DD2423

IMAGE ANALYSIS AND COMPUTER VISION

LABORATORY REPORT

LAB 1: FILTERING OPERATIONS

Jiang, Sifan
sifanj@kth.se

November 15, 2018

1 Properties of the discrete Fourier transform

1.3 Basis functions

- **Question 1**: Repeat this exercise with the coordinates p and q set to (5, 9), (9, 5), (17, 9), (17, 121), (5, 1) and (125, 1) respectively. What do you observe?
  - The direction of the sinusoid wave depends on p and q.
  - The wavelength of the spatial image depends on p and q. The further the point from the midpoint, the bigger the wavelength of the spatial image is.
  - The amplitudes of all the spatial images are same.

  The output of `fftwave` for each case is illustrated in Figure 1, 2, 3, 4, 5 and 6 accordingly.

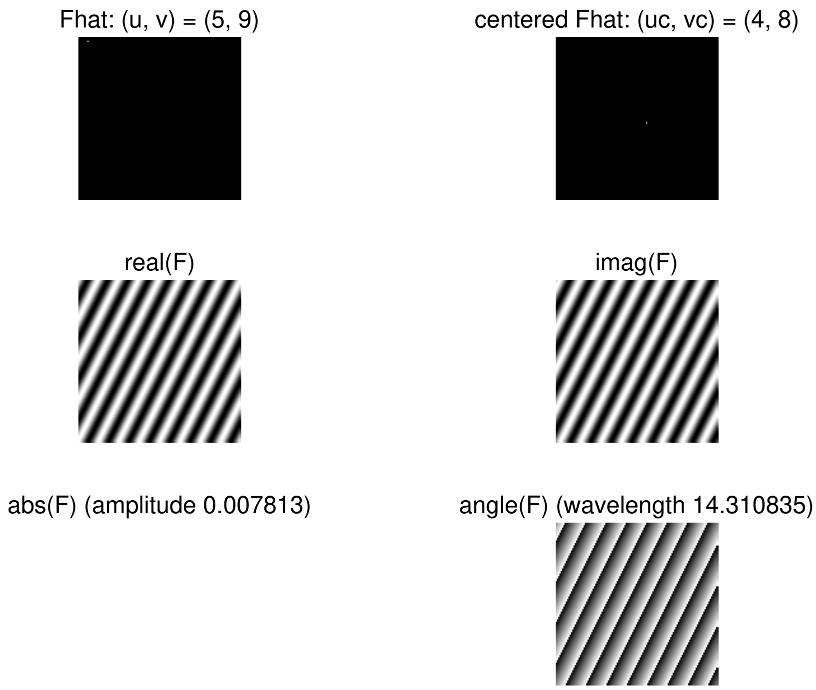


Figure 1: $(p, q) = (5, 9)$.

- **Question 2**: Explain how a position (p, q) in the Fourier domain will be projected as a sine wave in the spatial domain. Illustrate with a MATLAB figure.

  The equation for the spatial domain is:

$$f(x, y) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \delta(u - p, v - q) e^{\frac{2\pi i(xu+yv)}{N}} = \frac{1}{N} e^{\frac{2\pi i(px+qy)}{N}} \tag{1}$$

  The wavelength in x direction and the wavelength in y direction will determine the result of the sine wave in the spatial domain.

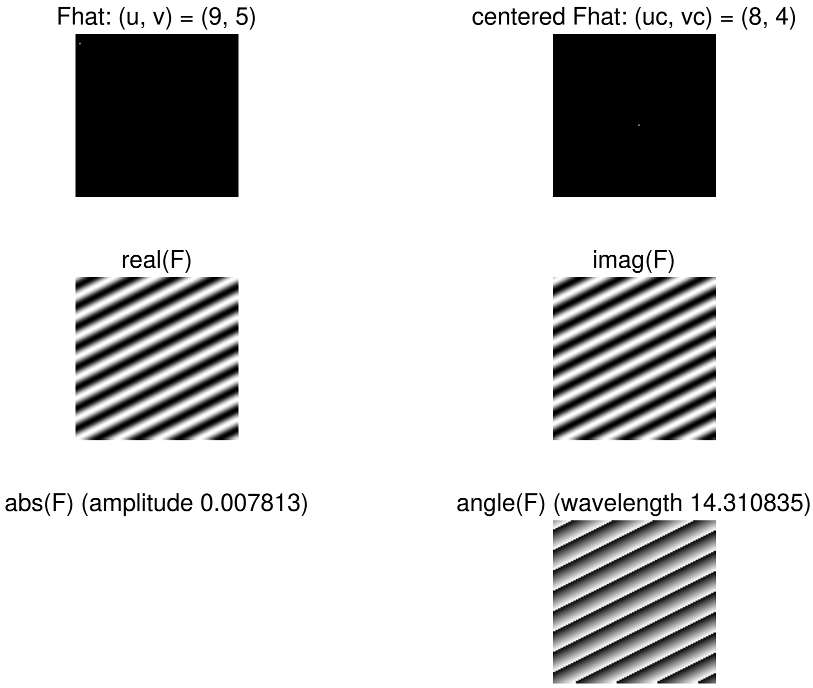


Figure 2: $(p, q) = (9, 5)$.

