Topic 3: Digital Representation and Sampling

Aim:
These two lectures cover the main theoretical background to representation, storage and sampling of digital images.

Contents:

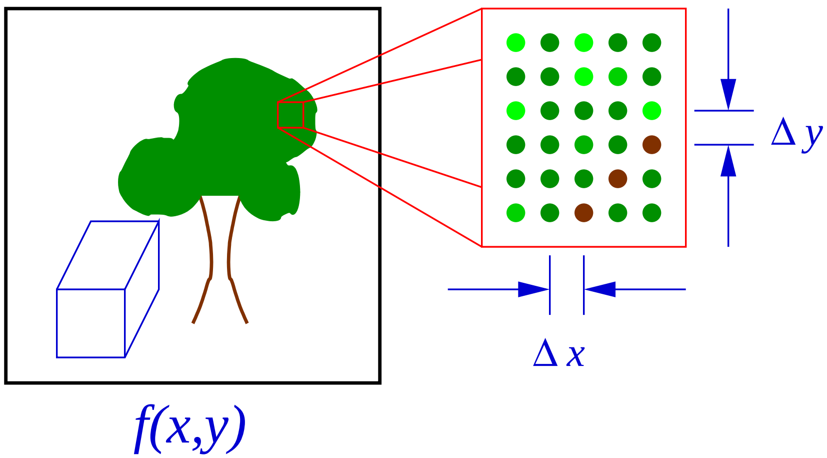
- Representation of a Digital Image
- Discrete Fourier Transform
- Properties of Discrete F.T.
- Sampling Theory
- Reconstruction & Interpolation
- Summary

Digital Images

Real Space: Represented by a two-dimensional array of numbers by, sampling $f(x,y)$ at points

$$x_0 + i\Delta x\,,\, y_0 + j\Delta y) \quad \text{where } i \;\&\; j = 0, 1, \ldots, N-1$$

Where Δx and Δy are the x and y sampling intervals.



Gives an N-by-N array of samples (numbers)

$$f(i,j) \quad \text{where } i \;\&\; j = 0, 1, \ldots, N-1$$

which we will hold in a computer as a Two-Dimensional array.

Digital Images I

For example:

Picture contains 128^2 points.

The criteria for $\Delta x, \Delta y$ depends on the imaging system and will be covered later.

Digital Images II

Fourier Space:

The two-dimensional Fourier Transform of $f(x,y)$ is given by

$$F(u,v) = \int\int f(x,y)\exp(-2\pi\imath(ux+uy))\,\mathrm{d}x\,\mathrm{d}y$$

Similarly, $F(u,v)$ can be sampled at intervals of Δu and Δv to give:

$$F(k,l) \quad \text{where } k \,\&\, l = 0,1,\ldots,N-1$$

where, *it will be shown* (in the next lectures), that for optimal sampling that:

$$\Delta u = \frac{1}{N\Delta x} \quad \text{and} \quad \Delta v = \frac{1}{N\Delta y}$$

In Fourier space the samples will be Complex, so we will not usually be able to sample $F(k,l)$ directly.

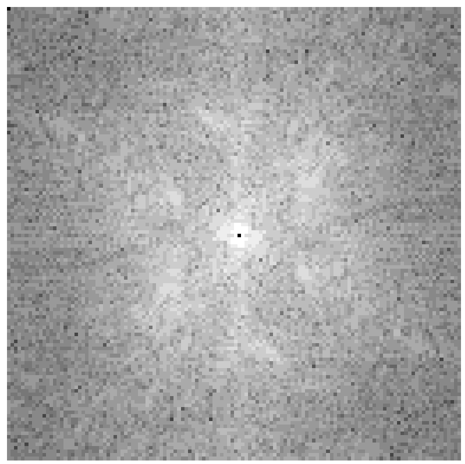
Digital Images III

] If we have the relation between

$$f(i,j) \Leftrightarrow F(k,l)$$

we can calculate $F(k,l)$ from $f(i,j)$.

Normally display $|F(k,l)|^2$.



which the DFT of the toucan.

Discrete Fourier Transform

One Dimensions:

For a continuous function $f(x)$ the Fourier Transform is:

$$F(u) = \int f(x)\exp(-\imath 2\pi u x)\,\mathrm{d}x$$

where for the sampled function, $f(i)$, with N samples, the Discrete Fourier Transform (DFT), is

$$F(k) = \sum_{i=0}^{N-1} f(i)\exp\left[-\imath 2\pi \frac{ki}{N}\right]$$

And the inverse DFT by:

$$f(i) = \frac{1}{N}\sum_{k=0}^{N-1} F(k)\exp\left[\imath 2\pi \frac{ki}{N}\right]$$

by convention the $1/N$ is applied to inverse transform.

Discrete Fourier Transform I

Two Dimensions:

Similarly in Two-Dimensions we have sampled image $f(i,j)$, so that

$$F(k,l) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\exp\left[-\imath 2\pi\left(\frac{ki}{N}+\frac{lj}{N}\right)\right]$$

and similarly the inverse is given by,

$$f(i,j) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} F(k,l)\exp\left[\imath 2\pi\left(\frac{ki}{N}+\frac{lj}{N}\right)\right]$$

again with the $1/N^2$ normalisation applied to the inverse only.

Note: We have assumed the images to be square. If not square same mathematics applies.

The image and its Fourier Transform are both of size $N \times N$.

Properties of One-Dimensional DFT

If the sampled function $f(i)$ is **Real Only**, we can write

$$F(k) = F_R(k) - \imath F_I(k)$$

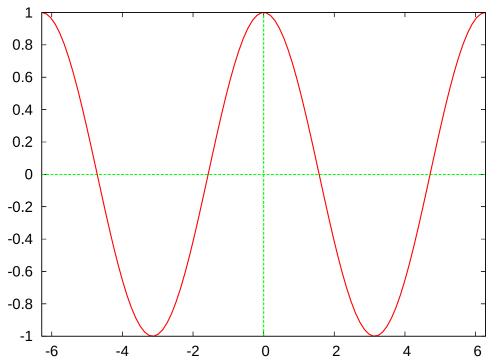
where we have that

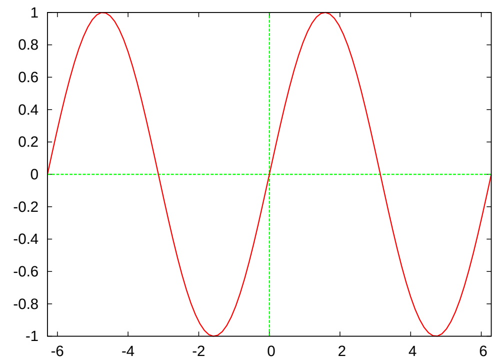
$$\begin{aligned} F_R(k) &= \sum_{i=0}^{N-1} f(i) \cos\left[-\imath 2\pi \frac{ki}{N}\right] \\ F_I(k) &= \sum_{i=0}^{N-1} f(i) \sin\left[-\imath 2\pi \frac{ki}{N}\right] \end{aligned}$$

We have that

$$\begin{aligned} \cos() &\rightarrow \text{Symmetric Function} \\ \sin() &\rightarrow \text{Anti-symmetric Function} \end{aligned}$$

Properties of One-Dimensional DFT





So that

$$F_R(k) \quad \Rightarrow \quad \text{Symmetric Function} \quad \Rightarrow F_R(-k) = F_R(k)$$
$$F_I(k) \quad \Rightarrow \quad \text{Anti-symmetric Function} \quad \Rightarrow F_I(-k) = -F_I(k)$$

Which is true for all **Real inputs**.

Cyclic Properties

In addition $F(k)$ is *cyclic* of period N, so that,

$$F(k \pm nN) = F(k)$$

since

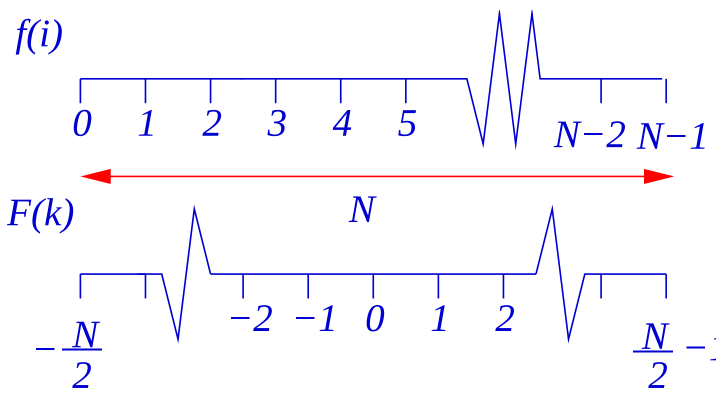
$$\exp\left[-\imath 2\pi \frac{(k \pm nN)i}{N}\right] = \exp\left[-\imath 2\pi \frac{ki}{N}\right]$$

So that k does **not** need to run from $0 \rightarrow N-1$, but *any* range of N sample specify $F(k)$.

