
Transforms PT II

Image Processing Extra Material

Dr. Clément Bled

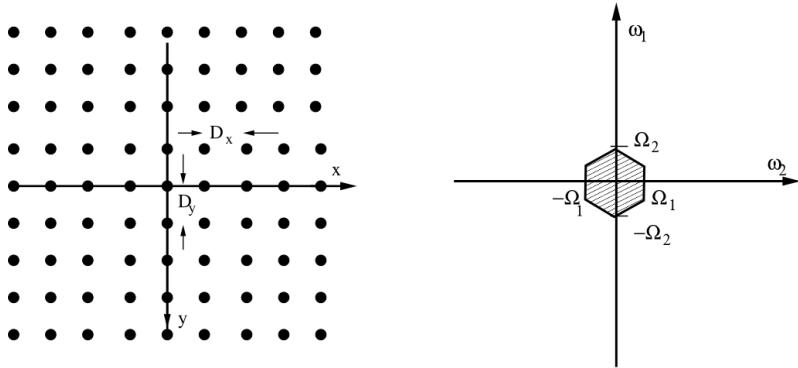


Figure 1: **Visualising the components of 2D sampling.** The left graph depicts the spatial sampling function $s(x, y)$ as a uniform grid of impulse points (Dirac deltas) spaced by intervals D_x and D_y . The right graph illustrates the frequency spectrum $F(\omega_1, \omega_2)$ of the original continuous image, represented as a shaded hexagonal region bounded by maximum frequencies $\pm\Omega_1$ and $\pm\Omega_2$.

1 2D Sampling Theorem

Have signal $f(x, y) \leftrightarrow F(\omega_1, \omega_2)$, we want to represent using a discrete sequence $f[m, n]$, where $f[m, n] = f(mD_x, nD_y)$. So ideal model multiplies $f(x, y)$ by a grid of delta functions $s(x, y)$.

$$s(x, y) = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} \delta(x - nD_x, y - n_2D_y), \quad (1)$$

Where D_x, D_y are the distance between points, i.e. the horizontal/vertical period.

As we said sampling is $f_s(x, y) = f(x, y)s(x, y)$. What's the relationship between $F(\omega_1, \omega_2)$ and $F_s(\omega_1, \omega_2)$?

$s(x, y)$ is the dirac comb.

In the Fourier domain we have:

$$F_s(\omega_1, \omega_2) \circledast S(\omega_1, \omega_2) \quad (2)$$

Need to get FT of $S(\omega_1, \omega_2)$.

If a signal is infinite, it can be represented by a Fourier **series**. Remember 2D Fourier series:

$$s(x, y) = \sum_{k_1=\infty}^{\infty} \sum_{k_2=\infty}^{\infty} = a_{k_1, k_2} e^{j(k_1 \omega_1^0 x + k_2 \omega_2^0 y)} \quad (3)$$

Infinite sum of sinusoids where each wave (ω_1, ω_2) has weight a_{k_1, k_2} . We know the grid spacing is D so the frequencies are going to fit inside those.

- The base frequency for x, ω_1 is therefore $\omega_1^0 = \frac{2\pi}{D_x}$.

- The base frequency for y , ω_2 is therefore $\omega_2^0 = \frac{2\pi}{D_y}$.

The other waves in the series are just going to be integer multiples (k_1, k_2) of the base frequencies.

Ok, you already know to get the fourier series coefficients you integrate the signal over one period and divide by the area of the period:

$$a_{k_1, k_2} = \frac{1}{D_x D_y} \int_{-Dy/2}^{Dy/2} \int_{-Dx/2}^{Dx/2} s(x, y) e^{-j(k_1 \omega_1^0 x + k_2 \omega_2^0 y)} dx dy \quad (4)$$

In this integration window of one period the repeating grid $s(x, y)$, contains a single impulse at the origin. So equivalently:

$$\begin{aligned} a_{k_1, k_2} &= \frac{1}{D_x D_y} \int_{-Dy/2}^{Dy/2} \int_{-Dx/2}^{Dx/2} \delta(x, y) e^{-j(k_1 \omega_1^0 x + k_2 \omega_2^0 y)} dx dy \\ &= \frac{1}{D_x D_y} \quad \text{By the sifting property of the } \delta \text{ function} \end{aligned} \quad (5)$$

a can be plugged back into eqn 3:

$$s(x, y) = \frac{1}{D_x D_y} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} e^{j(k_1 \omega_1^0 x + k_2 \omega_2^0 y)} \quad (6)$$

Reminder: The Fourier transform of a complex exponential is a Dirac Delta/impulse, with $e^{j\Omega x}$ having a frequency of Ω which will be a single impulse at that frequency: $\delta(\omega - \Omega)$.

