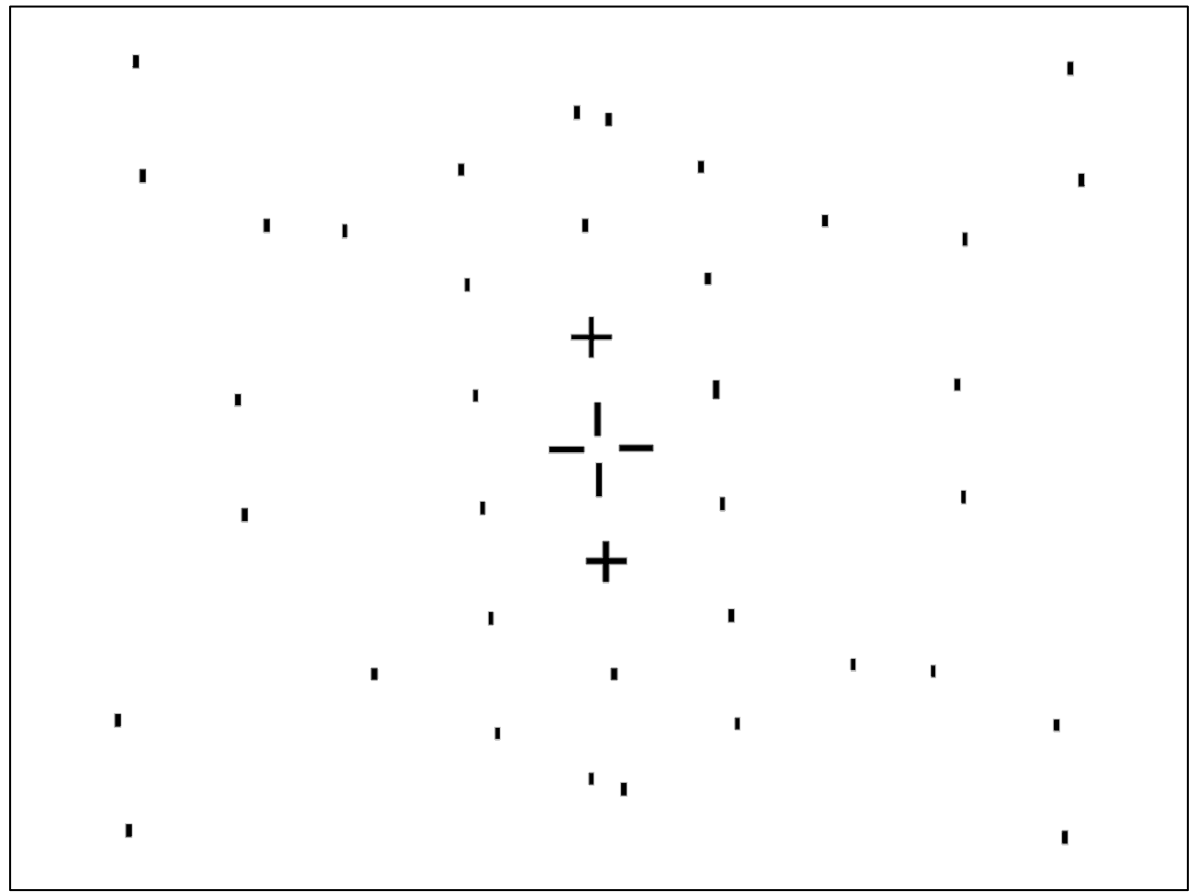


Figure 6. Filtered mask used in the activity.

After obtaining the FFT of the painting, a **filter mask** in the Fourier space was manually created. This mask is of the size as the image and is **all ones** but are **zeros** at the locations of the sinusoidal peaks. The FFT shift of the filter in Figure 6 was obtained and was multiplied to the FT of the RGB channels of the original painting image. The resulting products was subject to the invest FFT algorithm and were stacked together to resemble the **modified RGB channels** of the new image. The resulting image were expected to be much cleaner than the original and is almost free of the canvas weaving pattern!

# CANVAS WEAVE MODELLING


Figure 7. Final results after isolating the canvas patter from the original painting and white balancing the cleaned image.

Using the algorithm briefly explained in the previous slide, the **unwanted frequencies** from the canvas weaving pattern were filtered out and a much cleaner painting were obtained. The appearance of the deweaved painting could still somehow be **improved with the choice of mask**, but a stark contrast between the original and deweaved one was still obvious! In addition, the **gray world algorithm** (GWA) from the first activity was used to balance the color of the resulting image further. GWA was used since an apparent gray color was dominant on the painting – the color of the walls of the building -- by which the **average reflected color** of the painting was estimated and compared to gray color (Maulion, 2021).