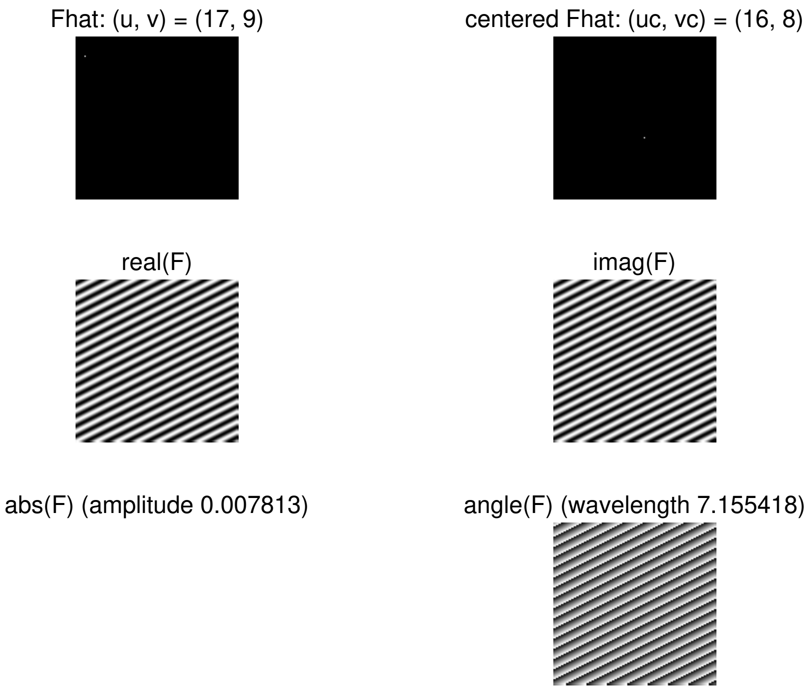


Figure 3: $(p, q) = (17, 9)$.

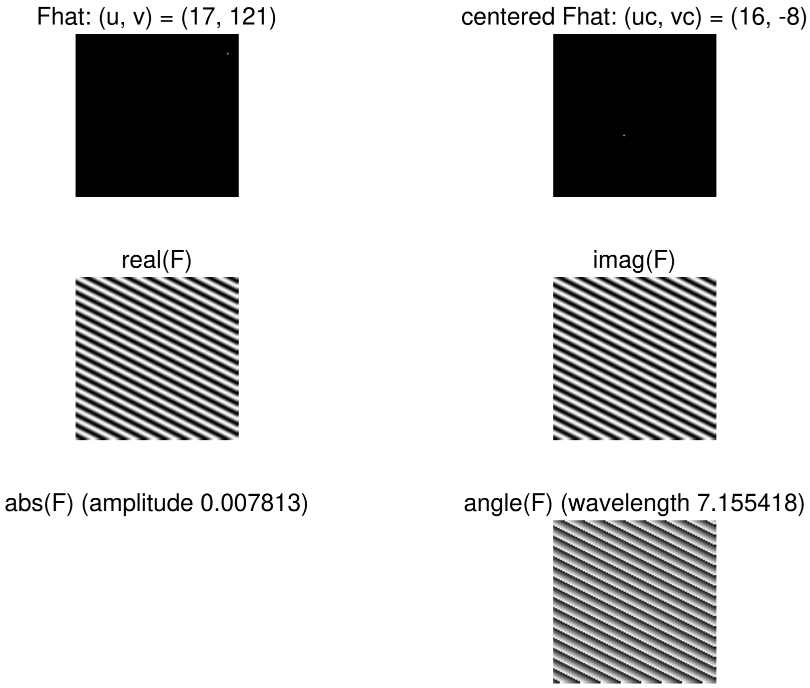


Figure 4: $(p, q) = (17, 121)$.

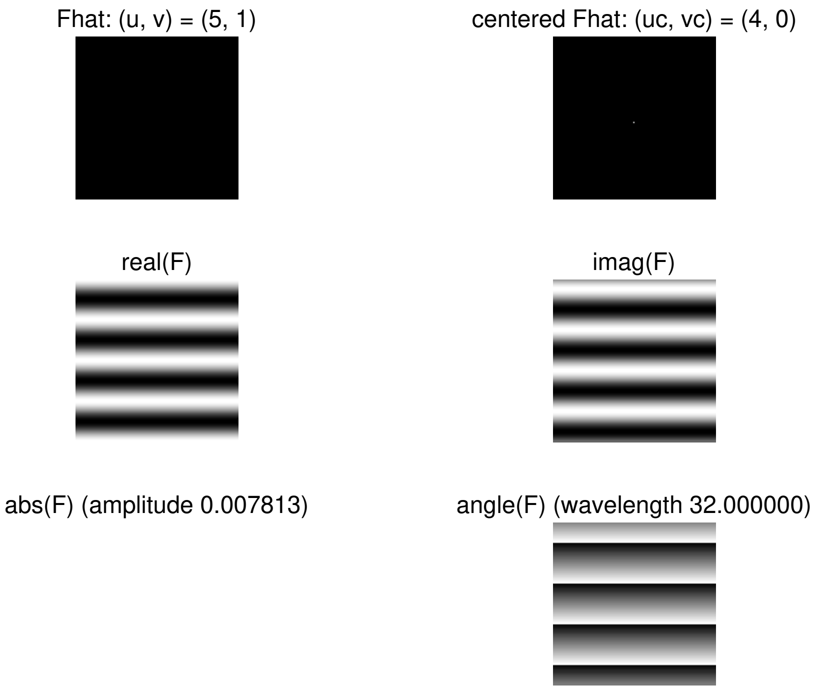


Figure 5: $(p, q) = (5, 1)$.

