Lecture 3

Gidon Rosalki

2025-11-02

1 2D Fourier

We are aiming to transform the image domain, with a natural basis, comprised of real numbers, into the frequency domain, with a Fourier basis, comprised of complex numbers:

$$f(x,y) \to F(u,v)$$

The image, and transform coefficients are arranged in a 2D array, with both image f, and Fourier transform F, being cyclic / periodic.
2D FT:

$$F(u,v) = \sum_{y=0}^{N-1}\sum_{x=0}^{N-1} f(x,y)\, e^{-\dfrac{2\pi i(ux+vy)}{N}}$$

Inverse 2D Fourier:

$$f(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} F(u,v)\, e^{-\dfrac{2\pi i(ux+vy)}{N}}$$

One may consider taking the Fourier transform of the image, getting the relative frequencies of various functions applied to the picture. For 5x5 bases, ie $-2 \leq u, v \leq 2$:

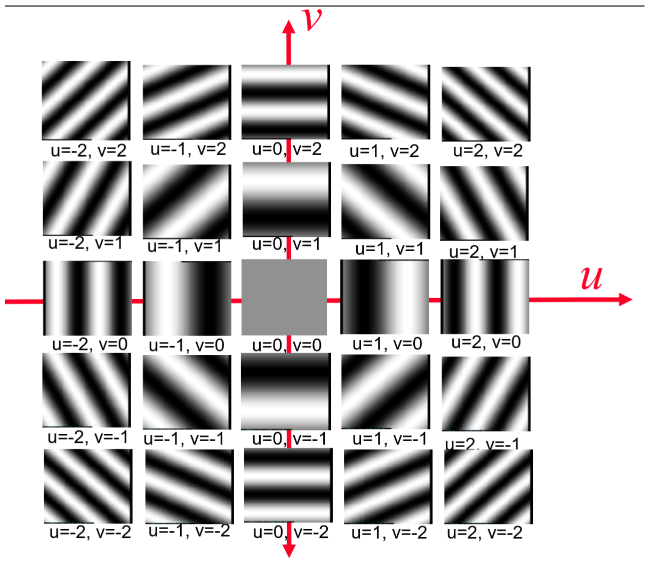

Figure 1: 5x5

Each square represents the function $\sin\left(\frac{2\pi}{N}(ux+vy)\right)$. So here there are 25 basis functions for each image of 5x5, with each function being a 5x5 matrix. $(0,0)$ is at the centre of the image, where -1 is black, +1 is white, and grey is 0. The original image is a weighted sum of all basis functions, where the basis function for frequency (u,v) is multiplied by the complex number $F(u,v)$ specifying both the amplitude, and the phase (shift).

The *Fourier spectrum* is as follows:

- Fourier coefficient (complex number): $F(u) = R(u) + Im(u)$
- Fourier spectrum (positive number): $|F(u)| = \sqrt{R^2(u) + Im^2(u)}$

- Fourier phase (angle): $\theta(u) = \tan^{-1}\left(\frac{Im(u)}{R(u)}\right)$
- Fourier coefficient (complex number): $F(u) = |F(u)| e^{i\theta}$

To display Fourier Spectrum as Picture, one needs to:

1. Compute $\log(|F(u)| + 1)$
2. Scale to full grey-level range (E.g. 0..255 or 0..1)
3. Move $(u = 0, v = 0)$ to the centre of image (Shift by N/2)

1.1 Translation

$$f(x - x_0, y - y_0) \Leftrightarrow F(u, v)\, e^{\frac{2\pi i (ux_0 + vy_0)}{N}}$$
$$F(u - u_0, v - v_0) \Leftrightarrow f(x, y)\, e^{\frac{-2\pi i (u_0 x + v_0 y)}{N}}$$

The Fourier spectrum is invariant to image translation, resulting in the same Fourier Spectrum for these 2 different images:

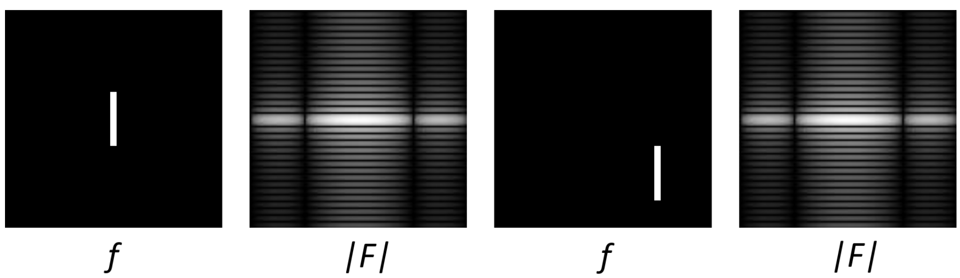

Figure 2: Spectrum invariance

1.2 Decomposition

We can decompose the Fourier transform into computation of the 1D Fourier of each column, and then on that, we compute the 1D Fourier on each row:

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)\, e^{\frac{-2\pi i (ux + vy)}{N}}$$

$$e^{\frac{-2\pi i (ux + vy)}{N}} = \left(e^{\frac{-2\pi iux}{N}}\right)\left(e^{\frac{-2\pi ivy}{N}}\right)$$

$$\implies F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \left(e^{\frac{-2\pi iux}{N}}\right) \sum_{y=0}^{N-1} f(x, y)\, e^{\frac{-2\pi ivy}{N}}$$

$$F(x, v) = \sum_{y=0}^{N-1} f(x, y)\, e^{\frac{-2\pi ivy}{N}}$$

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{\frac{2\pi iux}{N}} F(x, v)$$

So, to conclude, the 2D Fourier Transform can be computed using 1-D Fourier, by computing 1D Fourier on each column, and then on the result, computing 1D Fourier on each row. Therefore, the 1D Fourier Transform is enough to compute Fourier of **any** dimension.

1.3 Periodicity and Symmetry

$$
\begin{aligned}
F(u,v) &= F(u+N,v) \\
&= F(u,v+N) \\
&= F(u+N,v+N) \\
F(u,v) &= F^*(-u,-v) \ \left((a+bi)^* = (a-bi)\right) \\
|F(u,v)| &= |F(-u,-v)|
\end{aligned}
$$

1.4 Example

Let us consider the Fourier spectrum of $\cos\left(\frac{2\pi nx}{N}\right)$ (ie, there are N samples):

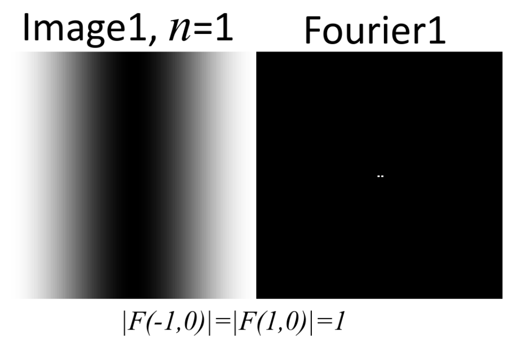

Figure 3:

The two points are the two options where the Fourier spectrum of that coordinate is 1. This can be seen when we repeat this for more images:

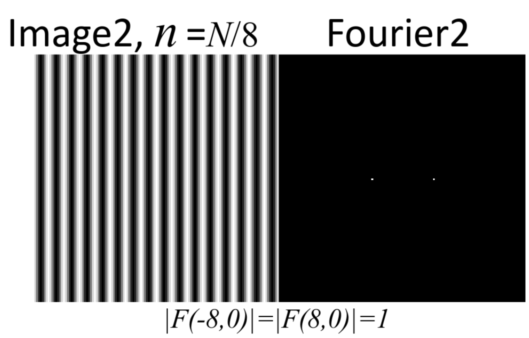

Figure 4:

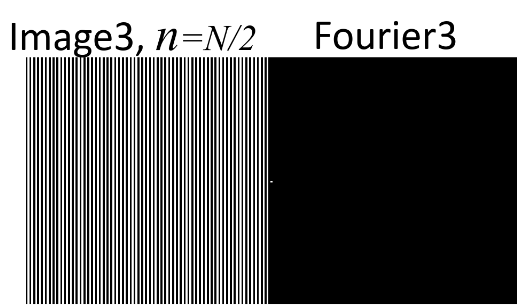

Figure 5: $\left|F\left(-\frac{N}{2},0\right)\right| = \left|F\left(\frac{N}{2},0\right)\right| = 1$