# CANVAS WEAVE MODELLING

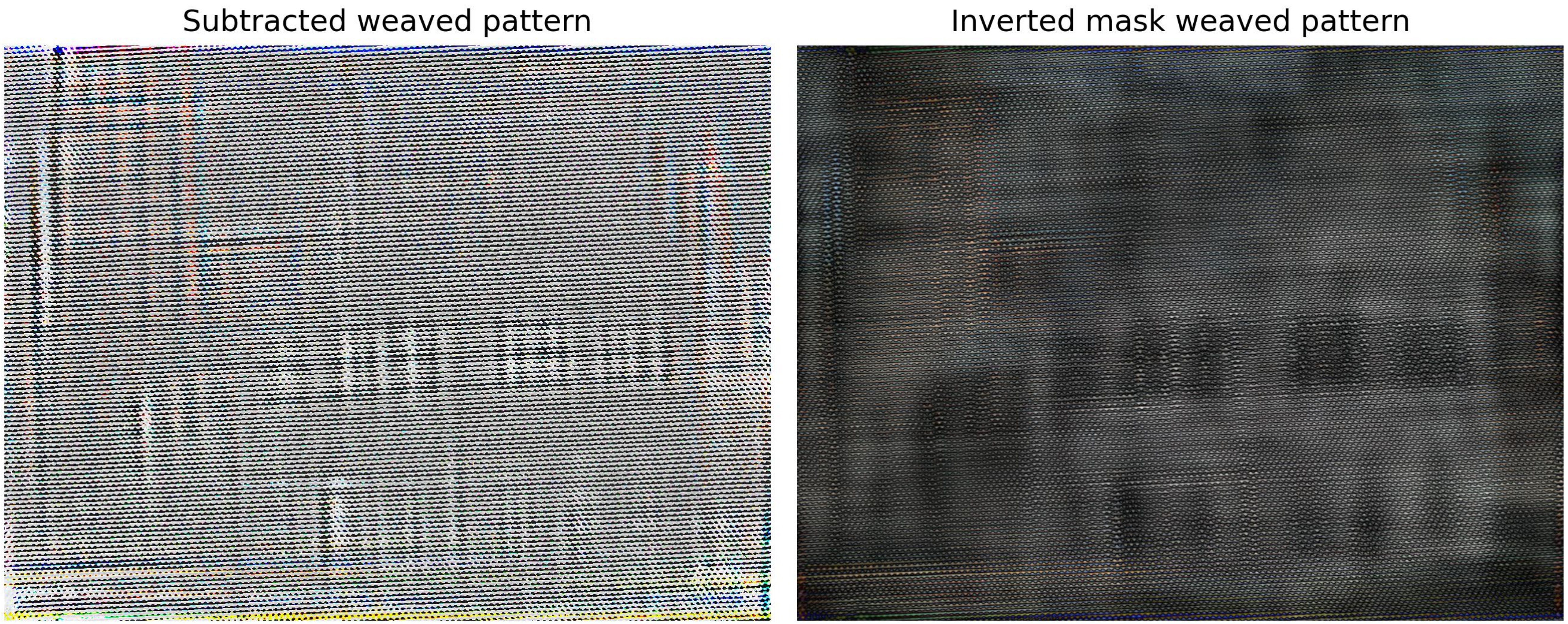
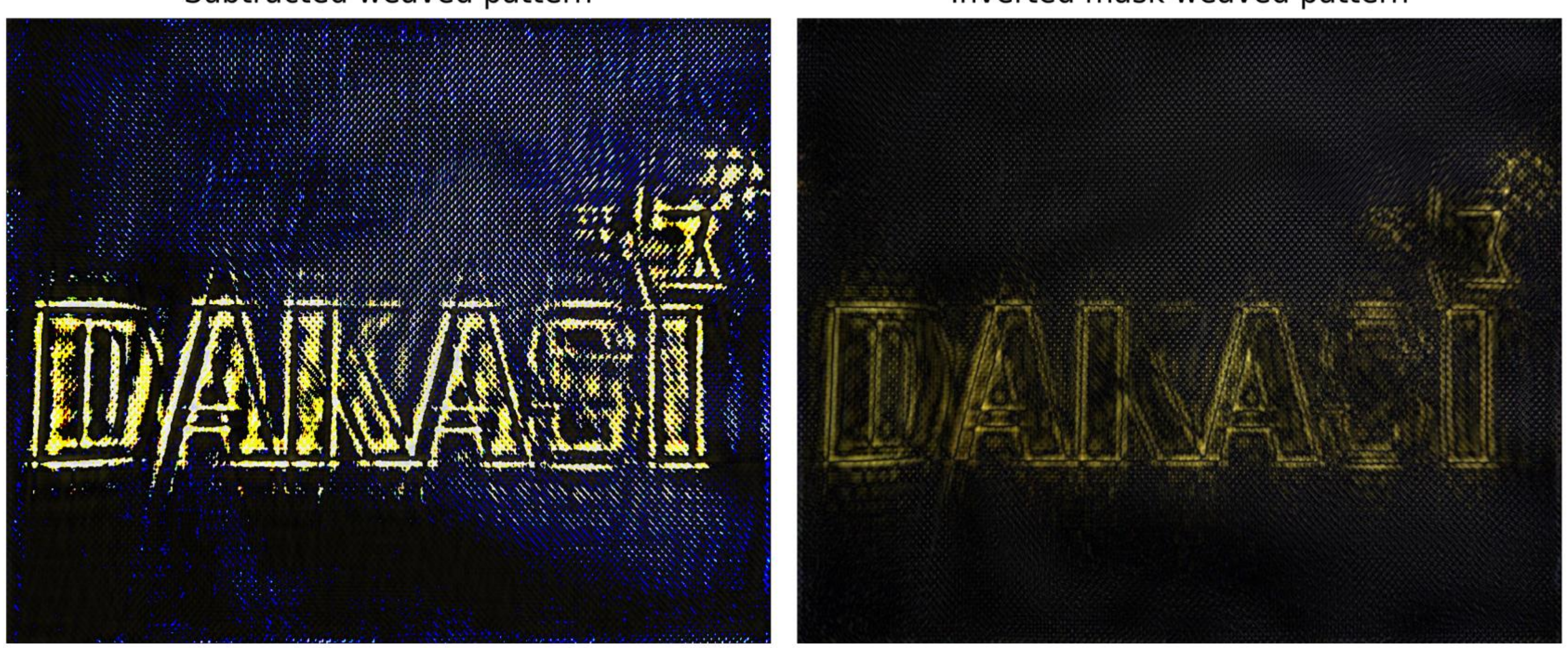



Figure 8. Isolated canvas weaved patterns after masking.

To further validate our results, we **subtract** the resulting image to the original one to **retrieve the canvas pattern**. As expected, ragged surfaces were observed that closely takes the form of the patterns! Similarly, taking the **inverse of the mask** from Figure 6, i.e. all ones becomes zero and vice versa, and feeding it into the coded algorithm also resulted to the retrieval of the sinusoidal patterns. By close examination, the patterns obtained were similar to each other (Yay!)

# CANVAS WEAVE MODELLING


Upon feeling inclined in doing the FFT algorithm for filtering unwanted frequencies in images, we also tried to isolate the **weaving patterns of some common household material** like ecobag. Employing the same **algorithm** used from the previous canvas, the following results were acquired. Room for improvement for the choice of mask must be considered!