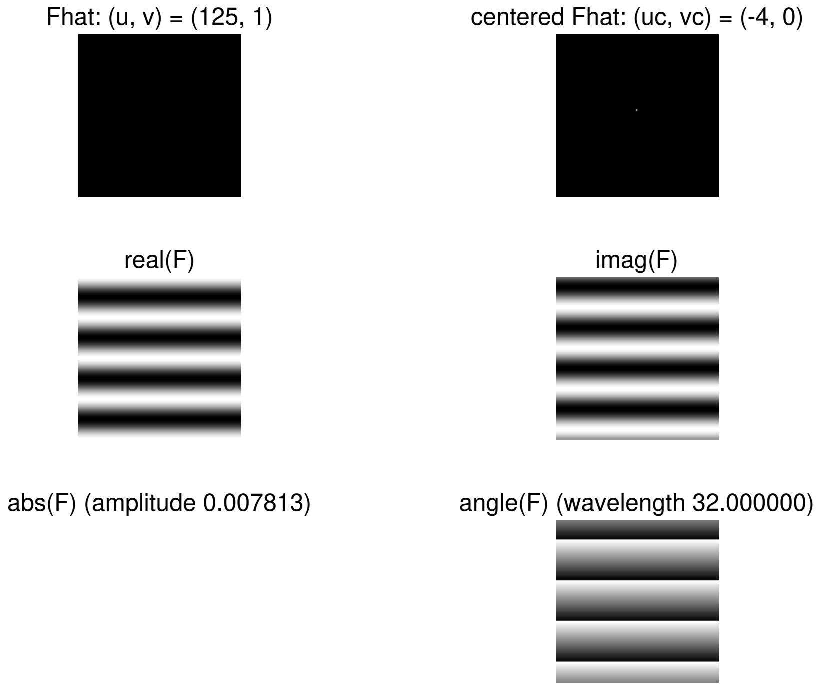


Figure 6: $(p, q) = (125, 1)$.

- **Question 3**: How large is the amplitude? Write down the expression derived from Equation (4) in these notes. Complement the code (variable **amplitude**) accordingly.

  As the spatial equation shown in Equation 1, the amplitude can be derived:

$$|f(x,y)| = \left| \frac{1}{N} e^{\frac{2\pi i(px+qy)}{N}} \right| = \frac{1}{N} \tag{2}$$

- **Question 4**: How does the direction and length of the sine wave depend on p and q? Write down the explicit expression that can be found in the lecture notes. Complement the code (variable **wavelength**) accordingly.

  Base on the equation in the lecture slides:

$$\lambda = \frac{2\pi}{||\omega||} = \frac{2\pi}{\sqrt{\omega_1^2 + \omega_2^2}} \tag{3}$$

$$\omega = \left[\frac{2\pi u}{N} \frac{2\pi v}{N} \right]^T \tag{4}$$

  So the wavelength equation becomes:

$$\lambda = \frac{N}{\sqrt{p^2 + q^2}} \tag{5}$$

  The direction could be obtained from the Equation 4, the angle will be `atan2(`p, q`)` in MATLAB expression.

- **Question 5**: What happens when we pass the point in the center and either p or q exceeds half the image size? Explain and illustrate graphically with MATLAB!

  If either p or q exceeds half the image size which in this case is larger than 64, then the wavelength would be smaller than 2 according to Equation 5. When the period is 2, the spatial domain would be alternative black and white stripes of one-pixel-width. The corresponding maximum angular frequency is $\omega_{max} = \frac{2\pi}{2} = \pi$.

  When either p or q exceed $\frac{N}{2} = 64$, according to Equation 4, the angular frequency of x or y direction would exceed ω_{max}, which means in such an quadratic image, the sinusoid wave could no longer be displayed. Figure 7 shows an example when $(p, q) = (70, 100)$.

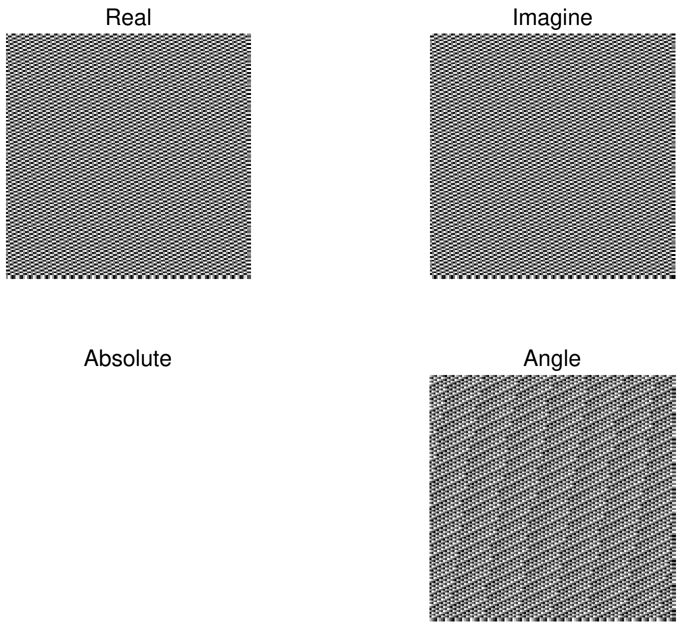


Figure 7: $(p, q) = (70, 100)$.

- **Question 6**: What is the purpose of the instructions following the question `What is done by these instructions?` in the code?

  The purpose of the code following after `What is done by these instructions?` is to shift the Fourier function so that the angular frequency in x and y direction could be inside $[-\pi, \pi]$, which can be also implemented by MATLAB function `fftshift`. Also, this part of code make the origin point move from $(1, 1)$ to $(0, 0)$.

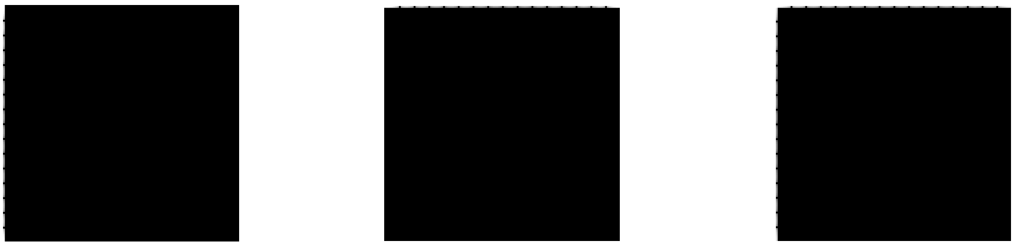
1.4 Linearity

The origin figure of F, G, and H are shown in Figure 8a, 8b, and 8c accordingly. Their Fourier transform and other operations are shown in Figure 9, 10, and 11.