1.5 Linearity

Let ϕ be the Fourier transform

$$\Phi\left(f_1\left(x,y\right)+f_2\left(x,y\right)\right)=\Phi\left(f_1\left(x,y\right)\right)+\Phi\left(f_2\left(x,y\right)\right)$$
$$\Phi\left(a\cdot f\left(x,y\right)\right)=a\cdot\Phi\left(f\left(x,y\right)\right)$$
$$\Phi\left(f\left(ax,by\right)\right)=\frac{1}{|ab|}F\left(\frac{u}{a},\frac{v}{b}\right)$$

1.6 Discrete derivatives using Fourier

Derivatives are defined over continuous functions, so how can we compute the derivative of a sequence of numbers? One possibility is to use the Fourier transform:

$$f\left(x\right)=\sum_u F\left(u\right)e^{\frac{2\pi uix}{N}}$$

The F term is constant, and then we may differentiate the e term.

$$f'\left(x\right)=\left(\sum_u F\left(u\right)e^{\frac{2\pi iux}{N}}\right)$$
$$=\sum_u F\left(u\right)\left(e^{\frac{2\pi iux}{N}}\right)'$$
$$=\frac{2\pi i}{N}\sum_u F\left(u\right)e^{\frac{2\pi iux}{N}}$$

So the derivatives **increase** the high frequencies, since $F\left(u\right)\to uF\left(u\right)$. The highest frequencies are always the noise.

To compute the x derivative of $f\left(x,y\right)$ using Fourier, we

1. Compute the Fourier transform $F\left(u,v\right)$
2. Multiply the Fourier coefficient $F\left(u,v\right)$ by $\frac{2\pi i}{N}u$
3. Compute the inverse Fourier Transform

and to compute the y derivative of $f\left(x,y\right)$:

1. Compute the Fourier transform $F\left(u,v\right)$
2. Multiply the Fourier coefficient $F\left(u,v\right)$ by $\frac{2\pi i}{N}v$
3. Compute the inverse Fourier Transform

The derivative can also just be a Fourier filter:

$$f'\left(x\right)=\frac{2\pi i}{N}\sum_u uF\left(u\right)e^{\frac{2\pi iux}{N}}$$

We multiply the Fourier Coefficient $F\left(u\right)$ with $\frac{2\pi iux}{N}$. This amplifies higher frequencies (and noise, since noise normally has higher frequencies).