Figure 9. Canvas weave modelling of an ecobag with print.

# Convolution Theorem [Revisited]

On the third part of the activity, we look into how the **convolution theorem redux** applies to convolving patterns. The algorithm is similar to what was previously done and recalling that the backbone of image convolution is simply the FFT. In the succeeding slides, we generally take note of the following:

1. The Fourier transform of a convolution of two functions f and g in space is the product of the Fourier transform of two functions F and G, i.e.

$$f * g = FG$$

2. The convolution of a dirac delta at some point and a function results in the replication of that function in the location of the dirac delta, i.e.

$$\int_{-\infty}^{\infty} \delta\left(x - x_0 - x', y - y_0 - y'\right) f(x', y') dx' dy' = f(x - x_0, y - y_0)$$

We will revisit some of the interesting **aperture geometries** coded from the previous activity on Fourier transforms and see how convolution theorem governs the FFTs of such aperture combinations!

# Binary image of two dots

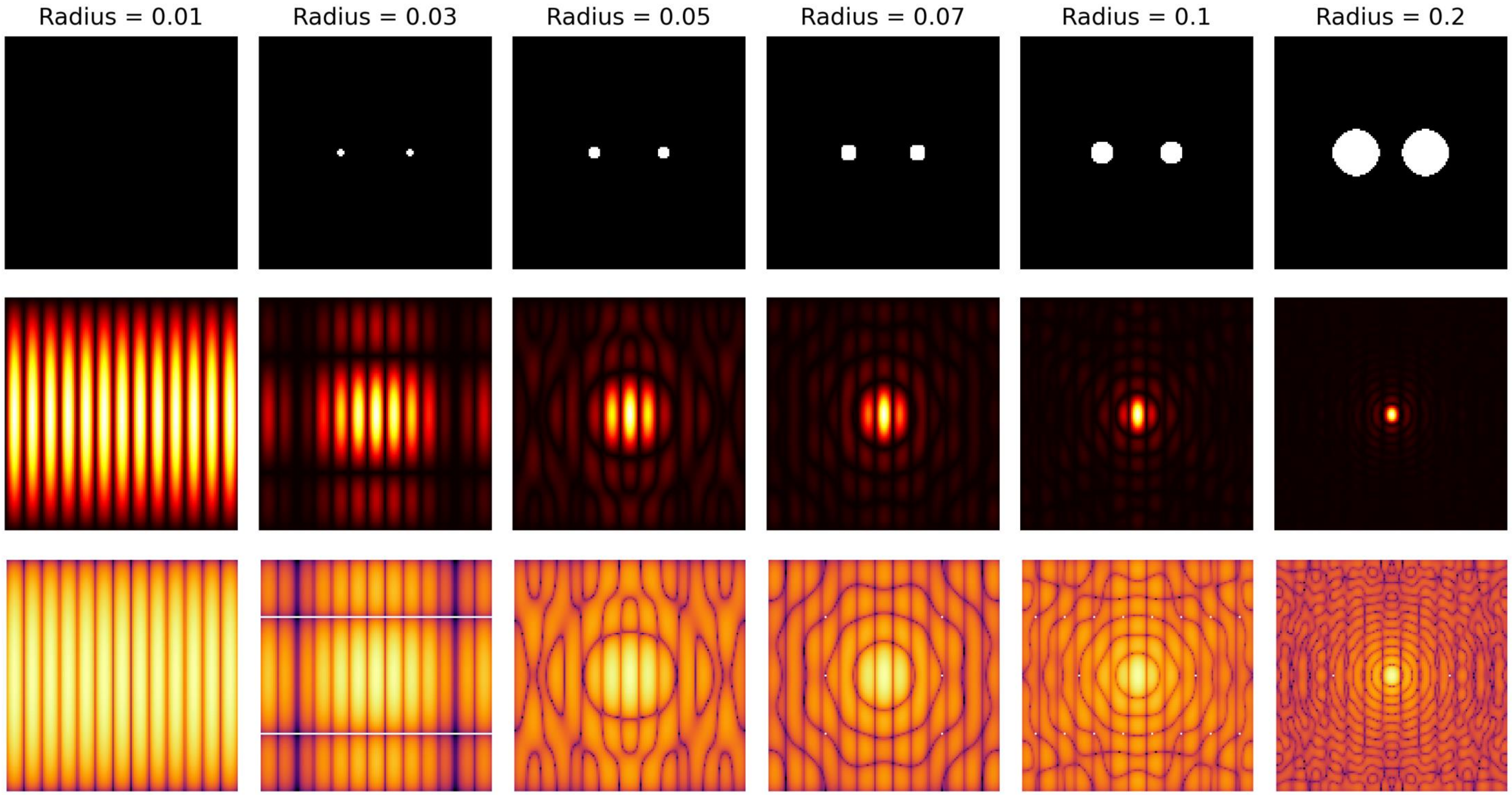



Figure 10. FFTs of two circular dots of varying radii.

From the previous activity, we know that the FFT of a sinusoid is **two points** similar to dirac delta functions. By the convolution theorem, taking the FT of two functions in real space must appear to be the **multiplication of the FTs** of the two functions. Since we know that the FT of a circular aperture is an **airy disk**, and the FT of two slits is like an **enveloped sine**, the FT patterns generally take the form of an **airy pattern and a sinusoid**, as expected!