Noting that N is always *even*, Typically take

$$F(k) \quad \text{for } k = -\tfrac{N}{2}, \ldots, 0 \ldots, \tfrac{N}{2} - 1$$

so:



This allow us to investigate the symmetry properties of $F(k)$.

Simple Example

Take $N = 4$, we have in Real Space, we have **4** samples,

$$f(i) \quad \text{for } i = 0,1,2,3$$

and in Fourier space,

$$F_R(k) + \imath F_I(k) \quad \text{for } k = -2,-1,0,1$$

which is **4** real components and **4** imaginary.

Real Part:

$$\begin{aligned} F_r(0) &= \sum_{i=0}^{3} f(i) = \text{DC term} \\ F_r(-1) &= F_r(1) \quad \text{Symmetric property} \\ F_R(-2) &= F_R(2) \quad \text{cyclic of period 4} \end{aligned}$$

so we have only **3** independent Real parts, the other given by symmetry.

Simple Example I

Imaginary Part:

$$
\begin{aligned}
F_I(0) &= 0 \quad \text{Since } \sin 0 = 0 \\
F_I(-1) &= -F_i(1) \quad \text{Antisymmetric property} \\
F_I(-2) &= 0 \quad \text{Since } \sin \pi = 0
\end{aligned}
$$

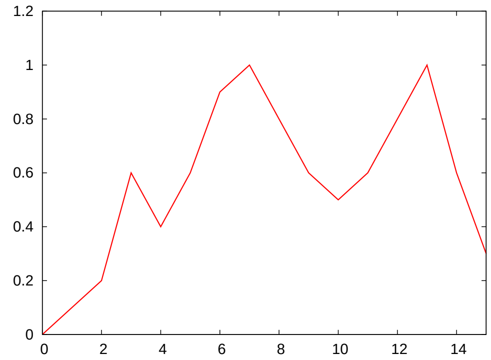
so we have only **1** independent Imaginary part, **1** given by symmetry, and **1** always being Zero.

So total of **4** independent values, **3** real and **1** imaginary.

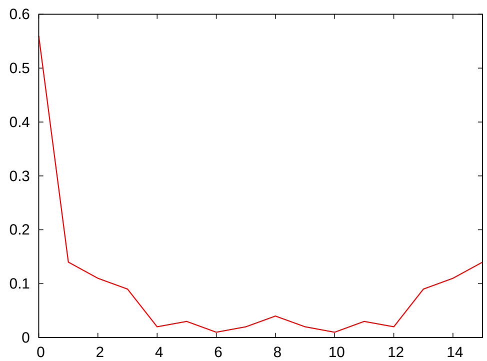
So **4** samples in Real Space give **4** samples in Fourier Space.

Bigger Example

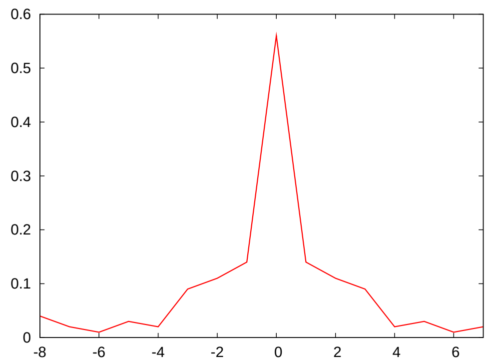
Take $N = 16$, with an input function of:



The modulus of the Fourier Transform.



$F(k)$ for $k = 0, 1, \ldots, 15$



$F(k)$ for $k = -8, \ldots, 0, \ldots, 7$

Bigger Example I

The both ranges of the Fourier Transform contain the same values.

Both Fourier Transforms show the expected symmetry, but it is easier to see and understand in the shifted version.

Note: we have *not* introduced negative frequencies, we have just shifted the DFT to make its structure easier to understand and analyse.

General Case

In the general case of a N point real function $f(i)$, then it DFT, $F(k)$ will have:

$$\begin{aligned} \frac{N}{2}+1 & \quad \text{Real Value} \\ \frac{N}{2}-1 & \quad \text{Imaginary Values} \\ F_i(0) &= F_i(-N/2) = 0 \end{aligned}$$

giving a **Total** of N independent values in Fourier space with the other values given by symmetry properties.

There is the same information in Real and Fourier space, so we *expect* the same number of values in each.

Aside: Useful when calculating DFT, able to use the same storage for Real Space and Fourier Space arrays.

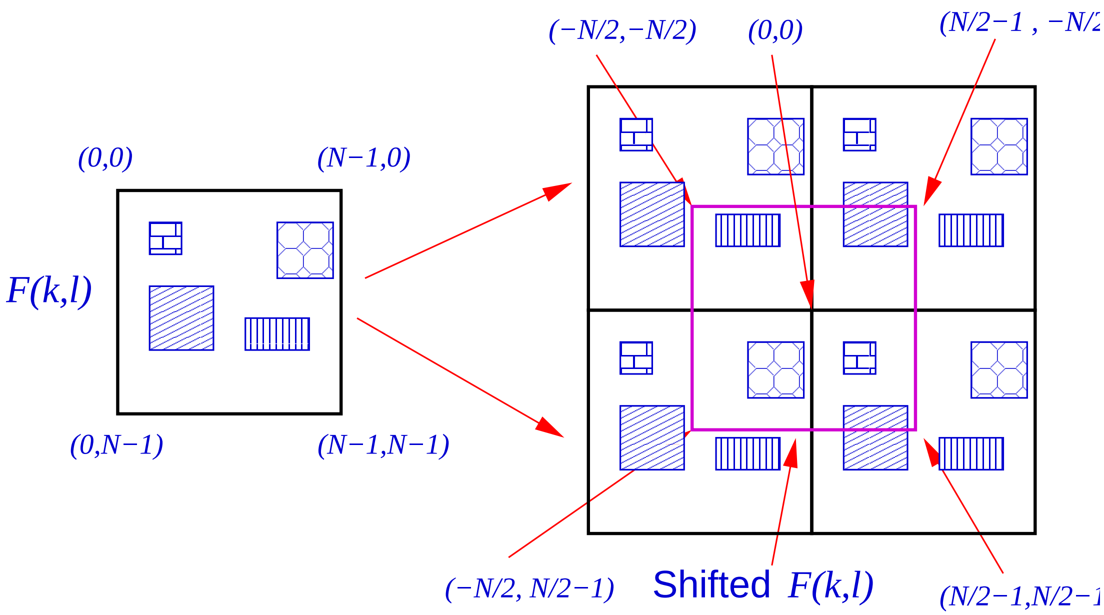
Properties of 2-Dimensional DFT

The $N \times N$ DFT is *cyclic* of period N, in both k and l direction,

$$F(k \pm nN, l \pm mN) = F(k,l)$$

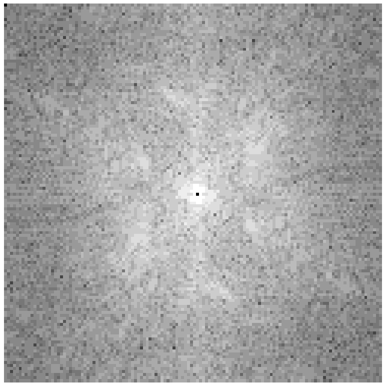
so we can shift the $F(0,0)$ term in two dimensions to give,

$$F(k,l) \quad \text{for } k \,\&\, l = -\frac{N}{2}, \ldots, 0 \ldots, \frac{N}{2} - 1$$



Properties of 2-Dimensional DFT I

This allows us to have the $F(0,0)$ to appear at the centre of the Fourier array.



Aside: The $|F(u,v)|^2$ is identical to the Optical Diffraction pattern. Usually displayed with the bright centre in the middle.