- **Question 7** Why are these Fourier spectra concentrated to the borders of the images? Can you give a mathematical interpretation? Hint: think of the frequencies in the source image and consider the resulting image as a Fourier transform applied to a 2D function. It might be easier to analyze each dimension separately!

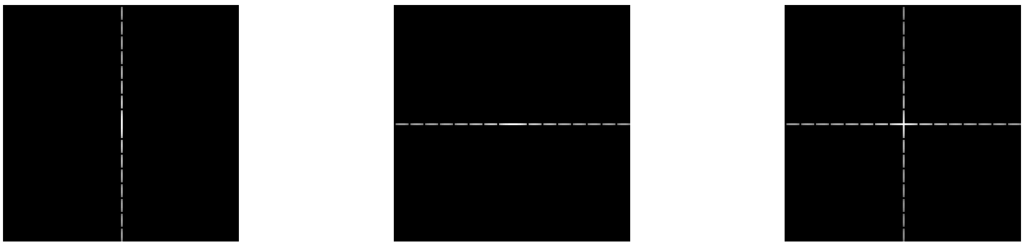
(a) Image F (b) Image G = F′ (c) Image H = F + 2 * G

Figure 8: Origin images.



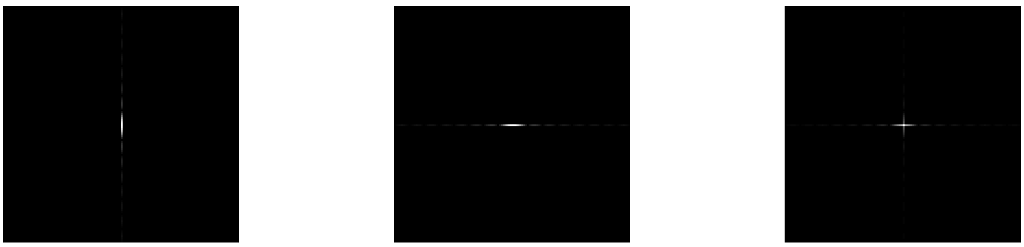
(a) Fourier spectra of F. (b) Fourier spectra of G. (c) Fourier spectra of H.

Figure 9: Fourier spectra originated at upper left corner.



(a) Shifted Fourier spectra of F. (b) Shifted Fourier spectra of G. (c) Shifted Fourier spectra of H.

Figure 10: Fourier spectra originated at middle.



(a) Shifted Fourier spectra of F without `Log`. (b) Shifted Fourier spectra of G without `Log`. (c) Shifted Fourier spectra of H without `Log`.

Figure 11: Fourier spectra originated at middle without `Log`.

The function of F in this case could be written as $F(x, y) = 1$ for all $y \in [y_1, y_2]$ and equals to 0 otherwise. Then we will have:

$$\mathcal{F}(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} F(x,y)e^{-\frac{2\pi i(xu+yv)}{N}} \tag{6}$$

$$= \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=y_1}^{y_2} e^{-\frac{2\pi i(xu+yv)}{N}} \tag{7}$$

$$= \frac{1}{N}\sum_{y=y_1}^{y_2} e^{-\frac{2\pi iyv}{N}} \sum_{x=0}^{N-1} e^{-\frac{2\pi ixu}{N}} \tag{8}$$

$$= \frac{\delta(u)}{N}\sum_{y=y_1}^{y_2} e^{-\frac{2\pi iyv}{N}} \tag{9}$$

Based on Equation 9, we can see that only when $u = 0$ that the Fourier spectra of image F would not be zero, which means the Fourier spectra of F would be concentrated to the left border. Sample reason could be applied to G to explain why the Fourier spectra is concentrated to the up border of the image. As for H, since H is a linear combination of F and G, the Fourier spectra of H would also be a linear combination of the Fourier spectrum of F and G, which gives Figure 9c.

- **Question 8** Why is the logarithm function applied?

  The logarithm function is applied to making the difference between each pixels smaller, so that enhance the dark parts in Figure 11, and making them being seen clearly as shown in Figure 10.

- **Question 9** What conclusions can be drawn regarding linearity? From your observations can you derive a mathematical expression in the general case?

  Comparing Figure 10c with Figure 10a and 10b, we can find that shifted Fourier spectra of H is a linear combination of the one of F and G. In general, the expression of this property could be written as:

$$\mathcal{F}[\alpha g(x,y) + \beta h(x,y)] = \alpha\mathcal{G}(u,v) + \beta\mathcal{H}(u,v) \tag{10}$$

1.5 Multiplication

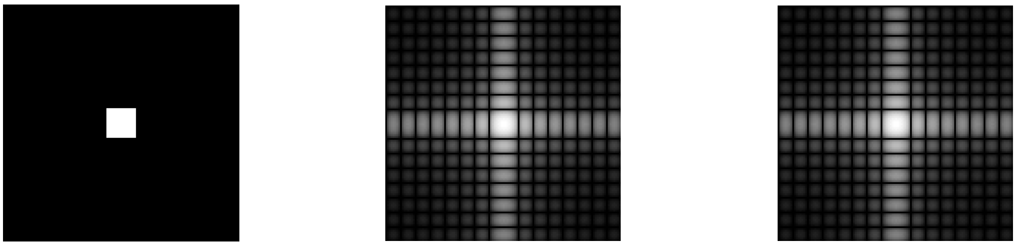
In this section, image F is Figure 8a and image G is Figure 8b.

- **Question 10** Are there any other ways to compute the last image? **Remember what multiplication in Fourier domain equals to in the spatial domain!** Perform these alternative computations in practice.