# Binary image of two squares

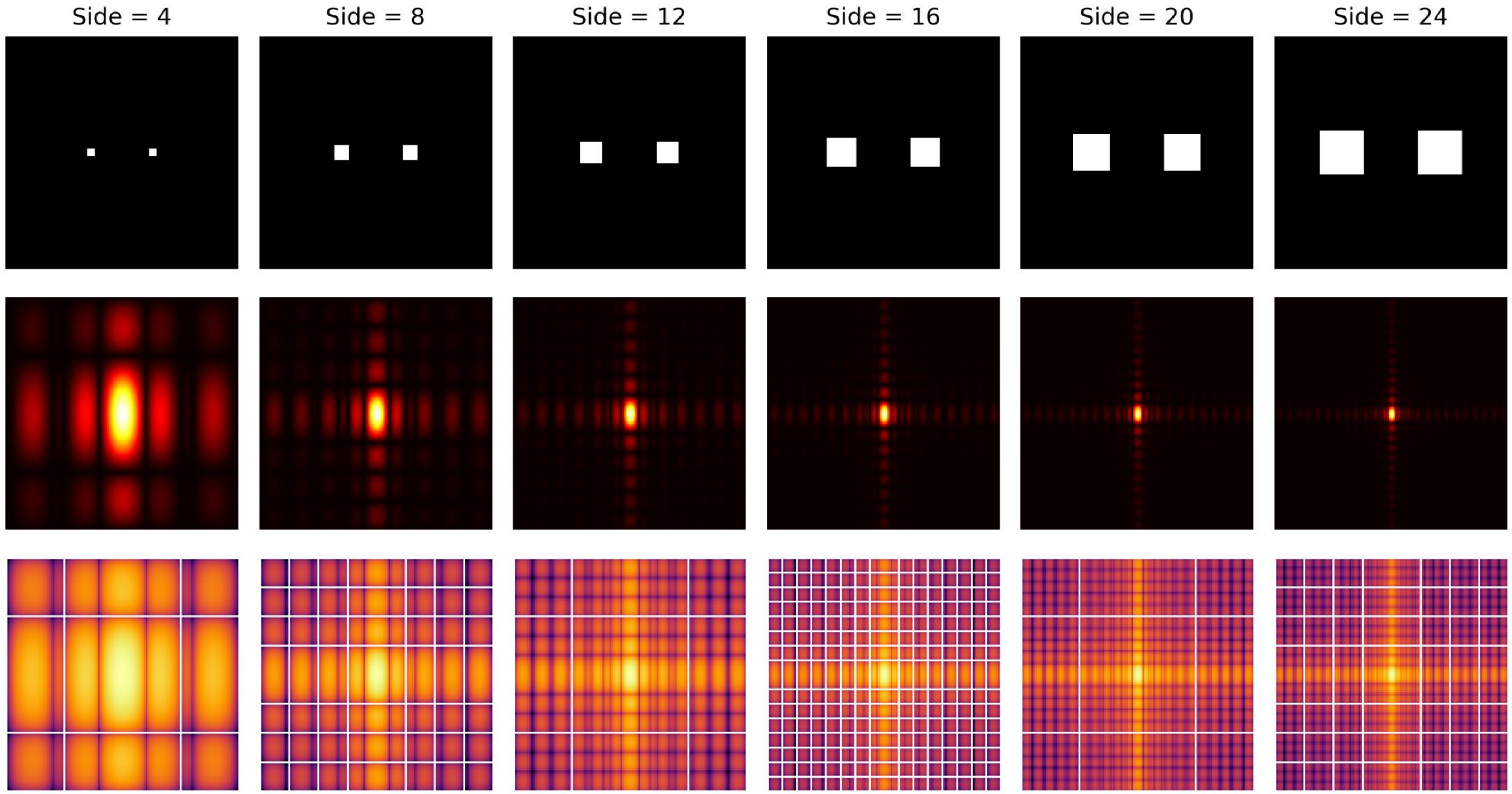



Figure 11. FFTs of two circular squares of varying sizes.

Similarly, we can recall from the previous activities that the FT of a square is an **enveloped sine across the axes** characterized by a **bright band** near the center. Since the FT of two dots is a **sinusoid**, taking the FFTs of two squares resulted to a **combination of an enveloped sine across the axes and the pattern of the sinusoid** in general – which is inherent for double slit apertures!