Two-Dimensional Symmetry

For a real input image $f(i,j)$ again we can write the Fourier Transform

$$F(k,l) = F_R(k,l) - \imath F_I(k,l)$$

where, after some effort, we get that:

$$F_R(k,l) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\left[\cos\left(2\pi\frac{ik}{N}\right)\cos\left(2\pi\frac{jl}{N}\right) + \sin\left(2\pi\frac{ik}{N}\right)\sin\left(2\pi\frac{jl}{N}\right)\right]$$

and that

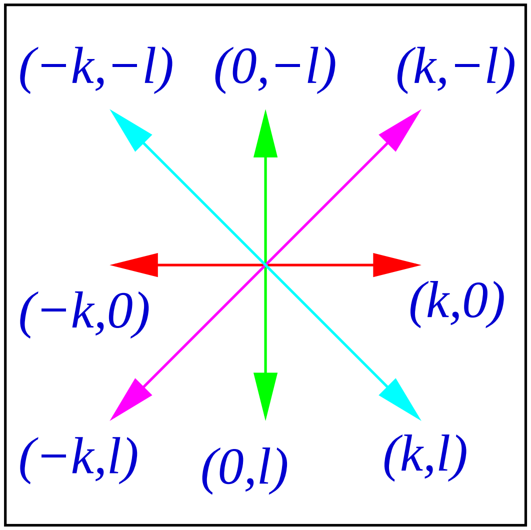
$$F_I(k,l) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\left[\cos\left(2\pi\frac{ik}{N}\right)\sin\left(2\pi\frac{jl}{N}\right) + \sin\left(2\pi\frac{ik}{N}\right)\cos\left(2\pi\frac{jl}{N}\right)\right]$$

Again noting that

$\cos()$ $\rightarrow$ Symmetric Function
$\sin()$ $\rightarrow$ Anti-symmetric Function

Two-Dimensional Symmetry

These have **Point** symmetry properties about the centre, being



When can be written as:

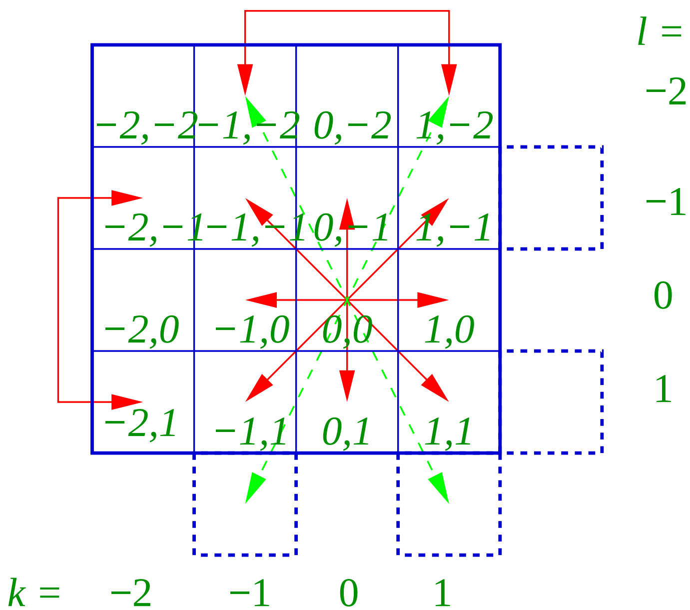
$$
\begin{aligned}
F_R(k,l) &= F_R(-k,-l) \\
F_R(-k,l) &= F_R(k,-l) \\
F_I(k,l) &= -F_I(-k,-l) \\
F_I(-k,l) &= -F_I(k,-l)
\end{aligned}
$$

The major problem is that N is almost always **even** which complicates the symmetry at the edge of the Fourier Plane.

Simple Example in 2-D

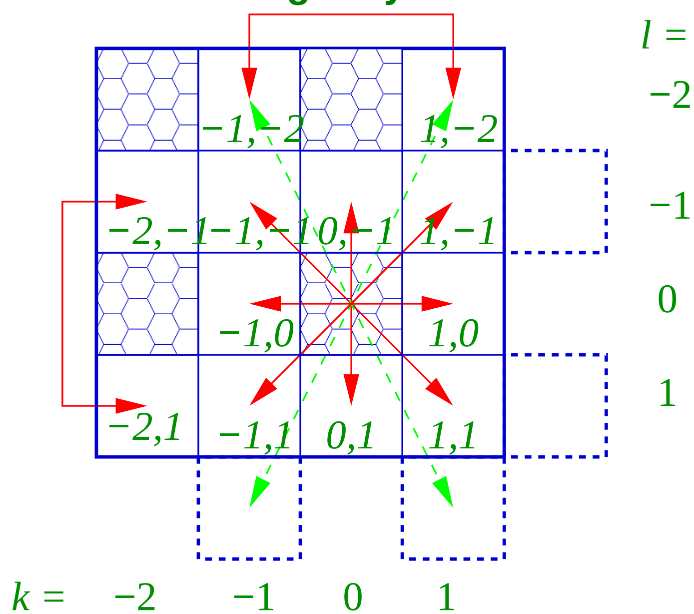
Again take the simple case of a 4×4 image $f(i,j)$,

Real Part



10 Real Values

Imaginary Part



6 Imaginary Values

Giving 10 real values and 6 imaginary values, a total of **16**.

General Two-Dimensional Case

In the general case for the DFT of an $N \times N$ point real image,

$$\frac{N^2}{2} + 2 \quad \rightarrow \quad \text{Real Values}$$
$$\frac{N^2}{2} - 2 \quad \rightarrow \quad \text{Imaginary Values}$$

giving a **Total** of N^2 independent values in the Fourier plane.

Again as you would expect, since N^2 values in Real Space give rise to N^2 values in Fourier Space.

Able to reuse image space in program, important of N is large.

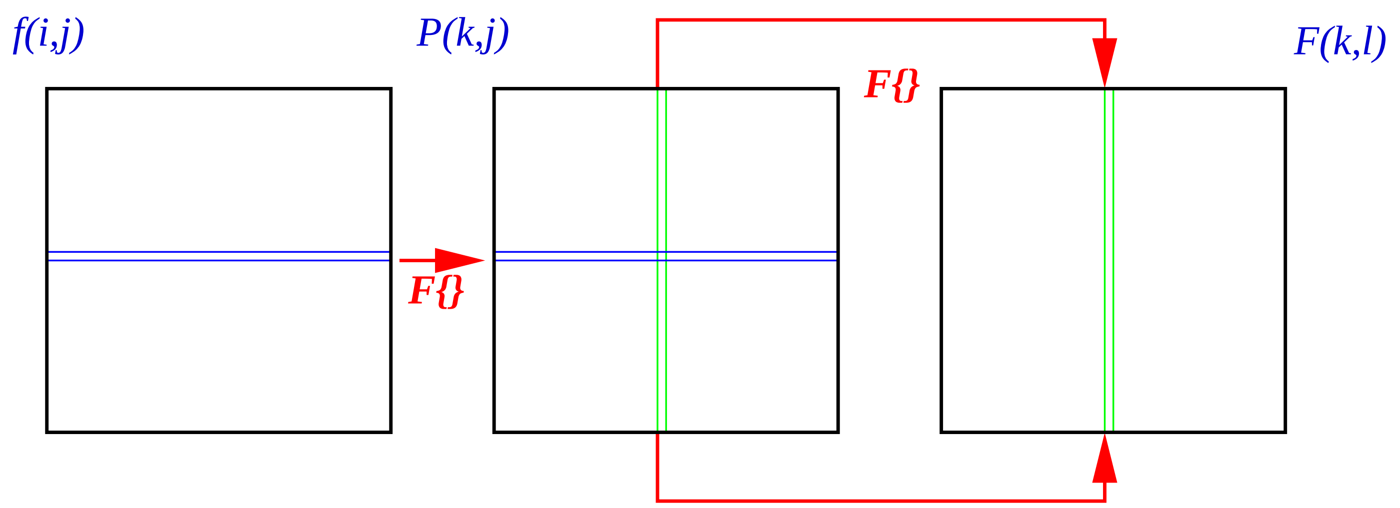
Calculation of Two Dimensional DFT

The exponential term is separable, then the Two Dimensional DFT can be implemented in **two** parts.

$$F(k,l) = \sum_{j=0}^{N-1} P(k,j) \exp\left(-\imath 2\pi \frac{lj}{N}\right)$$

where

$$P(k,j) = \sum_{i=0}^{N-1} f(i,j) \exp\left(-\imath 2\pi \frac{ki}{N}\right)$$



Calculation of Two Dimensional DFT

1. N 1-D DFTs "along-the-rows"
2. N 1-D DFTs "down-the-columns"

So we can implement the **Two Dimensional** DFT, by a series of $2N$ **One Dimensional** DFTs.