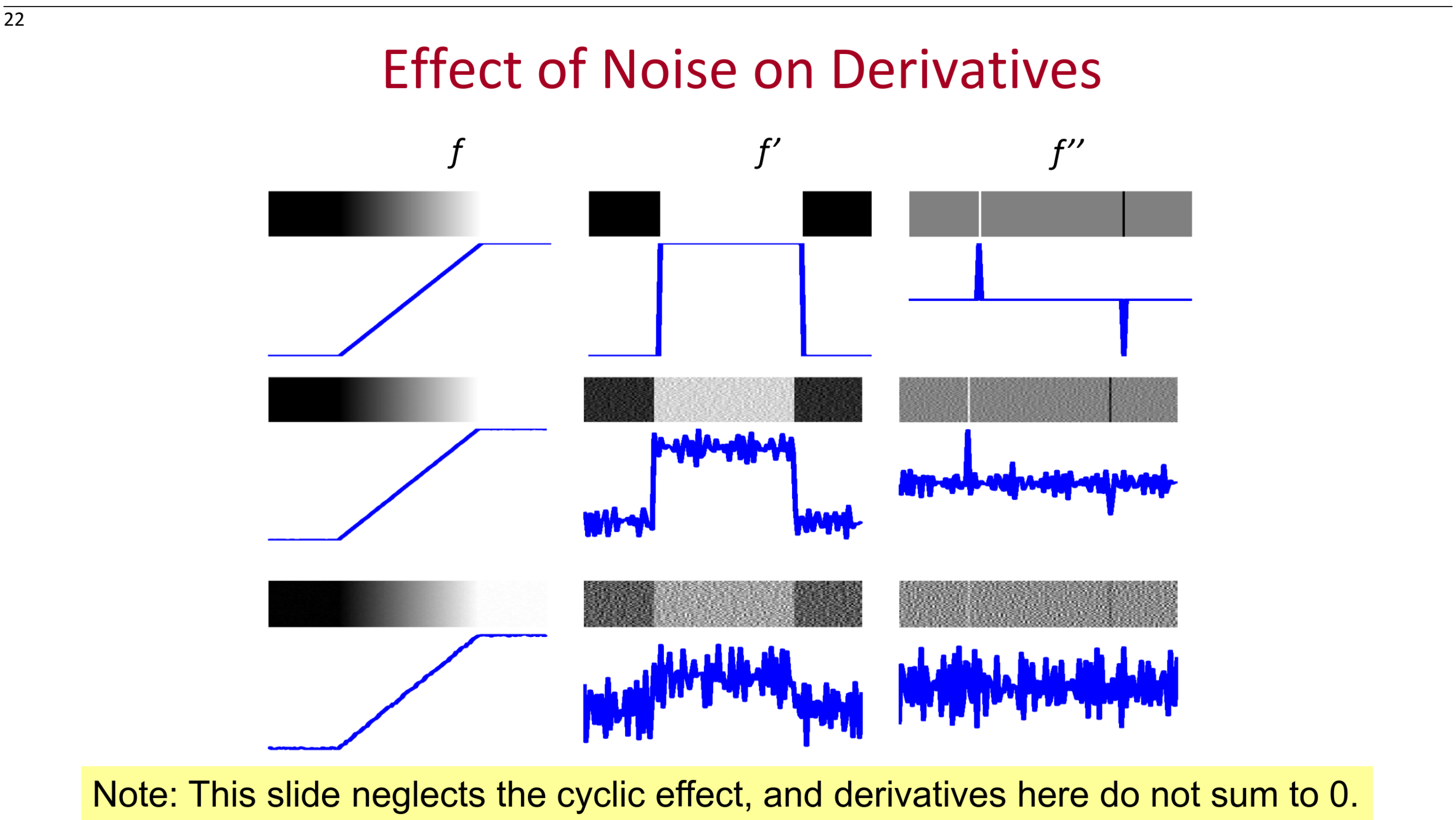

Figure 6: Effect of noise

We see in the first set, with no noise, The second derivative has two discrete spikes where the first derivative changes, whereas in the third example, it is nigh on impossible to find the spikes.

Given an original image, and its Fourier transform:

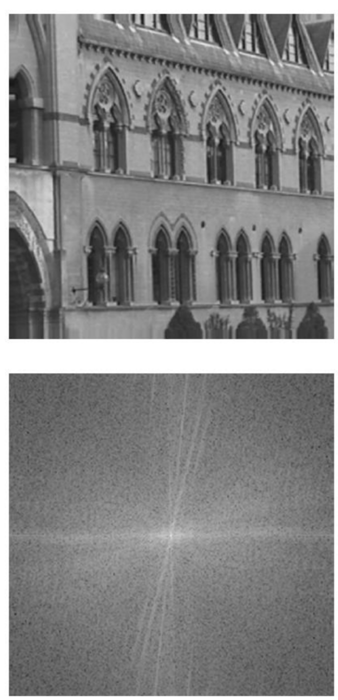

Figure 7: Original

a “low pass”, where we only pass the low frequencies, results in a blurred image,