# Binary image of two Gaussians

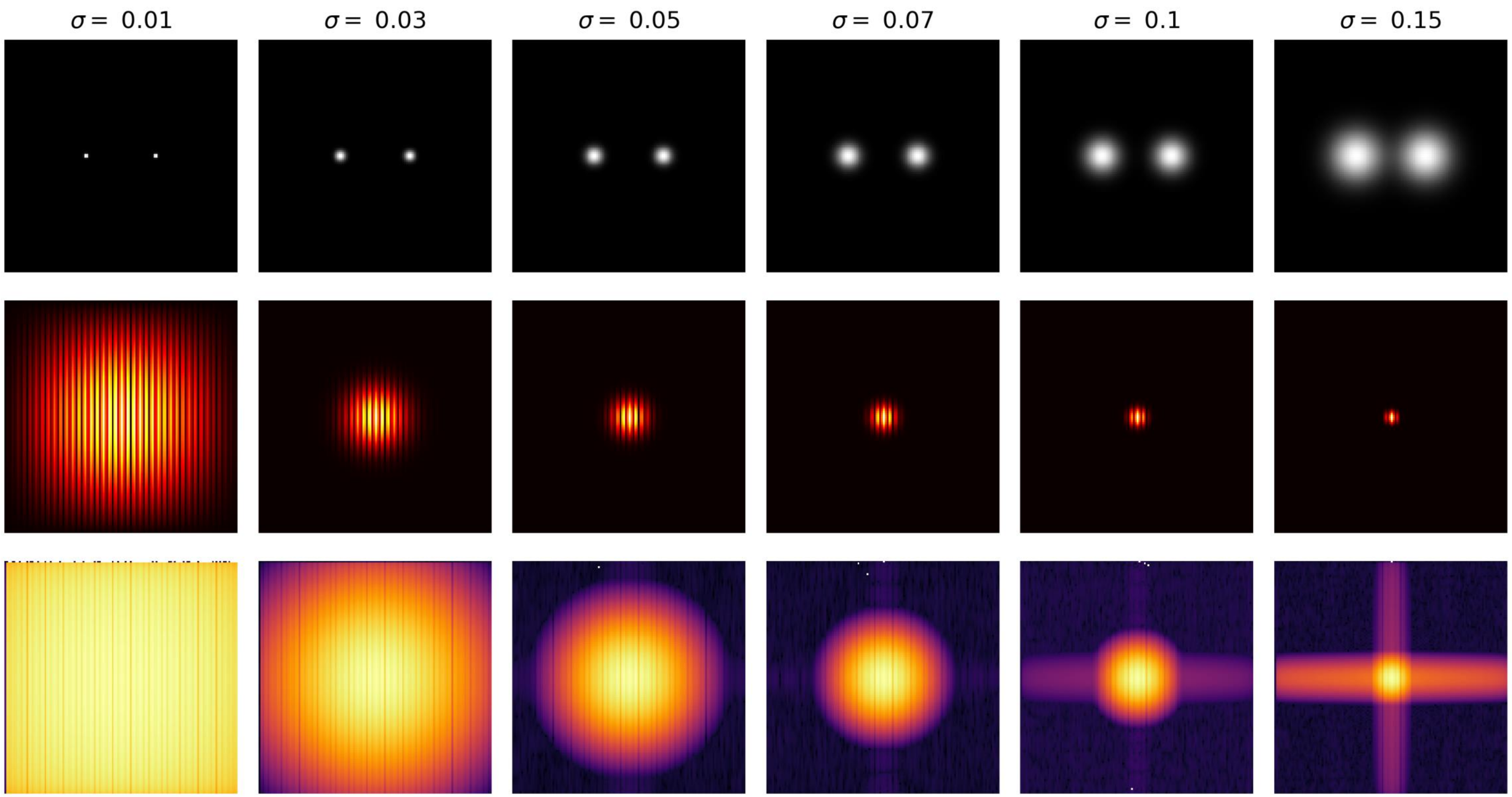



Figure 12. FFTs of two circular Gaussian of varying spread values.

Interestingly, since the FFT of a Gaussian is also a **Gaussian**, the resulting FFTs of a Gaussian of varying spread parameter will always be Gaussian as well. **Sinusoidal patterns** enveloped inside the Gaussian FT were observed for two binary Gaussians since two dots in general will always yield a sinusoid, by the convolution theorem redux!

# Random point patterns

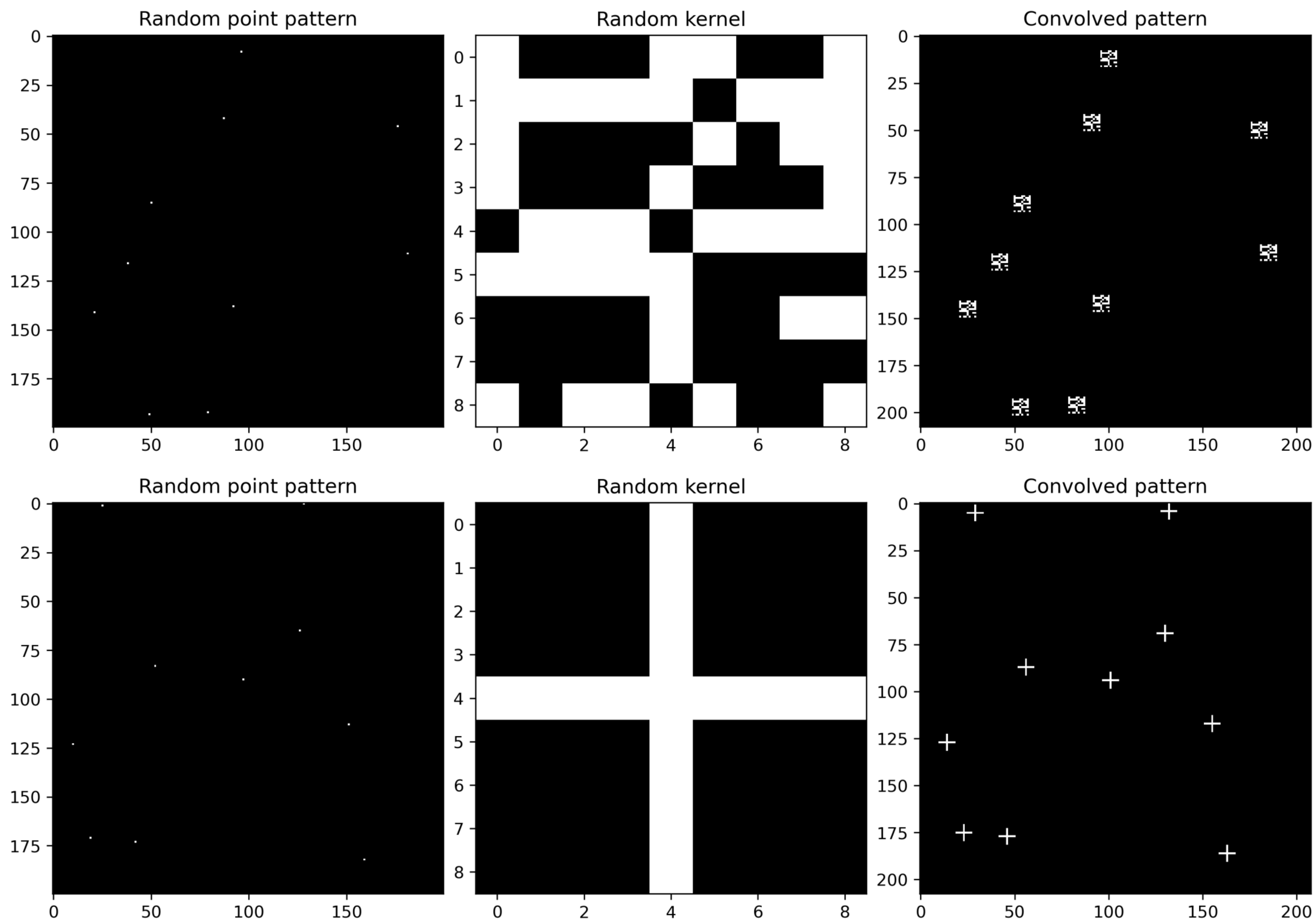



Figure 13. Random point patterns convolved with random kernels.