Extend the same argument to Three-Dimensions, where for $N \times N \times N$ which we can again break into one-dimensional DFTs

Calculation of DFT

The 2-D DFT can be calculated from $2N$ 1-D DFT of the type,

$$F(k) = \sum_{i=0}^{N-1} f(i) \exp\left[-\imath 2\pi \frac{ki}{N}\right]$$

Computational complexity proportional to N^2.

Typically calculated by **FFT** algorithm which has a computational complexity proportional to $N\log_2(N)$.

Restrictions on N:

- **Original algorithm:** $N = 2^n$ only. *Radix 2 DFT*
- **Typical algorithm:** $N = 2^{n+1}3^m5^p$ *Mixed Radix DFT*
- **FFTW**: gives $N\log_2(N)$ for any N using highly optimised code.

The 1-D FFT is a *non-trivial* algorithm, use a standard piece of code and *don't* try and write one.

Aside: Good Two-Dimensional codes do **not** use the single 1-D technique. Exploit symmetry of 2-D system able to make additional 10-15% computational saving.

Practical Considerations

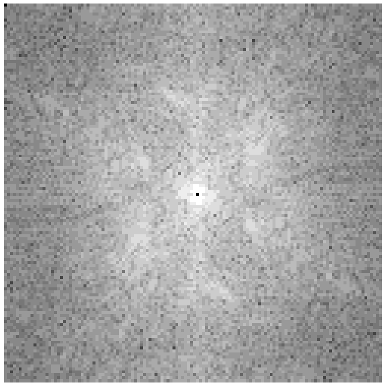
Dynamic Range:

- Most input images $f(i,j)$ in range $0 \rightarrow 255$ (8-bit)
- DFT is Complex, with range typically 10^{12}

Must use Floating Point arithmetic to calculate DFT. (Slows it down considerable).

Typically display:

$$\log\left(|F(k,l)|^2 + 1\right)$$



Practical Considerations

Location of Centre:

Most algorithms locate $F(0,0)$ at top/left.

Shift to centre by *Convolution* of DFT with

$$\delta\left(i - \frac{N}{2}, j - \frac{N}{2}\right)$$

Use *Convolution Theorem*, so “Multiply in real space”, with “checker pattern”

$$\begin{matrix} 1 & -1 & 1 & \dots & -1 \\ -1 & 1 & -1 & \dots & 1 \\ 1 & -1 & 1 & \dots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & 1 & -1 & \dots & 1 \end{matrix}$$

This is **much** faster than direct shift.

Sampling Theory

Before we can *sample* an image we *must* what,

$$\Delta x \quad \text{and} \quad \Delta y$$

should we use to retain the information in $f(x,y)$.

We find it depends on the **maximum** spatial frequency in the image.

Shannon Sampling Theory

If function $f(x)$ has Fourier Transform of width a, so that:

$$F(u) = 0 \quad \text{for } |u| > a/2$$

then $f(x)$ **completely** specified by samples at

$$\Delta x = \frac{1}{a}$$

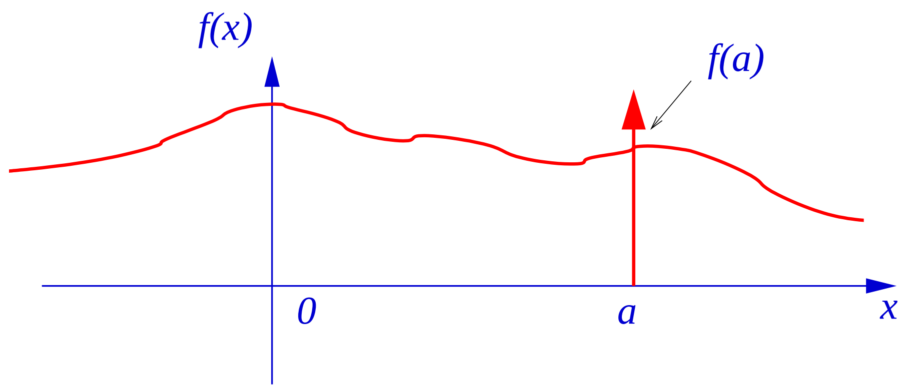
Note: No limit or range of $f(x)$, will in general be infinite.

We now need to understand what this means?

Sampling Property of δ-function

We have from properties of $\delta(x)$, that for function $f(x)$, then

$$\int_{-\infty}^{\infty} f(x)\,\delta(x-a)\,\mathrm{d}x = f(a)$$



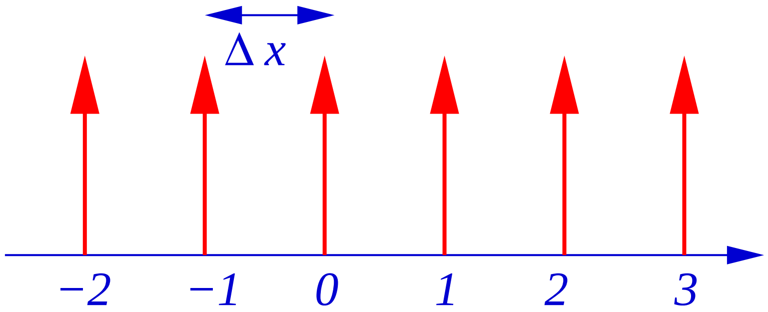
so that convolution with a shifted $\delta(x)$ at position $x = a$ measures or samples the function at $x = a$

Sampling Function

Define a *sampling function* as a

$$s(x) = \sum_{i=-\infty}^{\infty} \delta(x - i\Delta x)$$

Which is a *Comb* of δ-function,



so the sampled function is given by

$$f(x)\, s(x)$$

So in Fourier space we get a convolution of,

$$F(u) \odot S(u)$$

where $S(u)$ is the Fourier Transform of the *Comb*.

Sampling Function

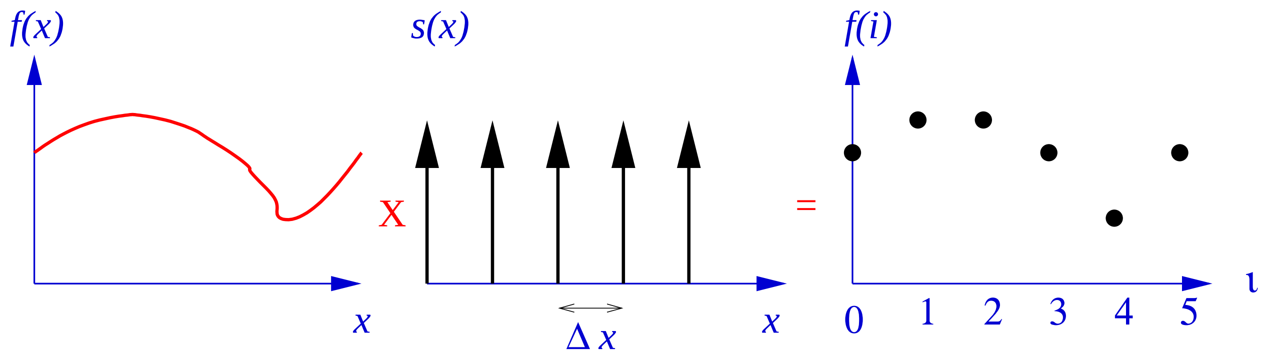
It-can-be-shown (see Tutorial 7 of Fourier Transform Booklet) that this is *comb* is reciprocal spacing, given by:

$$S(u) = \sum_{i=-\infty}^{\infty} \delta(u - \frac{i}{\Delta x})$$

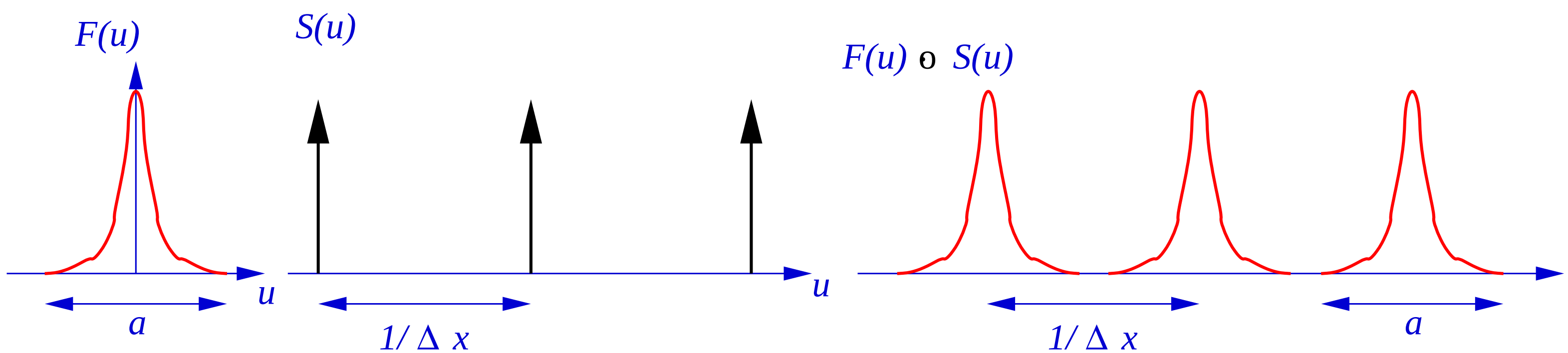
Now consider the affect of this on sampling.