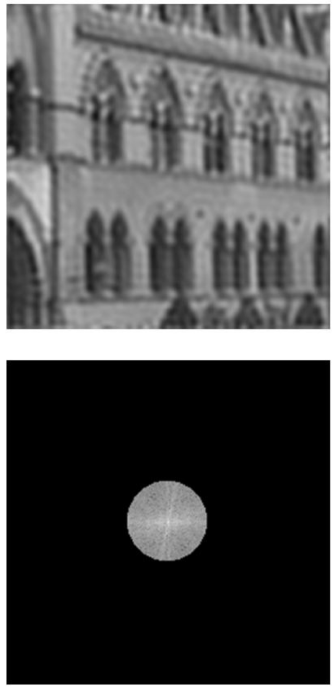

Figure 8: Low pass

The high pass only passes through the high frequencies:

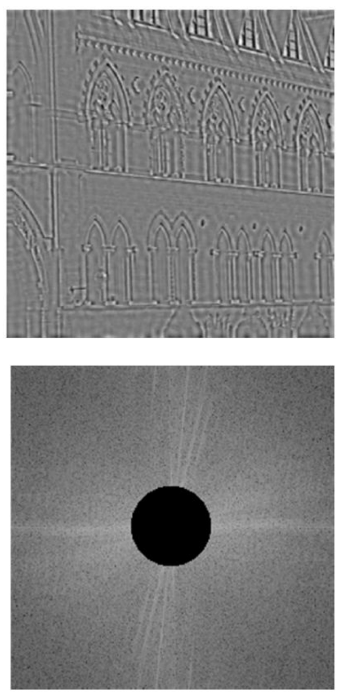

Figure 9: High pass

1.7 Aliasing

Sampling can result in aliasing. If we sample at 1.5π:

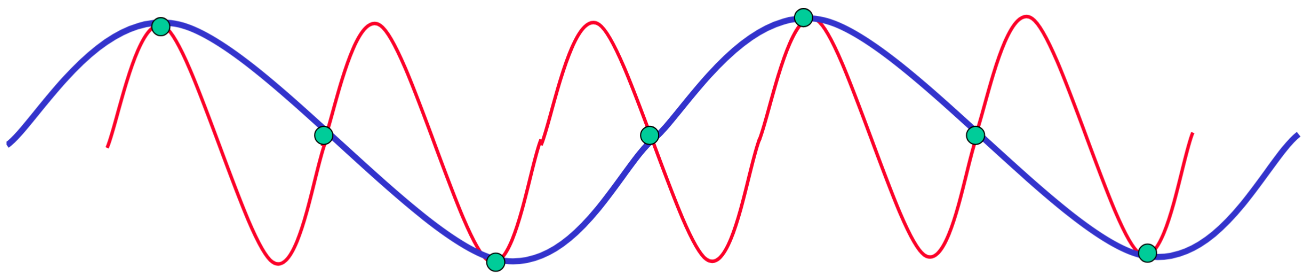

Figure 10: Aliasing

Then as we can see, we may wind up with a wave (the blue one) which is drastically different from our source (the red one). To avoid aliasing, the distance between the sample points should be **less than** $\frac{1}{2}$ of the wavelength (Niquest).

VERY IMPORTANT Sampling:
In order to avoid aliasing, when you want to sample a photo, **blur it first**! (As in, perform a low pass filter, to reduce the highest frequencies). This avoids aliasing, since this way we do not need too high a frequency for measuring. One does **NOT** shrink an image by sampling every other image, but rather you blur, and then sample.

1.8 Signal resizing

For example, we want to resize from N samples to $\frac{N}{2}$ samples:

1. Compute Fourier (N samples to N coefficients)
2. Bring $u = 0$ to the centre (FFT shift)
3. Crop Fourier from N to $\frac{N}{2}$ coefficients - remove the $\frac{N}{2}$ high frequencies
4. Compute the Inverse Fourier ($\frac{N}{2}$ coefficients to $\frac{N}{2}$ samples)

To reduce an image:

1. Blur, and subsample every 2nd pixel, in every 2nd row
2. Use Fourier, compute $N \times N$, and shift $(0, 0)$ to the center. Crop Fourier (eg to $\frac{N}{2} \times \frac{N}{2}$), and compute the inverse Fourier

To compute we do something very similar, and have it behave similarly to reconstructing an image from a low pass. We compute the Fourier ($N \times N$), pad it with zeroes ($2N \times 2N$), and then compute the inverse Fourier.