  Based on the convolution theory of Fourier transform, we have such equation:

$$\mathcal{F}[H(x,y) \times G(x,y)] = \mathcal{H}(u,v) * \mathcal{G}(u,v) \tag{11}$$

  So, we can convolute $\mathcal{F}$ and $\mathcal{H}$ in the Fourier domain. Since `fft2` in MATLAB uses 1 as factor in the FFT-routine, we divide the convolution result by N^2. Also, the convolution would give a 255×255 matrix for the result, so we would only use the 128×128 matrix at the left upper corner. And finally we could get the shifted Fourier spectra as Figure 12c which is same to Figure 12b, the result given by multiplication and then Fourier transformation.



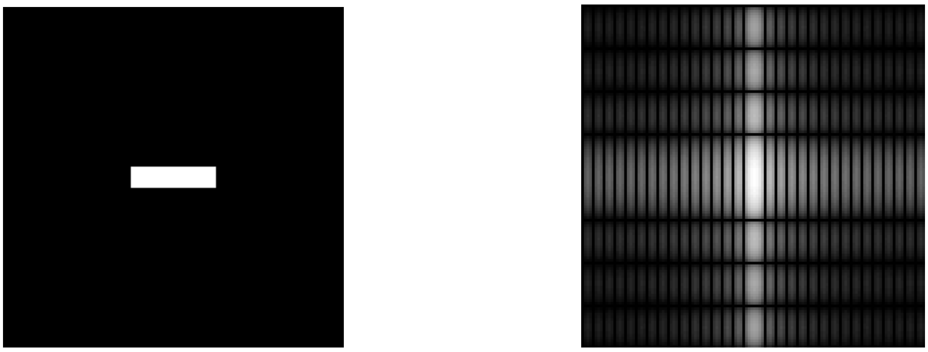
(a) Multiplication of F and G. (b) Shifted Fourier spectra of multiplication of F and G. (c) Shifted Fourier spectra of convolution of $\mathcal{F}$ and $\mathcal{G}$.

Figure 12: Convolution.

1.6 Scaling

- **Question 11** What conclusions can be drawn from comparing the results with those in the previous exercise? See how the source images have changed and analyze the effects of scaling.

  Image F and its Fourier transform are illustrated in Figure 13a and 13b respectively.



(a) Scaled F. (b) Shifted Fourier spectra of scaled F.

Figure 13: Scaling.

Comparing to Figure 12a and 12b, we can see that F in Figure 13a is a scaled version of Figure 12a which is half of its height and double of this width. While Figure 13b is stretched in height and compressed in width when comparing to Figure 12b. The scaling property of discrete Fourier transform can be derived as Equation 12.

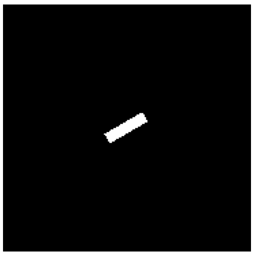
$$\mathcal{F}[G(\alpha x, \beta y)] = \frac{1}{|\alpha\beta|}\mathcal{G}(\frac{u}{\alpha}, \frac{v}{\beta}) \tag{12}$$

Since $F_2(x, y) = F_1(\frac{x}{2}, 2y)$, $\mathcal{F}_2(u, v) = \mathcal{F}_1(2u, \frac{v}{2})$.

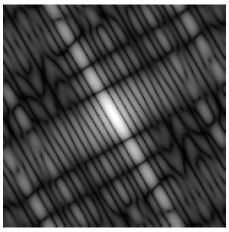
1.7 Rotation

- **Question 12** What can be said about possible similarities and differences? Hint: think of the frequencies and how they are affected by the rotation.

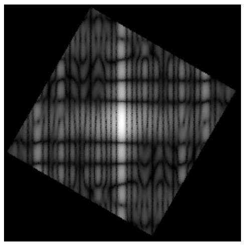
The original image of F and its shifted Fourier with logarithm function are illustrated in Figure 13a and 13b. The rotated image of F, their Fourier spectrum, and their rotated back Fourier spectrum with rotation angle varies from 30°, 60°, and 90° are illustrated in Figure 14.



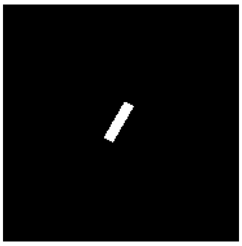
(a) Rotated by 30°.



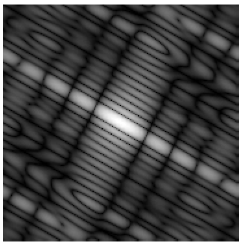
(b) Fourier transform with 30° rotation.



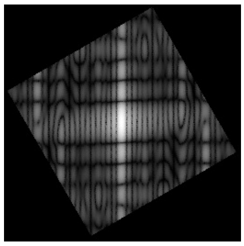
(c) Image rotated back by 30°.



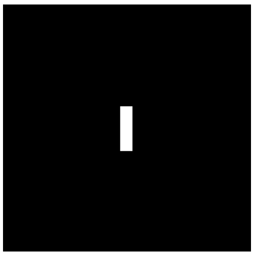
(d) Rotated by 60°.



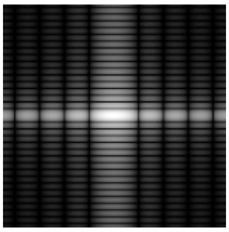
(e) Fourier transform with 60° rotation.



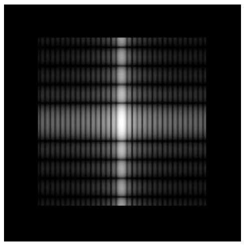
(f) Image rotated back by 60°.



(g) Rotated by 90°.



(h) Fourier transform with 90° rotation.



(i) Image rotated back by 90°.

Figure 14: Rotation.

Based on Figure 14, we can conclude that a rotation angle of θ in spatial domain would give a rotation angle of θ in Fourier domain. However, the rotation of original image result in the loss of smoothness on the edges of the white rectangle especially in the case of 30° and 60°. This gives the Fourier spectrum a wave-like grain.

1.8 Information in Fourier phase and magnitude

- **Question 13** What information is contained in the phase and in the magnitude of the Fourier transform?

