Assignment: Fourier Transform
Signal and Image Processing

February 16, 2026

You can work on this assignment and submit your solution (report and code) as a GROUP. This assignment counts towards your grade and have to be submitted in order to pass the course. You must follow the report guidelines found in `guidelines.pdf`. The page limit for this assignment is **8 pages** including everything, i.e. illustrations and code snippets.

1. **Fourier Transform – Theory**

   1.1. (1 point) Prove that the continuous Fourier transform of a real and even function is real and even.

   1.2. Consider the box function

$$b_a(x) = \begin{cases} 1/a & \text{if } |x| \leq \frac{a}{2} \\ 0 & \text{otherwise} \end{cases}. \tag{1}$$

   i. (1 point) Show that $\int_{-\infty}^{\infty} b_a(x)\, dx = 1$

   ii. (1 point) Show that the continuous Fourier transform of b_a, using the definition of the Fourier transform given in Bracewell Chapter 2 (system 1), is $B_a(k) = \frac{1}{ak\pi} \sin(ak\pi)$. Rewrite $B_a(k)$ using the $\text{sinc}(x) = \frac{\sin x}{x}$.

   iii. (1 point) Show that $\lim_{a\to 0} B_a(k) = 1$ (Hint: $\lim_{x\to 0} \frac{\sin x}{x} = 1$).

   iv. (1 point) When a is near zero, then b_a is narrow in the x space domain. In this case, would you consider its Fourier transform narrow or wide in the frequency domain, k? What is the relation, when a is large? Explain your answer.

   Deliverables: Include crucial steps in derivations, and a short comment to each answer.

2. **Fourier Transform – In Practice:**

   2.1. (1 point) Use Python to calculate the power spectrum of `trui.png` (we suggest you use `scipy.fft.fft2` for computing the discrete Fourier transform). Apply the function `scipy.fft.fftshift` and interpret the resulting representation of the image.
   Deliverables: Your answer should include examples of the input and output, crucial Python code snippets, and definitely a description of which problems were solved, how, and an evaluation of the results.

2.2. (1 point) Write a program that adds the function $a_0 cos(v_0 x + w_0 y)$ to `cameraman.tif`. Compute and describe the power spectrum of the result. Design a filter, which removes any such planar waves given v_0 and w_0. You do not need to automate the process of finding v_0, w_0, i.e. you are allowed to hand-craft a filter for a particular combination of v_0, w_0.
Deliverables: Your answer should include examples of the input and output, crucial Python code snippets, and definitely a description of which problems were solved, how, and an evaluation of the results.

2.3. (1 point) Write a function that computes the average of the power spectrum at each spatial frequency, where each spatial frequency corresponds to the euclidean distance from the center of the spectrum. *Hint*: You can ignore distances that are larger than the maximum distance from the center to the edge of the spectrum. Apply the function to visualize how the average power is varying over the spatial frequencies. Do this for the `'bigben_cropped_gray.png'` and a random noise image with the same spatial dimensions.
Deliverables: Include the code for the function. Plot the average power spectrum over all spatial frequencies both images (in log/log scale). Explain the differences between both images based on the plot.

2.4. (1 point) Write a function that computes the average of the power spectrum at each angle within a specified range of spatial frequencies, where each angle corresponds to a direction from the center of the spectrum. As a starting point, re-implement the template shown in the Figure 1 below. Use the same configuration for the angular bin size and frequency range. Compute the average power spectrum across the different angles for `'bigben_cropped_gray.png'` and the random noise image.
Deliverables: Include the code for the function. Plot your re-implementation of the template. Plot the averaged power spectrum for both images over the angles. Describe what you see in the plot and the differences between the angular distribution of their power spectrum.

2.5. (1 point) Spatial derivatives may be written as the multiplication of a kernel in the frequency domain. Using the convolution theorem and a derivative kernel specified in the frequency domain, implement a function which takes as input two derivative orders, one for the x- and one for the y-directions, as well as a 2-dimensional image, and returns the corresponding partial derivative of the image.
Deliverables: Include a code snippet for your function in the report. Include illustrations of the function applied to an image of your choice at different combinations of orders of derivatives. The illustrations should include a colorbar.

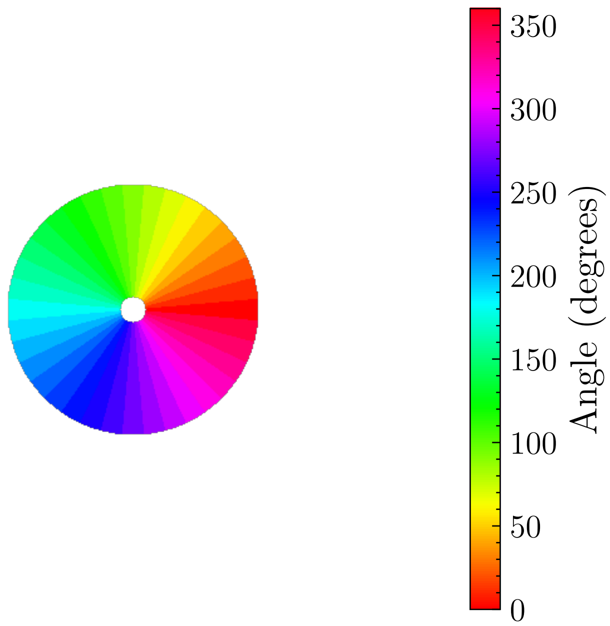

Figure 1: Angular Bins with 10° bins and spatial frequency range of [10-100]