# Random point patterns

From Figure 13, random point patterns similar to **dirac delts** in space were set up in a 200 by 200 grid. A **random kernel of patterns** were also generated in a 9 by 9 array. Convolving a dot pattern with a such kernel produces an image where the **kernel appears wherever the dots are**, as expected, since convolution in 2-D space is a multiplication in Fourier space.

From the definition highlighted in the previous activities, the **convolution** is a smearing of one function against another, such that the resulting pattern looks a **little like both the random dots and kernel**.

**Some note in this part:** I used `scipy.signal.convolved2d()` function instead of the algorithm that was previously used on the activity since the random point pattern and kernel were not of the same size, which results to an error!

# More point patterns in space

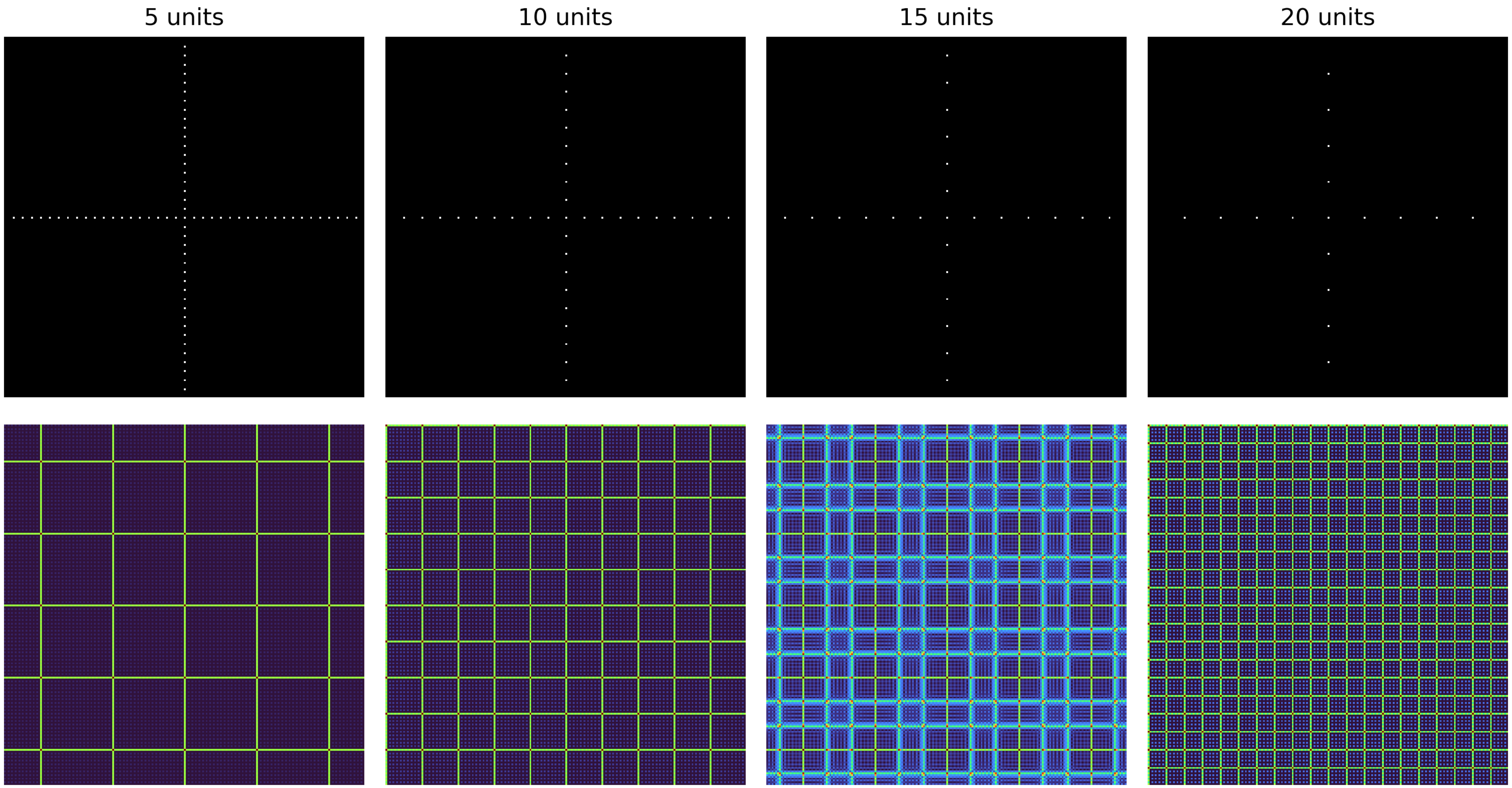



Figure 14. FFTs of equally spaced dots along the axes for varying spacing values.

In general, the dots or peaks in the **Fourier domain** becomes closer to each other as the spacing of the dots increases across the axes in the **space domain**.