Finally:

$$S(\omega_1, \omega_2) = \frac{1}{D_x D_y} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \delta(\omega_1 - k_1 \omega_1^0, \omega_2 - k_2 \omega_2^0) \quad (7)$$

So coming back to the sampled signal:

$$\begin{aligned} F_s &= F(\omega_1, \omega_2) \circledast S(\omega_1, \omega_2) \\ &= \frac{1}{D_x D_y} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} F(\omega_1 - k_1 \omega_1^0, \omega_2 - k_2 \omega_2^0) \end{aligned} \quad (8)$$

What do we learn from this?

- **Replication:** Convoluting any function by a delta function shift the function to the location of the delta.
- **Scaling:** The coefficients are inversely proportional to the sampled area. Large area = low density, small weight, Small area = high density, large scaling factor.

1.1 Sampling Theorem

Right so we know the signal is repeated every ω/k steps. We want to avoid overlap between these harmonics.

If we **lower the sampling rate, by making D larger**, the **sampling frequency $\omega = \frac{2\pi}{D}$ get smaller**. Visually, the spectrums (hexagons) get closer together, eventually colliding. This called **Aliasing**, the negative part of the closest aliases/fundamentals/harmonics crash into the base frequency spectrum.

Nyquist's Theorem in 2D states: $\omega_1^0 > 2\Omega_1$ AND $\omega_2^0 > 2\Omega_2$ for no aliasing to occur.
“Sampling frequency w_0 must be greater than twice the maximum frequency in the image”.

This is a practical problem that comes into play when sampling from analogue to digital: the max frequency a camera can capture is dictated by the pixel pitch. A low pass filter is needed to avoid sampling those higher frequencies that would cause frequency overlap.

Same principle when resizing digital images. When you shrink an image, the new sampling rate is lower, the sampling rate drops to $2\pi/N \cdot D = \omega^0/N$. We need to make sure the max frequency is less than half ω^0/N . Apply a LPF.

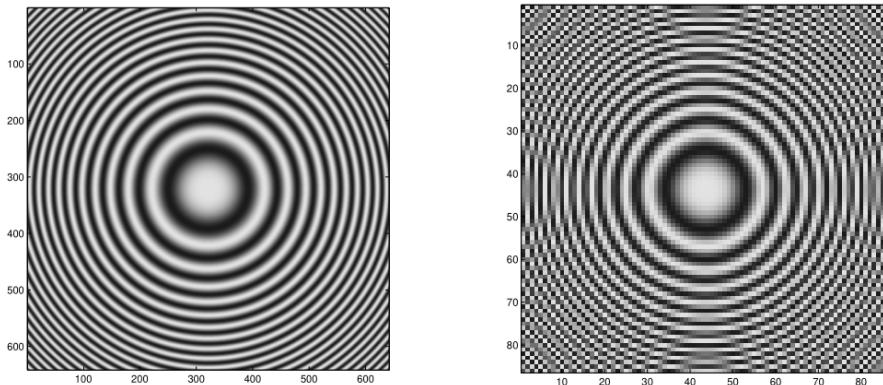


Figure 2: Aliasing example. worst at high frequency (edges).

2 2D Discrete Fourier Transform

2D DFT is similar to the 1D version and is separable.

Given image of size $M \times N$:

Theorem 2.1.

$$\begin{aligned}\text{FFT : } F[h, k] &= \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f[n, m] e^{-j(\frac{2\pi}{N} nh + \frac{2\pi}{M} mk)} \\ \text{Inverse FFT : } f[n, m] &= \frac{1}{NM} \sum_{h=0}^{N-1} \sum_{k=0}^{M-1} F[h, k] e^{-j(\frac{2\pi}{N} nh + \frac{2\pi}{M} mk)}\end{aligned}\quad (9)$$

2.1 Properties of 2D FFT

Periodicity Like in the 1D case, the 2D FFT is infinitely periodic in the direction of ω_1 and ω_2 .

$$F(\omega_1, \omega_2) = F(\omega_1 + k_1 M, \omega_2) = F(\omega_1, \omega_2 + k_2 N) = F(\omega_1 + k_1 M, \omega_2 + k_2 N) \quad (10)$$

and

$$f(x, y) = f(x + k_1 M, y) = f(x, y + k_2 N) = f(x + k_1 M, y + k_2 N) \quad (11)$$

where k_1 and k_2 are integers.