One Dimensional Example

Take a One Dimensional function:



Then in Fourier space we have



which will be separated **ONLY** if

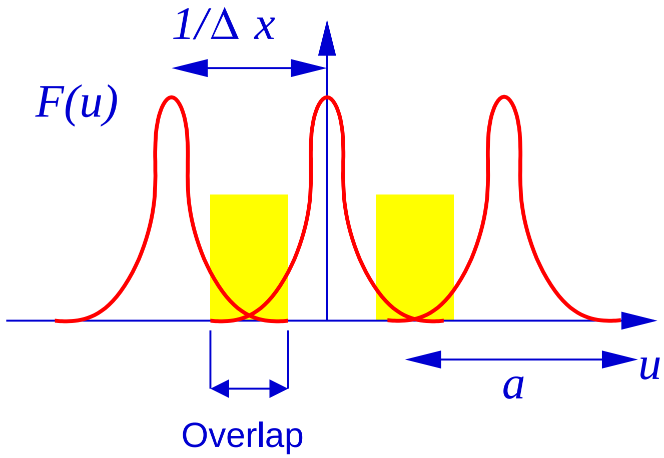
$$a \le \frac{1}{\Delta x}$$

Implications for Sampling

Three possible conditions:

Understampling: Sample less-often than required.

$$\Delta x > \frac{1}{a}$$

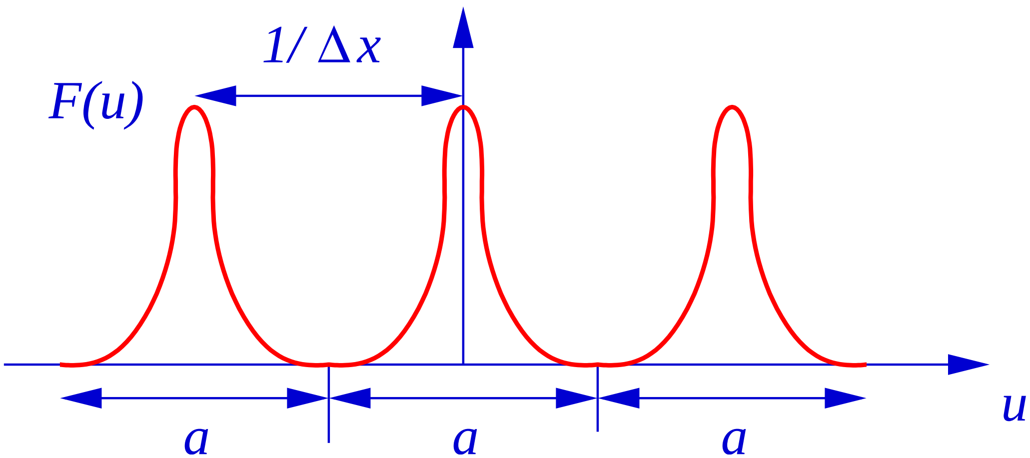


Overlap in Fourier Space. Information lost (Known as *aliasing*).

Implications for Sampling I

Shannon Sampling: Exactly what is required.

$$\Delta x = \frac{1}{a}$$

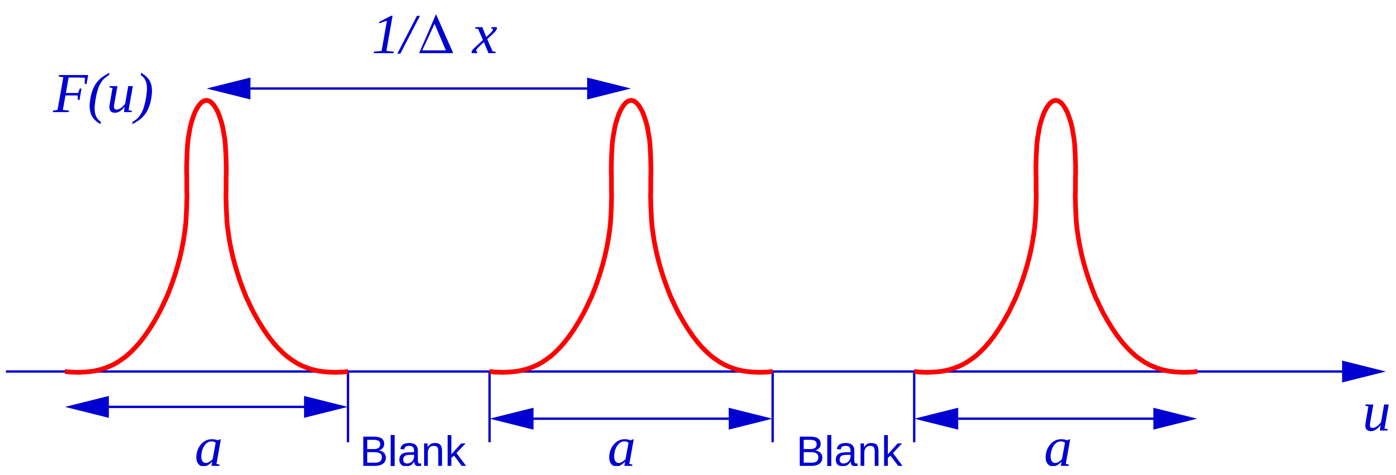


No overlap of orders in Fourier Space. Each replication is just Fourier Transform or unsampled input. Able to extract **all** information about original input.

Implications for Sampling I

Oversampling: More often than is required.

$$\Delta x < \frac{1}{a}$$



Orders on Fourier Space well separated. No added information about $f(x)$, but have more samples to deal with.

Aside: This assume that there is no noise on the signal. If noise some improvement possible with oversampling.

Details of this beyond this course.

Sampling in Two Dimensions

Here we have a two dimensional function $f(x,y)$

Define a 2-D sampling function,

$$s(x,y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \delta(x - i\Delta x\,, y - j\Delta y)$$

Grid of δ-functions with separation $\Delta x\,, \Delta y$.

In real space we have

$$f(i,j) = f(x,y)\, s(x,y)$$

so in Fourier plane we get:

$$F(u,v) \odot S(u,v)$$

where we have that

$$S(u,v) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \delta\left(u - \frac{i}{\Delta x}, v - \frac{j}{\Delta y}\right)$$