# More point patterns in space

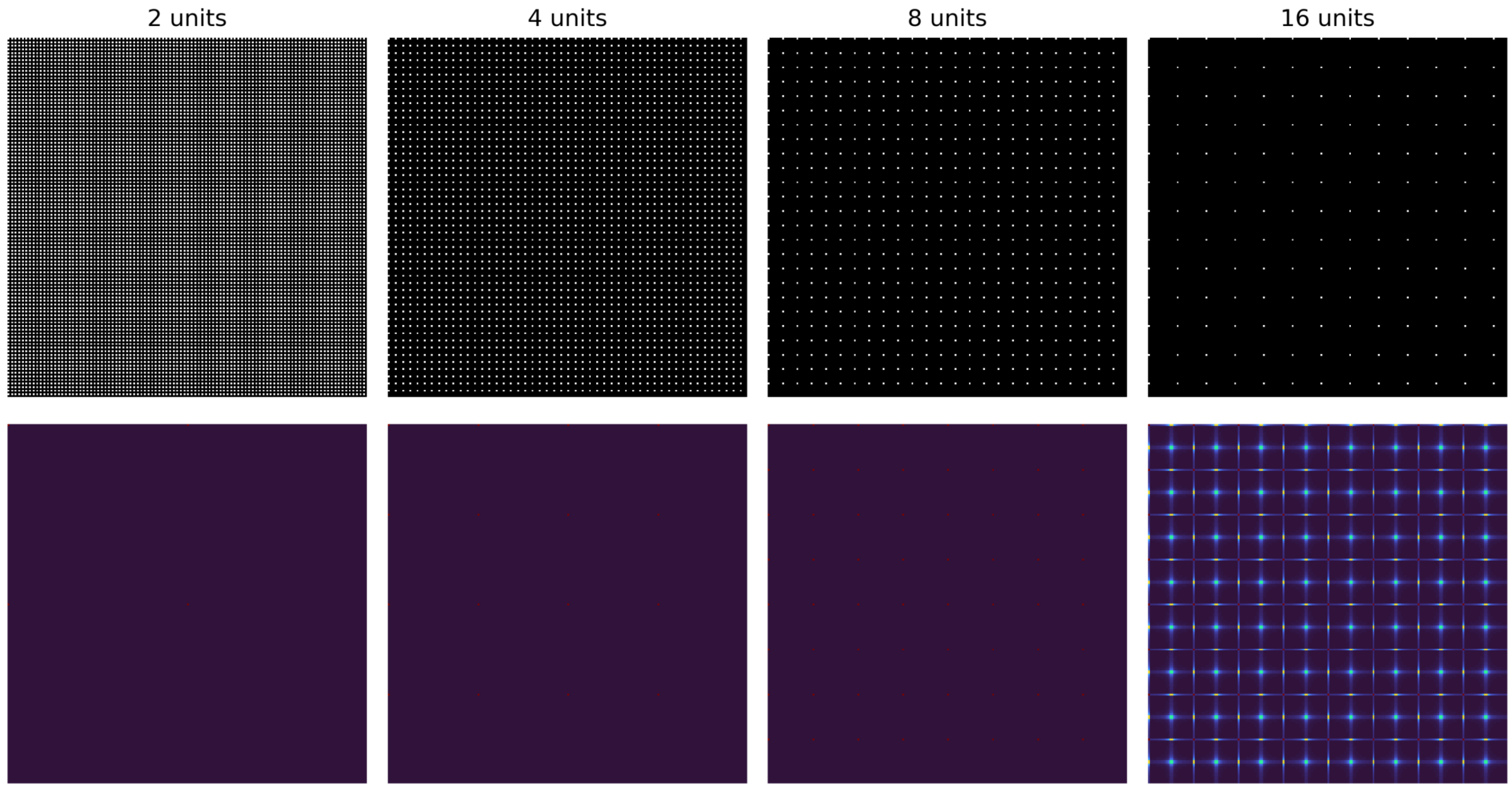



Figure 15. FFTs of equally spaced dots along the grid for varying spacing values.

It can be observed that for very tight spacing, the dot pattern approaches the appearance of a **sinusoid** in the **Fourier domain**. As the spacings become more distant, the resulting pattern of the Fourier transform is a pattern of dots with narrower spacing. This can be interpreted as a superposition of many sines that **destructively interfere** in the right places to form the appearance of a dot in the grids.

# Fingerprint ridge enhancement

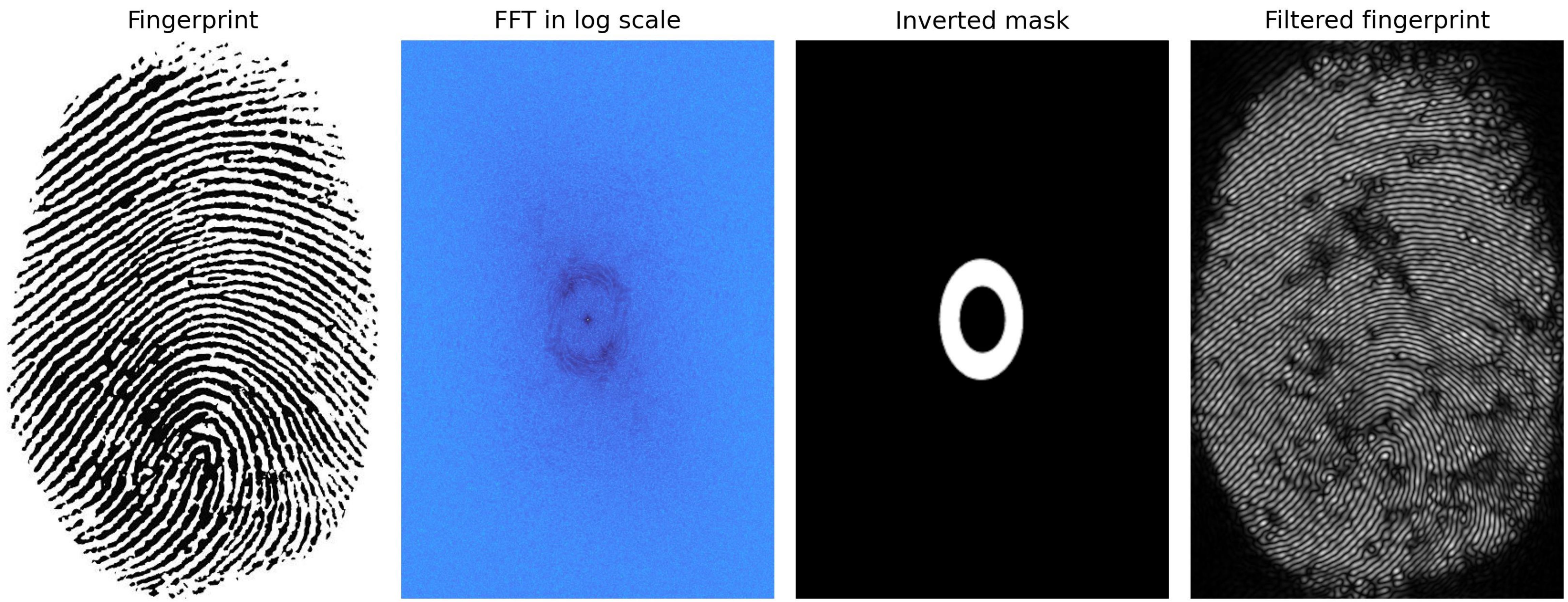



Figure 16. FFTs of equally spaced dots along the grid for varying spacing values.