The data in the interval 0 to $M - 1$ is actually two half periods meeting at $M/2$, with the lower part of the spectrum appearing at the higher frequencies.

This is not super meaningful when visualised so we reorder it using the translation:

$$F(x) e^{j2\pi(u_0 x/M)} \Leftrightarrow F(u - u_0) \quad (12)$$

This moves the origin to u_0 , so in our case we let $u_0 = M/2$, the exponential becomes $e^{j\pi x}$, which is equal to $(-1)^x$ because x is an integer. In 1D we have:

$$f(x)(-1)^x \Leftrightarrow F(u - M/2) \quad (13)$$

So multiplying $f(x)$ by $(-1)^x$ shift the data so that $F(u)$ becomes centers on the interval $[0, M - 1]$. In 2D it can be similarly be proved that:

$$f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2), \quad (14)$$

Shifting the rectangle as shown in Fig. 3.

Symmetry Similarly, if $f(x, y)$ is real, the Fourier transform exhibits conjugate symmetry, it is centrally symmetric.

$$F(\omega_1, \omega_2) = F^*(-u, -v) \quad (15)$$

or

$$|F(u, v)| = |F(-u, -v)| \quad (16)$$

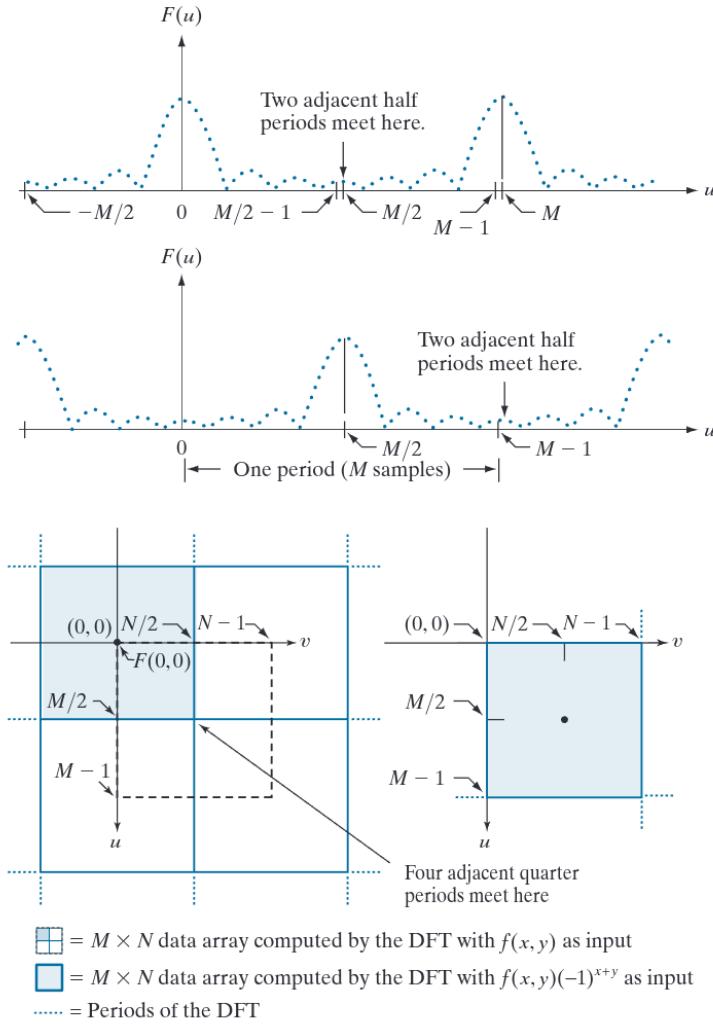


Figure 3: centering the Fourier Spectrum

Transform scan be expressed as matrix operations on rows, then columns:

$$F = W^T I W$$

3 Downsampling

We want to downsample an image by N . Same as taking the discrete time signal $f[m, n]$ and sampling every N samples.

$$f_N[mN, nN]$$

From our sampling discussion, we know that the original image sampling rate is $[\Omega_0, \Omega_0] = [(2\pi/D), (2\pi/D)]$ radians per pel. See Fig. 4.