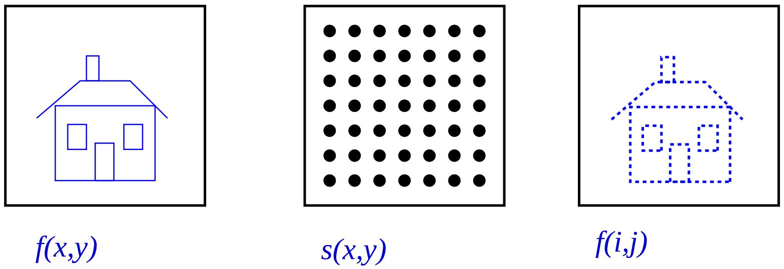
Then if $F(u,v)$ is rectangular of size $a \times b$ *Shannon Sampling Rate* is given by,

$$\Delta x = \frac{1}{a} \quad \Delta y = \frac{1}{b}$$

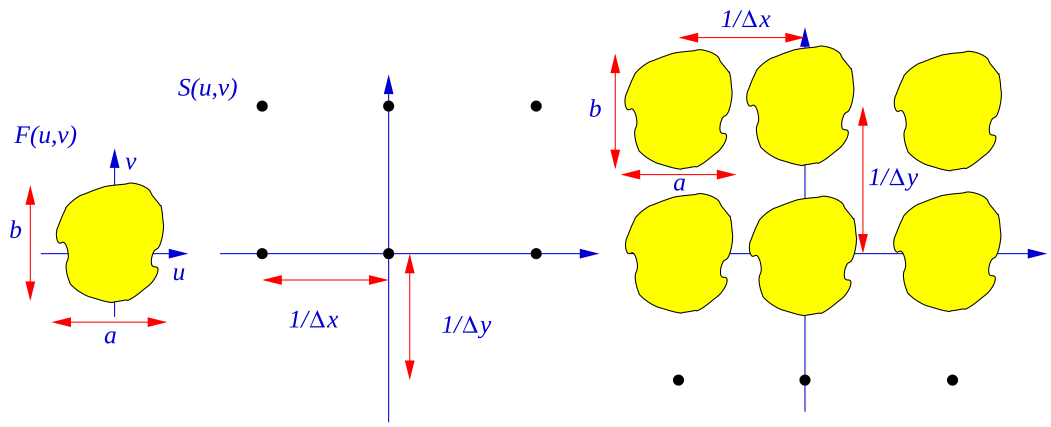
Typically take $\Delta x = \Delta y$.

Two Dimensional Example

In two-dimensions we have



Then in Fourier space we have



$$\Delta x < \frac{1}{a} \quad \Delta y < \frac{1}{b}$$

Functions of Finite Extent

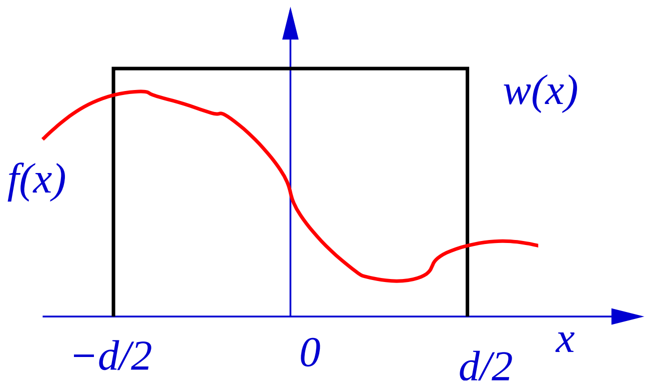
Take the more practical case a finite function $\tilde{f}(x)$ being

$$\tilde{f}(x) = f(x)\,w(x)$$

where

$$w(x) = \Pi\left(\frac{x}{2d}\right)$$

so that $\tilde{f}(x)$ is of width d.



Multiplication in Real space, so in Fourier space,

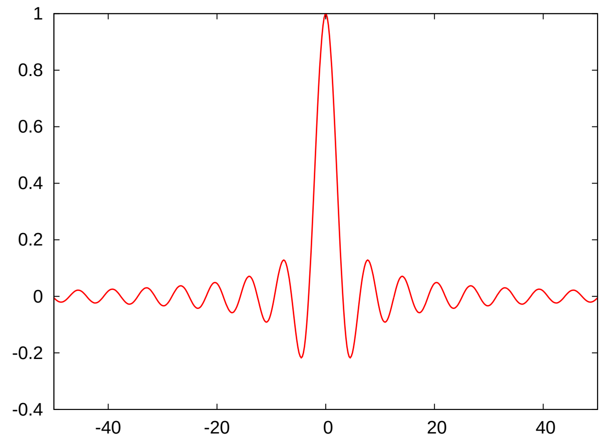
$$\tilde{F}(u) = F(u) \odot W(u)$$

where

$$W(u) = \mathrm{sinc}(\pi d u)$$

Functions of Finite Extent I

But $\mathrm{sinc}()$ is **infinite** in extent.



Looks like a massive problem:

$$W(u) \quad \text{infinite in extend} \quad \Longrightarrow \quad \tilde{F}(u,v) \quad \text{infinite in extend}$$

so *width* in Fourier space $a \longrightarrow \infty$ so that

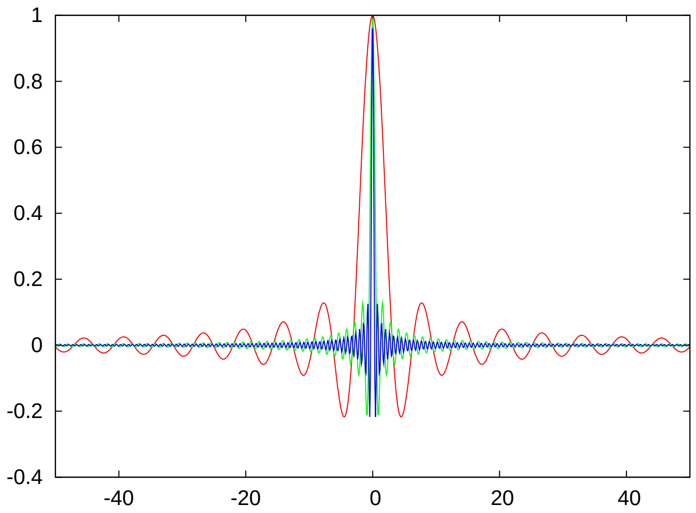
$$\Delta x = \frac{1}{a} \longrightarrow 0$$

so sample point become *infinitely* close together!

Functions of Finite Extent II

Often possible to assume that finite functions obey *Shannon Sampling* (True if d is large).

$$W(u) = \mathrm{sinc}(\pi d u)$$



is Sharply peaked with first zeros at $\pm 1/d$.

Functions of Finite Extent III

Note that d is the length of the signal sampled, so if the sampling rate is Δx then

$$d = N\,\Delta x$$

so if N large, (take a lot of samples), then we can assume Shannon Sampling, where

$$\Delta x = \frac{1}{a}$$

where a is the width of the Fourier Transform of the *unwindowed* function $f(x)$.

Sampling Rate in Real and Fourier Space

Real Space:

Function $f(x)$ has bandwidth a, so that

$$F(u) = 0 \quad \text{for } |u| > a/2$$

then Shannon Sampling rate as:

$$\Delta x = \frac{1}{a}$$

If in real space we take N samples, then window length is length $N\Delta x$, so that

$$d = N\,\Delta x$$

Fourier Space:

Sample function $\tilde{F}(u)$ in Fourier Space. In real space we have function of width d

$$\tilde{f}(x) = f(x)\,w(x) = 0 \quad \text{for } |x| > d/2$$

so we define a *Shannon Sampling* rate in Fourier space is

$$\Delta u = \frac{1}{d} = \frac{1}{N\Delta x}$$

So for a function sampled at a rate Δx is real space, the equivalent sampling is Fourier space is $\Delta u = \frac{1}{N\Delta x}$

Sampling Rate in Two Dimensions

Identical in Two-Dimensions, with

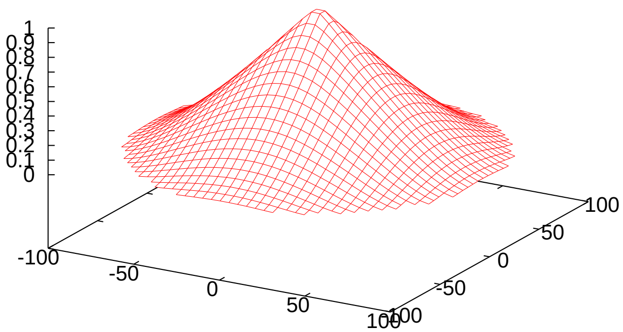
$$\Delta u = \frac{1}{N\Delta x} \quad \& \quad \Delta v = \frac{1}{N\Delta y}$$

Note we assume that the sampled image is Square, if not different sampling frequency in each direction.

Example Ideal Imaging System

For ideal optical system we have FT limited by $H(w)$ (OTF), where

$$H(u,v) = \frac{2}{\pi}\left[\cos^{-1}\left(\frac{w}{v_0}\right) - \frac{w}{v_0}\left(1-\left(\frac{w}{v_0}\right)^2\right)^{\frac{1}{2}}\right]$$



where

$$v_0 = \frac{d}{\lambda f} = \frac{1}{\lambda \mathrm{F_{No}}}$$

so that

$$H(w) = 0 \quad \text{for } |w| > v_0$$

with the FT contained in a square of $2v_0 \times 2v_0$.