It can be seen that a bunch of **high-intensity spots** in an ellipse form around the center of the Fourier domain. These are most likely the components we can safely erase since they are of **relatively low frequency**. Note that the point peak at the center or near the origin was not masked as it is highly likely the information that separates the **actual fingerprint from the background** is contained here. The results can further be improved by creating a better mask to ensure the **filtering of unwanted peaks** in the frequency space. Also, the mask was **inverted** since we are only concerned on the pattern itself, which is similar to what was done in the canvas weave modelling activity!

# Lunar landing scanned pictures

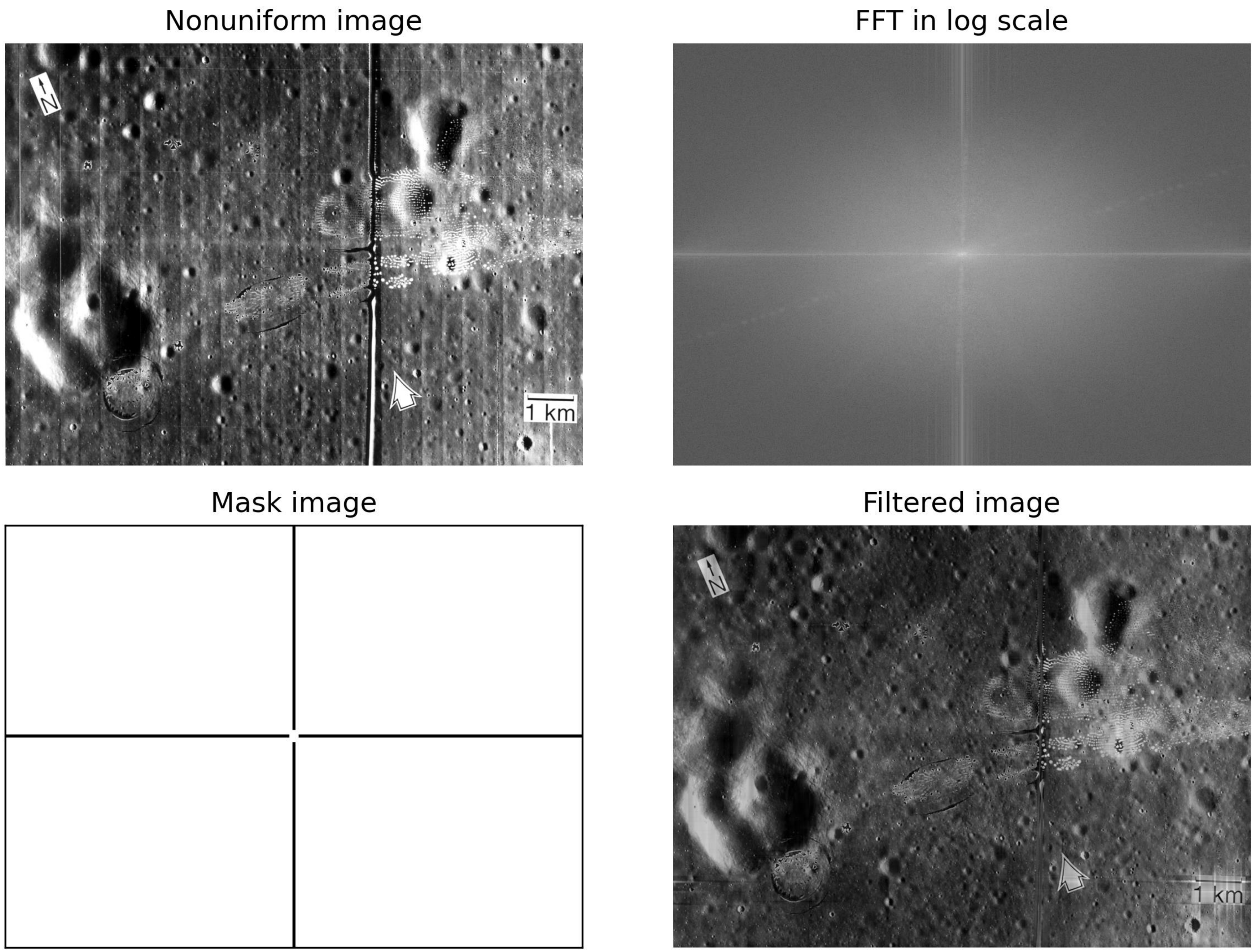



Figure 17. Results of the lunar landing scanned pictures after filtering in the Fourier domain.

# Lunar landing scanned pictures

It can be observed from the original image that the regularly spaced **vertical lines** are the result of combining **individually digitized framelets** to take a composite photograph, and the irregularly-shaped bright and dark spots are due to nonuniform film development. These can be filtered out by using what we have learned in the previous activities on canvas modelling and ridge enhancements!

Upon examining the rotation properties of Fourier transform, lines in the spatial domain are usually rotated by **90 degrees** when represented in the frequency domain. Thus, we can **safely mask out the horizontal line** in the Fourier transform excluding the frequencies at/near zero to get rid of the vertical lines in the original image! The vertical line in the mask were added to **clean further** our image – this eventually erased the bright horizontal line along the center of the image, as It should!

# REFLECTION

I find the activity generally fun since it covers some of the real-world applications of Fourier Transform, which I really look forward to since it transcends beyond the what was taught in the lectures. I was also able to compare the differences of my results to what the theory suggests, specially on the activity on the convolution theorem redux since I was able to revisit some concepts I have picked up on the previous activities in the class. Overall, I would give myself a score of **110/100**. Yay.

# REFERENCES | GITHUB

1. M. Soriano, Applied Physics 157 – Properties and application of the 2D Fourier transform, 2023.
2. White Balancing — An Enhancement Technique in Image Processing