We now want to reduce this to $[\Omega_0/N, \Omega_0/N]$ to get $f_N[m, n]$

Remember aliasing and periodicity. In the frequency domain, we have infinite copies of

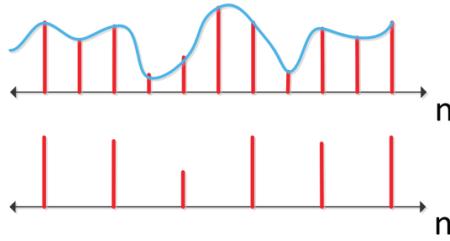


Figure 4: Top: Signal sampled at period of T . Bottom: Same signal downsampled by a factor of 2.

the spectrum at integer multiples of Ω . We don't want these to overlap. Nyquist frequency is half our sampling rate. If we sample every second pixel, the Nyquist frequency decreases. The high frequencies are now greater than Nyquist.

Before downsampling:

Sampling interval: D

$$\text{Sampling rate: } \Omega_0 = \frac{2\pi}{D}$$

$$\text{Nyquist Limit: } \frac{\pi}{D}$$

After downsampling:

Sampling interval: $2D$

$$\text{Sampling rate: } \Omega = \frac{\Omega_0}{N} = \frac{2\pi}{D} \frac{1}{2} = \frac{2\pi}{2D} = \frac{\pi}{D}$$

$$\text{Nyquist Limit: } \frac{\Omega_0}{N} \frac{1}{2} = \frac{\Omega}{2} = \frac{\pi}{2D}$$

According to slides: "The spectrum of the sampled signal $f[m, n]$ contains scaled copies of the original spectrum $F(\omega_1, \omega_2)$ at integer multiples of $[\Omega_0, \Omega_0]$. If the sampling rate is reduced to $[\Omega_0/N, \Omega_0/N]$ the replicas move closer together."

Frequencies greater than $\frac{\Omega_0}{2N} = \frac{\pi}{2T}$ will cause aliasing.

Anyway, eqn. 4 used to obtain $f[m, n]$. We now want to resample the original at a higher frequency. SO we need to re-estimate the original spectrum $F(\omega_1, \omega_2)$. Can be done by cropping at $(0, 0)$ and cutting off other replications and re-scaling.

This is done with a LPF $H_0(\omega_1, \omega_2) \rightleftharpoons (x, y)$ with the frequency response:

$$H_0(\omega_1, \omega_2) = D^2 \text{rect} \left(\frac{\omega_1}{2\Omega_0}, \frac{\omega_2}{2\Omega_0} \right)$$

$$= \begin{cases} D^2 & \text{for } |\omega_1| < \Omega_0 \text{ and } |\omega_2| < \Omega_0 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

Applying $F(\omega_1, \omega_2) = H_0(\omega_1, \omega_2)F_s(\omega_1, \omega_2)$ gives us the spectrum of the analogue signal. Next we want to resample. To avoid aliasing an ideal low-pass filter $H_N(\omega_1, \omega_2)$ is used

with cutoff at $[\pm\Omega_0/(2N), \pm\Omega_0/(2N)]$ with frequency response:

$$H_N(\omega_1, \omega_2) = \text{rect}\left(\frac{\omega_1 N}{2\Omega_0}, \frac{\omega_2 N}{2\Omega_0}\right)$$

$$= \begin{cases} 1 & \text{for } |\omega_1| < \frac{\Omega_0}{N} \text{ and } |\omega_2| < \frac{\Omega_0}{N} \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