  From Figure 15, we can see that the phase information is important for recognizing the feature of the image, since the second column of images have the same phase with the original images while different in magnitude. In contrast, the last column of images have the same magnitude with the original images while different in phase. However, its impossible to feature the images in the last column.

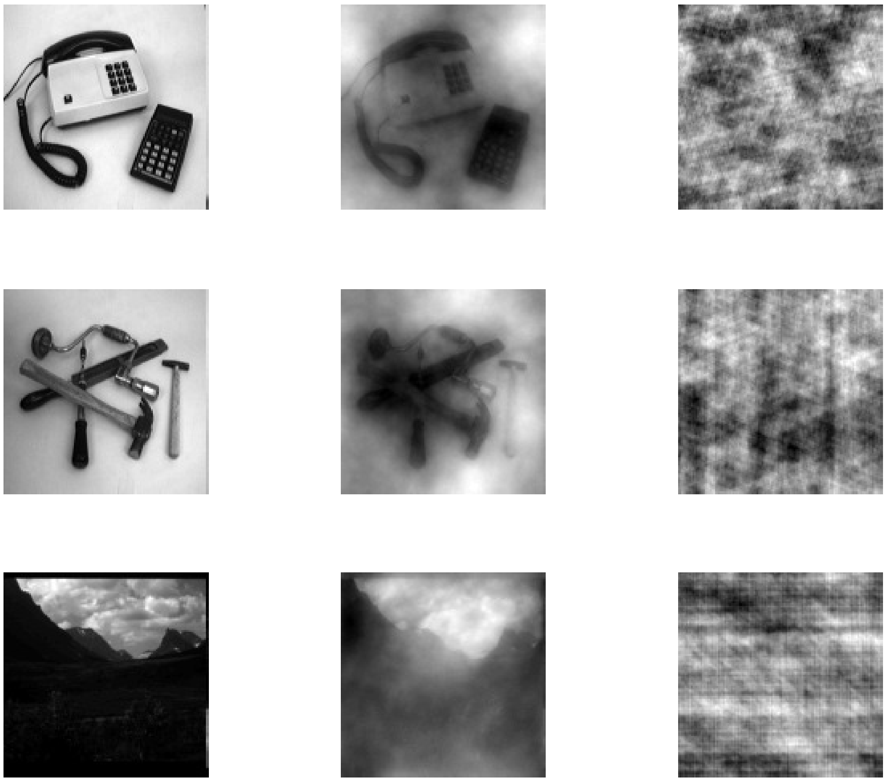


Figure 15: The first column are the original images. The second column contains the power spectrum for the images in the form of $\left|\hat{f}(\omega)\right|^2 = \frac{1}{a|\omega|^2}$. The last column contains the random phase images.

Phase contains the phase difference between the sinusoid waves, thus could be used to determine the profile of the objects in the image. With phase being replaced by random values, the image could not be recognized any longer.

2 Gaussian convolution implemented via FFT

2.3 Filtering procedure

- **Question 14** Show the impulse response and variance for the above mentioned t-values. What are the variances of your discretized Gaussian kernel for t = 0.1, 0.3, 1.0, 10.0 and 100.0?

  Figure 16 shows the impulse responses of the Gaussian kernel with different t.

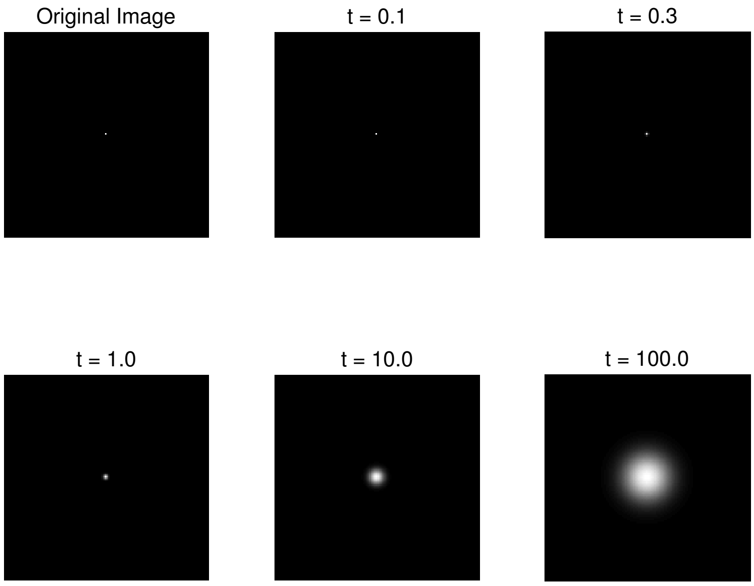


Figure 16: Impulse response.

The variances for each t are:

$$var_{t=0.1} = \begin{bmatrix} 0.0133 & 0.0000 \\ 0.0000 & 0.0133 \end{bmatrix}$$
$$var_{t=0.3} = \begin{bmatrix} 0.2811 & 0.0000 \\ 0.0000 & 0.2811 \end{bmatrix}$$
$$var_{t=1.0} = \begin{bmatrix} 1.0000 & 0.0000 \\ 0.0000 & 1.0000 \end{bmatrix}$$
$$var_{t=10.0} = \begin{bmatrix} 10.0000 & 0.0000 \\ 0.0000 & 10.0000 \end{bmatrix}$$
$$var_{t=100.0} = \begin{bmatrix} 100.0000 & 0.0000 \\ 0.0000 & 10.0000 \end{bmatrix}$$

- **Question 15** Are the results different from or similar to the estimated variance? How does the result correspond to the ideal continuous case? Lead: think of the relation between spatial and Fourier domains for different values of t.

  Referring to both Figure 16 and the variances, when $t = 0.1$ and $t = 0.3$ the results are different from the ideal continuous case. While, when $t \geq 1$, the result follows the ideal continuous case. The reason is when t is too small, the impulse response would only be a pixel, thus the variance of would be different to the ideal case.