Example Ideal Imaging System I

Sampling Interval given by,

$$\Delta x = \Delta y = \frac{1}{2\nu_0} = \frac{\lambda F_{\mathrm{No}}}{2}$$

So for

- $\lambda = 0.5\mu$m (Green Light)
- $F_{\mathrm{No}} = 8$ (Medium aperture)
- Then $\Delta x = 2\mu$m.

So for 35mm negative (36 by 24mm) need 18000 by 12000 samples.

In practice film grain reduces sampling requirement to about 5μm.

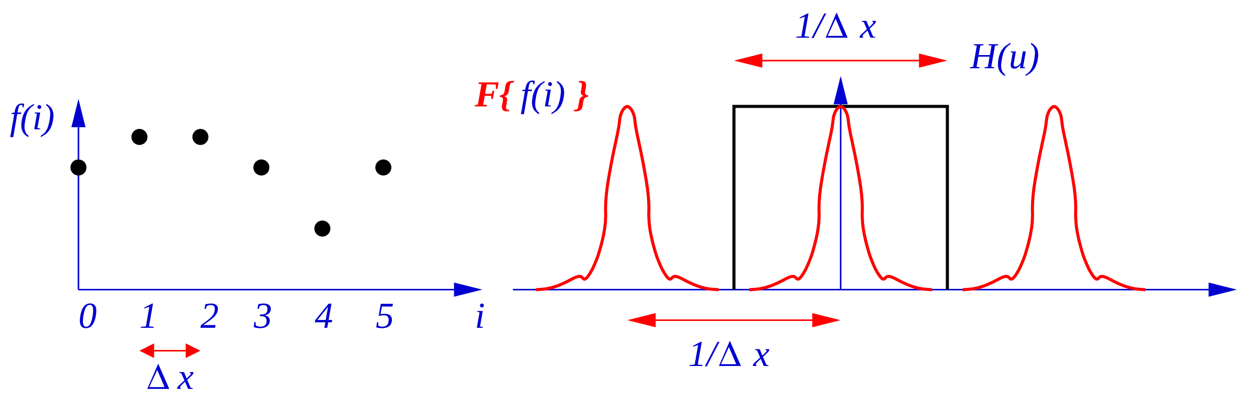
Vast amount of data. Do not usually need to consider all the data.

Reconstruction from Sampled Data

Inverse problem or recover of $f(x)$ from sampled version $f(i)$ samples at Δx.

The sampled signal $f(i)$ is given by

$$f(i) = f(x)\,s(x) = \mathcal{F}^{-1}\{F(u)\odot S(u)\}$$



In Fourier space $F(u)\odot S(u)$ is periodic with period $1/\Delta x$. We can isolate a single period by a filter

$$H(u) = \Pi\left(\frac{u}{\Delta x}\right)$$

Reconstruction from Sampled Data I

So that

$$(F(u) \odot S(u))\, H(u) = F(u)$$

which in real space we have that

$$f(x) = h(x) \odot (f(x)s(x)) = h(x) \odot f(i)$$

where

$$h(x) = \frac{1}{\Delta x} \mathrm{sinc}\left(\frac{\pi x}{\Delta x}\right)$$

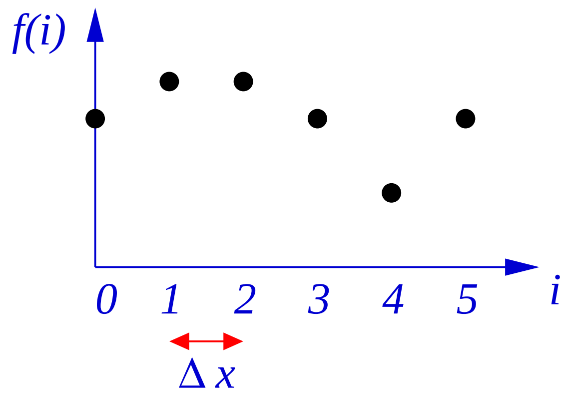
which is known as the interpolation function.

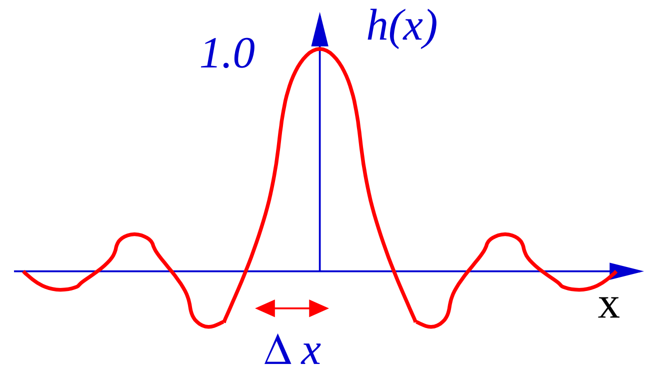
Typically we normalise to get,

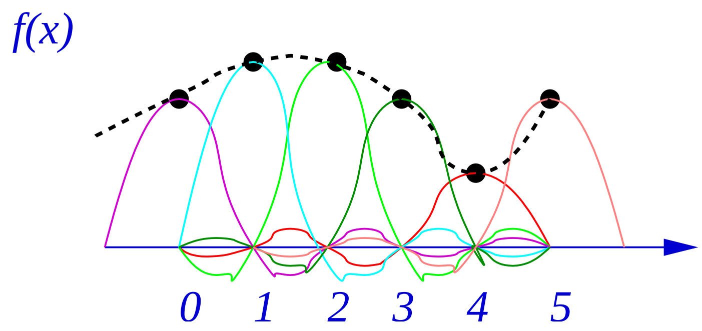
$$h(x) = \mathrm{sinc}\left(\frac{\pi x}{\Delta x}\right)$$

known as *ideal interpolation function*.

Reconstruction from Sampled Data I







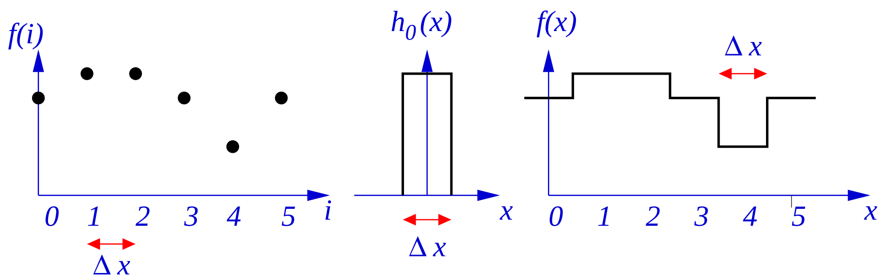
Note: that when $x = n\Delta x$ then $f(x)$ is value of sample at that point.

Problem: if $x \neq n\Delta x$ requires the $\mathrm{sinc}()$ to be convolved with all other points. Not computationally practical so approximations must be made.

Zeroth Order Interpolation

Simplest interpolation *nearest neighbour rule*,

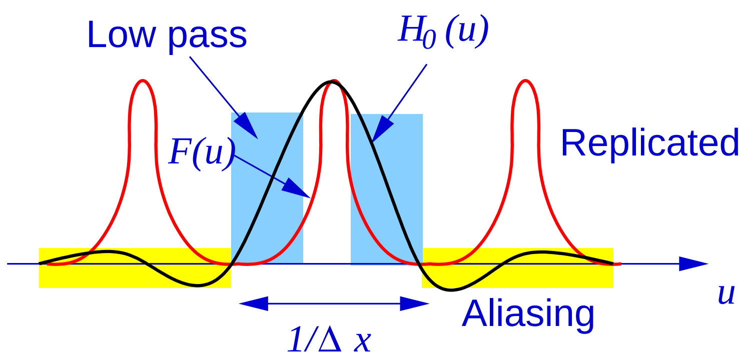
$$h(x) = \Pi\left(\frac{x}{\Delta x}\right)$$



Typical Stair-Case effect. Convolution in real space, so Multiplication in Fourier space by

$$H(u) = \mathrm{sinc}(\pi \Delta x u)$$

which attenuates high frequencies and aliasing effects from periodic orders



Zero Order in Two Dimensions

We get two dimensional interpolation function,

$$h(x,y) = \Pi\left(\frac{x}{\Delta x}\right)\,\Pi\left(\frac{y}{\Delta y}\right)$$