Combining both we get:

$$H(\omega_1, \omega_2) = H_0(\omega_1, \omega_2)H_N(\omega_1, \omega_2)$$

$$= D^2 \text{rect}\left(\frac{\omega_1}{2\Omega_0}, \frac{\omega_2}{2\Omega_0}\right) \text{rect}\left(\frac{\omega_1 N}{2\Omega_0}, \frac{\omega_2 N}{2\Omega_0}\right)$$

$$= D^2 \text{rect}\left(\frac{\omega_1 N}{2\Omega_0}, \frac{\omega_2 N}{2\Omega_0}\right)$$

$$= \begin{cases} D^2 & \text{for } |\omega_1| < \frac{\Omega_0}{N} \text{ and } |\omega_2| < \frac{\Omega_0}{N} \\ 0 & \text{otherwise} \end{cases}$$

In the spatial domain impulse response:

$$h(x, y) = \frac{1}{4\pi^2} \int \int D^2 \text{rect}\left(\frac{\omega_1}{2\Omega}, \frac{\omega_2}{2\Omega}\right) e^{j(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2$$

$$= \frac{D^2}{4\pi^2} \int_{-\Omega}^{\Omega} \int_{-\Omega}^{\Omega} e^{j(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2$$

$$= \frac{D^2}{4\pi^2} \left(\frac{2}{x} \sin(x\Omega) \right) \int_{-\Omega}^{\Omega} e^{j(\omega_2 y)} d\omega_2$$

$$= \frac{D^2}{4\pi^2} \frac{2}{x} \sin(x\Omega) \frac{2}{y} \sin(y\Omega)$$

$$= \frac{D^2}{4\pi^2} \underbrace{2\Omega \text{sinc}(x\Omega) 2\Omega \text{sinc}(y\Omega)}_{\text{Separable}} \quad (20)$$

Need to make the filter in digital domain. So has to be sampled. We want sampled at $y = hD$ $x = kD$ and $\Omega = \Omega_0/2N = \pi/(DN)$:

$$h[m, n] = D^2 \frac{1}{4\pi^2} \left[\frac{2\pi}{DN} \text{sinc}\left(\frac{mD\pi}{DN}\right) \right] \left[\frac{2\pi}{DN} \text{sinc}\left(\frac{nD\pi}{DN}\right) \right]$$

$$= \left[\frac{1}{N} \text{sinc}\left(\frac{m\pi}{N}\right) \right] \left[\frac{1}{N} \text{sinc}\left(\frac{n\pi}{N}\right) \right] \quad (21)$$

So the filter doesn't rely on the original sampling frequency, Ω_0 , only the sub-sampling factor N .

The sinc function is infinitely long so the filter is IIR. So we need to truncate it to a length of N . But truncating would create a sharp discontinuity that will cause a ripple artefact.

Need to use hamming window:

$$h[m, n] = \left[\frac{w[m]}{N} \text{sinc} \left(\frac{m\pi}{N} \right) \right] \left[\frac{w[n]}{N} \text{sinc} \left(\frac{n\pi}{N} \right) \right] \quad (22)$$

where w is the Hamming window. In Fig. 5 we see the FIR filter with rectangle and hamming window. Hamming sacrifices slower roll off to remove the ripple.

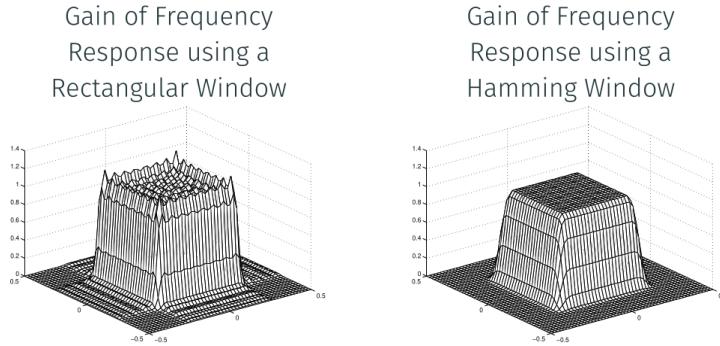


Figure 5: Left: truncate/rectangle/no window. Right: Hamming window.

Last slide says Sinc function is long and expensive, similar results can be achieved with a Gaussian filter which is also separable with less ringing.

$$h[m, n] = \exp - \left(\frac{m^2 + n^2}{2\sigma^2} \right), \quad (23)$$

where σ affects the bandwidth. Need to specify the number of taps and normalise the filter coefficients so that they sum to 1. $h = h/\text{sum}(h)$. With this can get away with shorter filter (15 taps).

Zooming or upsampling is *much* harder.