- **Question 16** Convolve a couple of images with Gaussian functions of different variances (like t = 1.0, 4.0, 16.0, 64.0 and 256.0) and present your results. What effects can you observe?

  Figure 17 and 18 show the result after applying Gaussian filter with various t to `phonecalc128` and `few128`. From the figures, we could conclude that the larger the t is, the more blurry the image becomes.

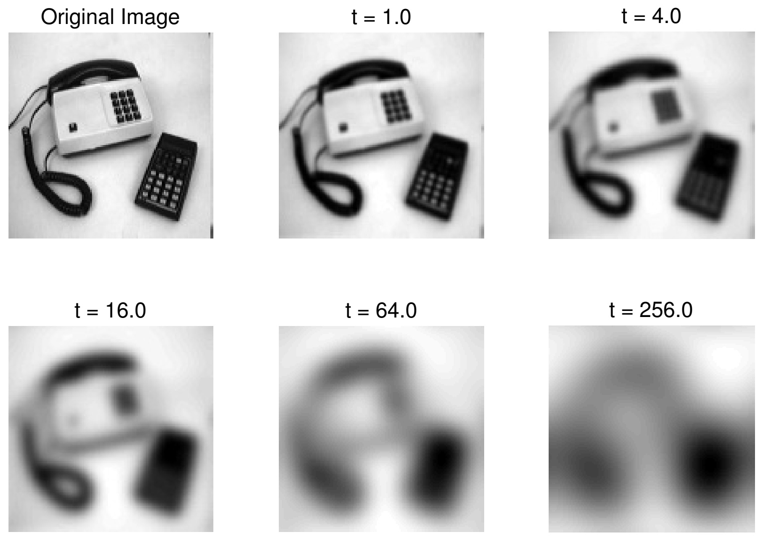


Figure 17: `phonecalc128` with Gaussian filter.

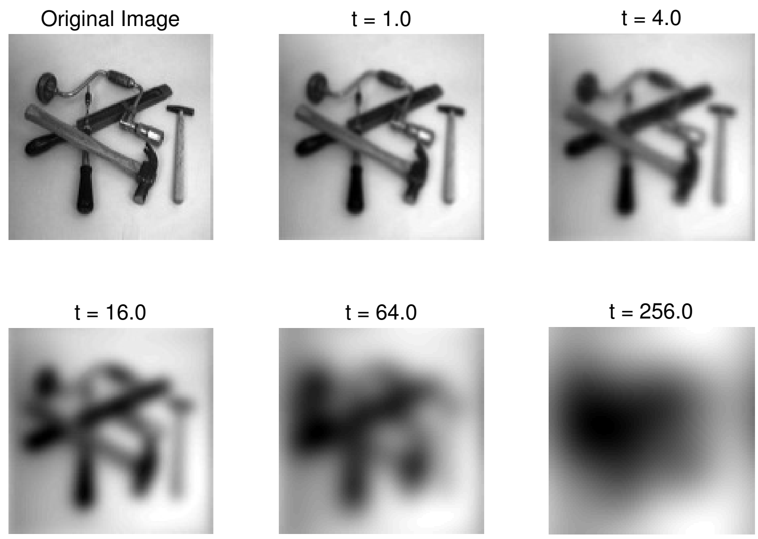


Figure 18: `few128` with Gaussian filter.

3 Smoothing

3.1 Smoothing of noisy data

The original image of `office256` and its noisy images are shown in Figure 19.

- **Question 17** What are the positive and negative effects for each type of filter? Describe what you observe and name the effects that you recognize. How do the results depend on the filter parameters? Illustrate with MATLAB figure(s).

  The result of Gaussian filter, median filter and ideal low-pass filter are shown in Figure

(a) Original image. (b) Image with Gaussian noise. (c) Image with Salt-and-pepper noise.

Figure 19: Original image and image with noise.

20, 21, and 22 correspondingly.

(a) Gaussian filtering to Gaussian noise with variance 0.1. (b) Gaussian filtering to Gaussian noise with variance 1. (c) Gaussian filtering to Gaussian noise with variance 4.

(d) Gaussian filtering to Salt-and-pepper noise with variance 0.1. (e) Gaussian filtering to Salt-and-pepper noise with variance 1. (f) Gaussian filtering to Salt-and-pepper noise with variance 4.

Figure 20: Gaussian filtering.

- Gaussian smoothing blurs the image to reduce noise obviously and is very effective in removing Gaussian noise. However, with the variance increases, the image loses detail. The performance in smoothing the image with Salt-and-pepper noise is not so good as Median filter, the "salt" and "pepper" pixels are blurred but still obvious.
- Median filtering is very effective in removing Salt-and-pepper noise as we can see from Figure 21d, the 2×2 window size of median filter removes all the Salt-and-pepper pixels of the image. Also, the edges of the images are still relatively sharp in the median filtering cases. However, median filtering loses details very quickly when the window size becomes bigger.
- Ideal low-pass filtering performs worst behavior in both types of noise. If the cut-off frequency is low, then all the moderate and high frequency would be removed, thus becoming more likely to have sinusiod like output image based on the theory

from Section 1. While, if the cut-off frequency is high, then small change would be applied to the noise.

(a) Median filtering to Gaussian noise with window size 2×2.

(b) Median filtering to Gaussian noise with window size 3×3.

(c) Median filtering to Gaussian noise with window size 4×4.

(d) Median filtering to Salt-and-pepper noise with window size 2×2.

(e) Median filtering to Salt-and-pepper noise with window size 3×3.

(f) Median filtering to Salt-and-pepper noise with window size 4×4.

Figure 21: Median filtering.

(a) Ideal low-pass filtering to Gaussian noise with cut-off frequency 0.1.

(b) Ideal low-pass filtering to Gaussian noise with cut-off frequency 0.25.