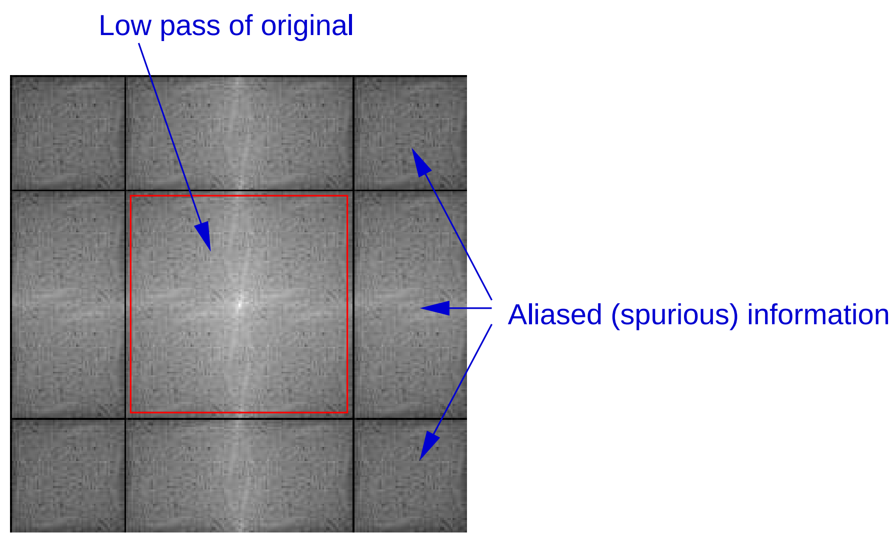
which we can interperate at closest values to (x,y), or

$$f(x,y) = f(i,j) \quad \text{for } |x - i\Delta x| \text{ and } |y - j\Delta y| \text{ minimised}$$

Typical Block structure and replications in Fourier Space.

Original

Zero−order expanded

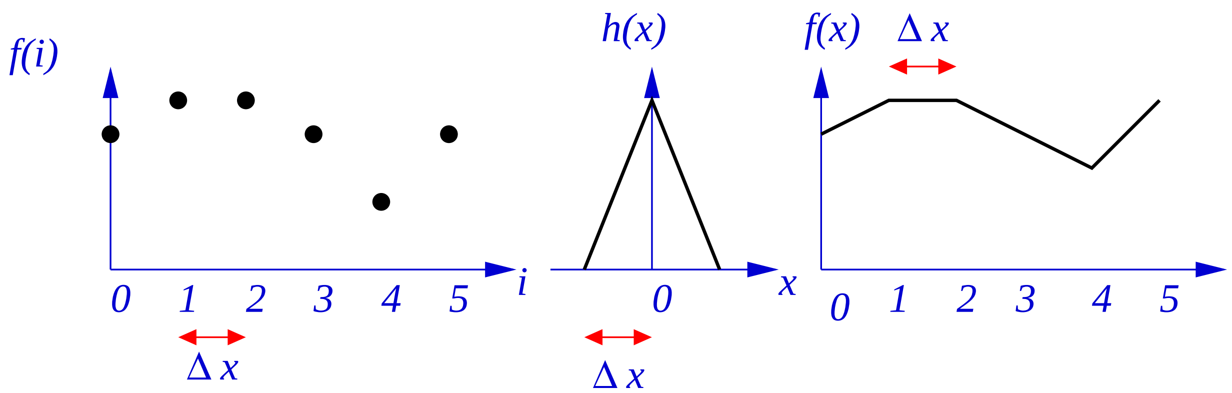


Fourier Transform

First Order Interpolation

Here we have that

$$h(x) = 1 - \frac{|x|}{\Delta x} \quad \text{for } |x| \leq \Delta x$$
$$= 0 \quad \text{else}$$

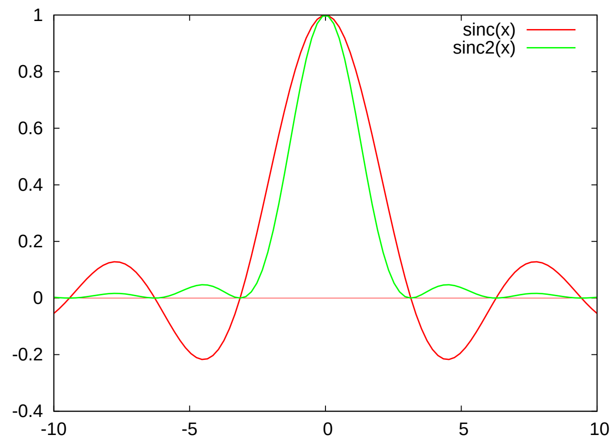


which in Fourier space gives,

$$H(u) = \text{sinc}^2(\pi \Delta x u)$$

First Order Interpolation I

Less high frequency attenuation and aliasing effects that zero order



Linear interpolation between the two points adjacent to the required position x.

$$i = \text{int}\left(\frac{x}{\Delta x}\right) \quad \text{and} \quad \alpha = \frac{x - i\Delta x}{\Delta x}$$

Then we have that

$$f(x) = (1-\alpha) f(i) + \alpha f(i+1)$$

Need to consider two sample points for each value of x.

First Order in Two Dimensions

In two dimensions the interpolation function becomes,

$$h(x,y) = \left(1 - \frac{|x|}{\Delta x}\right)\left(1 - \frac{|y|}{\Delta y}\right)$$

which can be implemented as the weighted average of **four** adjacent points. If we take :

$$i = \mathrm{int}\left(\frac{x}{\Delta x}\right) \quad \& \quad j = \mathrm{int}\left(\frac{y}{\Delta y}\right)$$

and

$$\alpha = \frac{x - i\Delta x}{\Delta x} \quad \& \quad \beta = \frac{y - j\Delta y}{\Delta y}$$

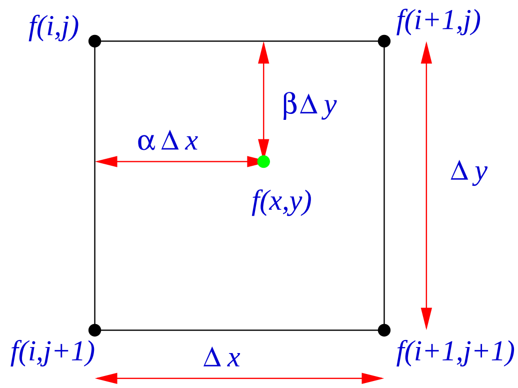
then we have that

$$\begin{aligned} f(x,y) \quad = \quad & (1-\alpha)(1-\beta) f(i,j) + \alpha(1-\beta) f(i+1,j) + \\ & (1-\alpha)\beta f(i,j+1) + \alpha\beta f(i+1,j+1) \end{aligned}$$

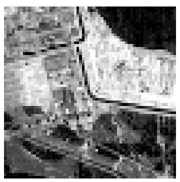
so we have to access **four** points for each x,y value.

First Order in Two Dimensions

In diagram form we have

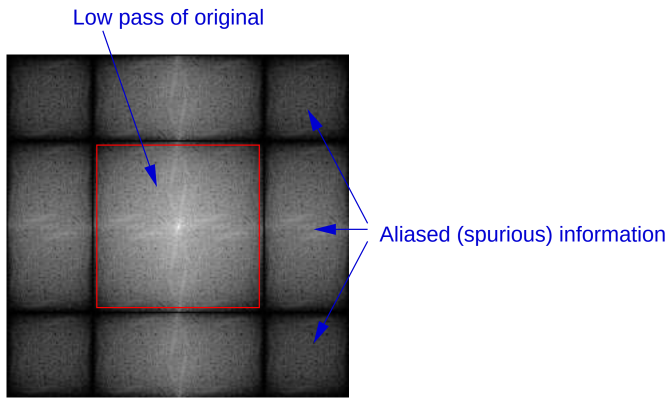


With real examples being



Original

First−order expanded



Fourier Transform

Other Interpolation Techniques

Range of other functions $h(x)$ based on Gaussians and limited range $\mathrm{sinc}()$s. In general the larger the window over which the interpolation is formed be better the reconstruction.

Most common *higher order* scheme is **Bicubic**, weighted average over 4×4 region. Used extensively in digital photography.

Summary

In this section we cane considered

1. Digital representation of images in real and Fourier space.
2. The discrete Fourier transform and its properties in one and two dimensions.
3. Calculation of the discrete Fourier transform by the FFT
4. Sampling theory in one and two dimensions from a Fourier viewpoint.
5. Limitations of the sampling theorem and its practical application.
6. Reconstruction from sampled and the ideal interpolation function.
7. Zero and First order interpolation and their effects in real and Fourier space.
8. Outline of higher order interpolation schemes.