(c) Ideal low-pass filtering to Gaussian noise with cut-off frequency 0.5.

(d) Ideal low-pass filtering to Salt-and-pepper noise with cut-off frequency 0.1.

(e) Ideal low-pass filtering to Salt-and-pepper noise with cut-off frequency 0.25.

(f) Ideal low-pass filtering to Salt-and-pepper noise with cut-off frequency 0.5.

Figure 22: Ideal low-pass filtering.

- **Question 18** What conclusions can you draw from comparing the results of the respective methods?

Gaussian smoothing is effective in removing Gaussian noise. Median filtering is effective in removing Salt-and-pepper noise. Both of Gaussian filter and ideal low-pass filter decrease the highest frequency of the image thus blurring the image in result. Higher variance of Gaussian filter in spatial domain, the lower the variance in the Fourier domain, which resulting in more high frequencies being discarded. Ideal low-pass filter remove the frequency that is higher than the cut-off frequency in the Fourier domain thus resulting in the "ringing" effect.

3.2 Smoothing and sub-sampling

- **Question 19** What effects do you observe when sub-sampling the original image and the smoothed variants? Illustrate both filters with the best results found for iteration $i = 4$.

  Figure 23 illustrate the sub-sampled images and the smoothed variants with Gaussian smoothing of variance 0.5 and ideal low-pass filtering of frequency 0.25. The Gaussian filter blurs the image while the ideal low-pass filter brings sinusoid texture to the image. Sub-sampling would make the pixels bigger and use one pixel to replace four pixels before sub-sampling. Since the loss of values, the image becomes rough. Higher resolution image contains more details after smoothing and filtering. Moreover, smoothing before sub-sampling introduces smoother edges to the image.

- **Question 20** What conclusions can you draw regarding the effects of smoothing when combined with sub-sampling? Hint: think in terms of frequencies and side effects.

  Smoothing and filtering with Gaussian or ideal low-pass filter decreases highest frequency of the image, thus preventing aliasing before sampling. Also, in this way, the side effects are decreased so that aliasing is less likely to take place.

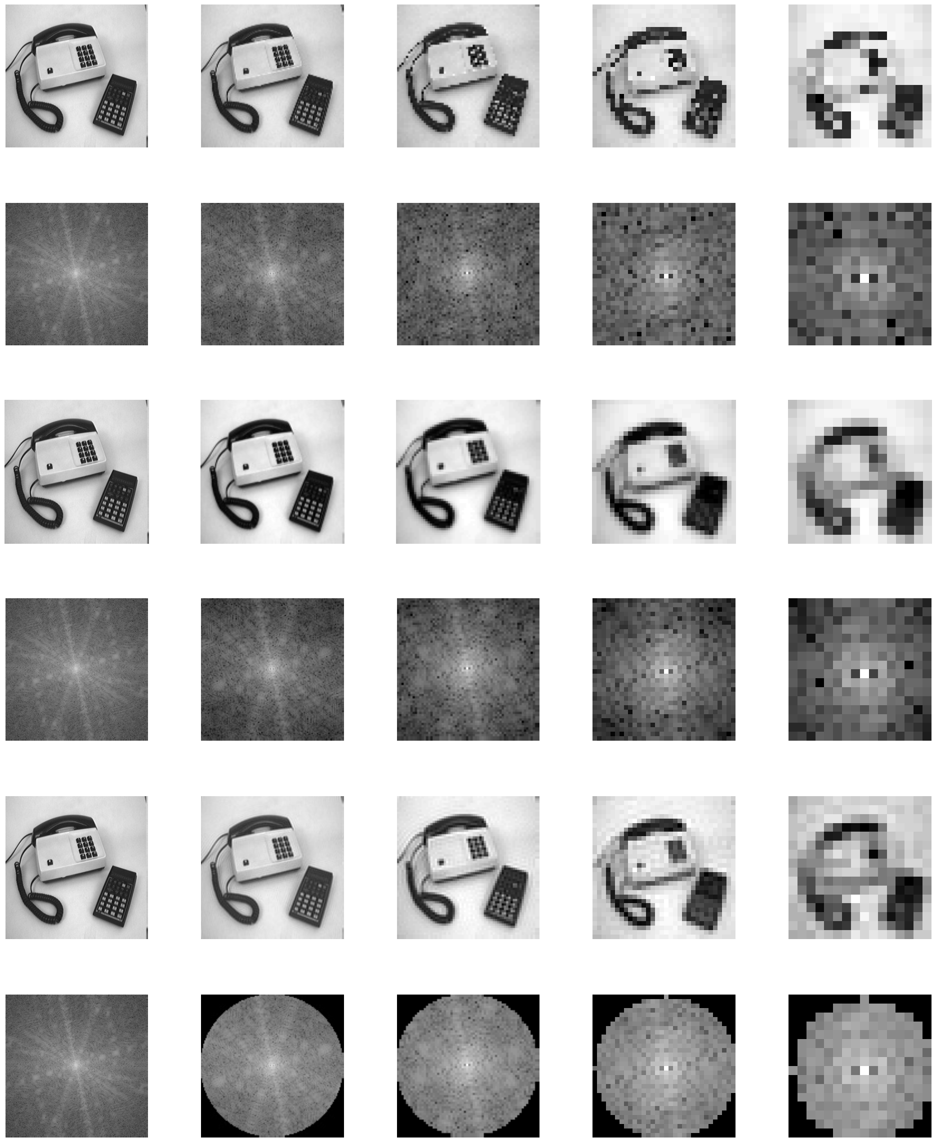


Figure 23: The first two rows contain the images with just sub-sampling and their Fourier spectrum. The second two rows contain the images with Gaussian smoothing and sub-sampling and their Fourier spectrum with variance 0.5. The last two rows contain the images with ideal low-pass filtering and sub-sampling and their Fourier spectrum with cut-off frequency 0.25. The first column of images are all original and their spectrum.