EECS C145B / BioE C165: Image Processing and Reconstruction Tomography

Lecture 5

Jonathan S. Maltz

jon@eecs.berkeley.edu http://muti.lbl.gov/145b
510-486-6744

Topics to be covered

1. Definition of several important 2D functions
2. Important 2D Fourier transform pairs
3. Some 2D Fourier transform properties and theorems
4. Separability of the n-D Fourier transform
5. Sampling in 2D
6. Aliasing in 2D
7. 2D low-pass filters in Fourier domain
8. 2D high-pass filters in Fourier domain
9. Reconstruction filters

Reading

- Gonzalez and Woods pp. 147-191, pp. 208-213.

Optional advanced reading

- Jain pp. 132-150, 244-251.
- Bracewell pp. 346-364.

Definition of important 2D functions

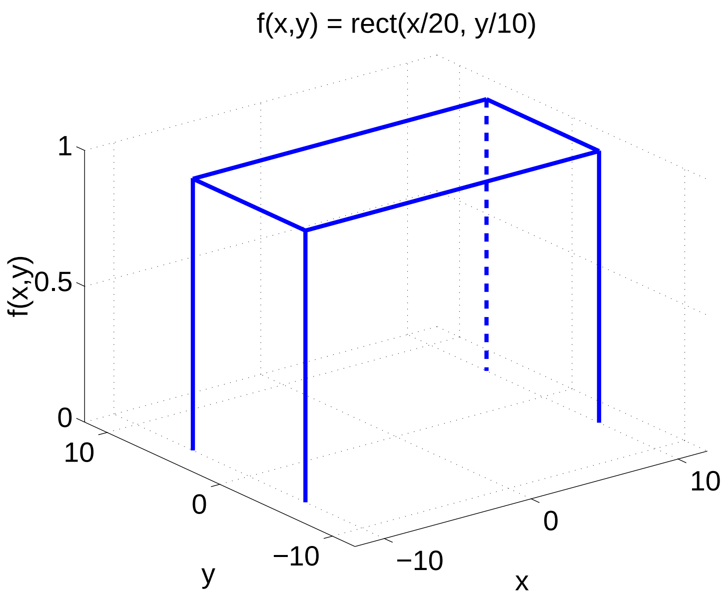
Recall for 1D:

$$\mathrm{rect}\left(\frac{x}{X}\right) = \begin{cases} 1, & |x| < X/2 \\ 0, & |x| > X/2 \end{cases}$$

In 2D we have:

$$\mathrm{rect}\left(\frac{x}{X}, \frac{y}{Y}\right) \triangleq \mathrm{rect}\left(\frac{x}{X}\right)\mathrm{rect}\left(\frac{y}{Y}\right)$$

Example rect function



Definition of important 2D functions: Rectangular function

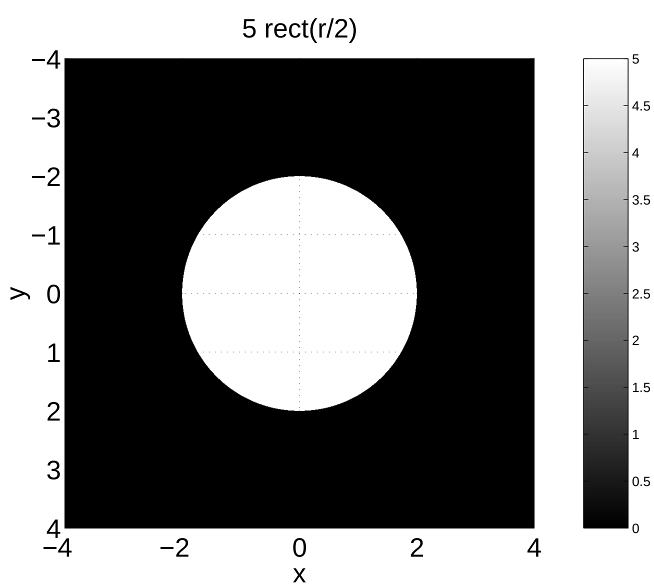
A radial rectangular function in 2D is defined as:

$$\mathrm{rect}\left(\frac{r}{R}\right) \triangleq \begin{cases} 1, & |r| < R \\ 0, & |r| > R \end{cases}$$

where:

$$r = \sqrt{x^2 + y^2}$$

Example radial rect function with $r = \sqrt{x^2 + y^2}$



Definition of important 2D functions: Sinc function

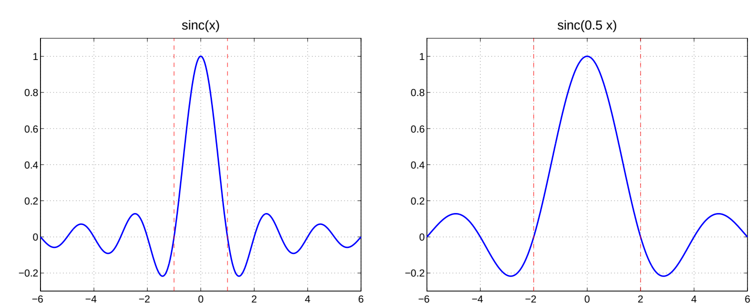
Recall in 1D:

$$\mathrm{sinc}(x) \triangleq \frac{\sin(\pi x)}{\pi x}$$

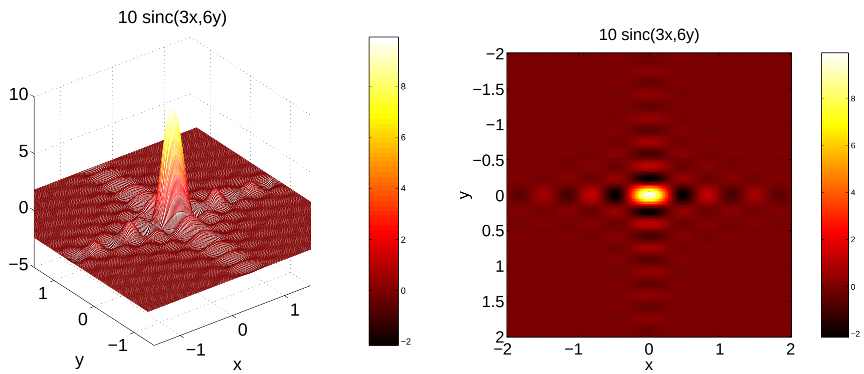
A special property of the 1D sinc function is that it contains all frequencies equally up to a cutoff.

In 2D:

$$\mathrm{sinc}(x, y) \triangleq \mathrm{sinc}(x)\,\mathrm{sinc}(y) = \frac{\sin(\pi x)}{\pi x}\frac{\sin(\pi y)}{\pi y}$$



Example 2D sinc function



Definition of important 2D functions: Jinc function

The jinc function is defined as:

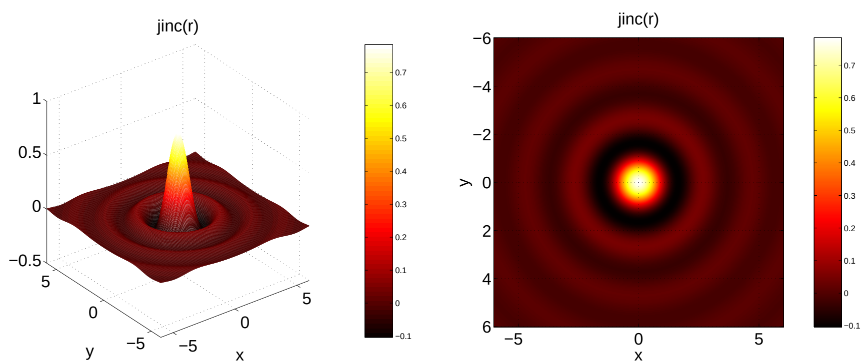
$$\mathrm{jinc}(r) \triangleq \frac{J_1(\pi x)}{2x}$$

where J_1 is a Bessel function of the first kind. A Bessel function of the nth kind is a solution of the differential equation:

$$x^2 \frac{d^2y}{dx^2} + x\frac{dy}{dx} + (x^2 - n^2)y = 0$$

- A special property of the 2D jinc function is that it contains all 2D frequencies equally up to a cutoff.
- As we will see later, the jinc function is the Fourier transform of the radial rect function.

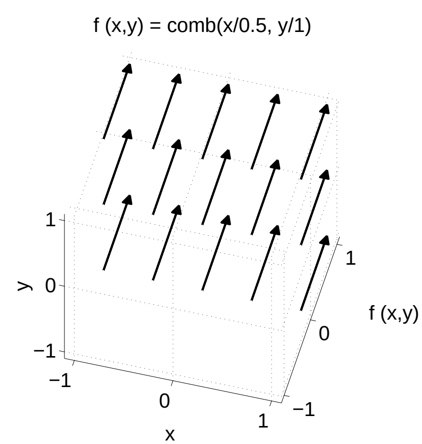
Example jinc function with $r = \sqrt{x^2 + y^2}$



Comb of impulses

A comb of impulses is a periodic train of impulses:

$$\mathrm{comb}(x/X, y/Y) \triangleq \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(x - kX,\, y - lY)$$



Useful Fourier transform pairs

Space domain	Frequency domain
$\delta(x,y)$	1
$\delta(x-x_0, y-y_0)$	$\mathrm{e}^{\pm \jmath 2\pi x_0 u}\mathrm{e}^{\pm \jmath 2\pi y_0 v}$
$\mathrm{rect}\left(\frac{x}{A}, \frac{y}{B}\right)$	$AB\,\mathrm{sinc}(Au, Bv) = AB\,\frac{\sin(\pi u A)}{\pi u A}\frac{\sin(\pi v B)}{\pi v B}$
$\mathrm{rect}(r/R),\ r = \sqrt{x^2+y^2}$	$R\ \mathrm{jinc}(R\rho),\ \rho = \sqrt{u^2+v^2}$
$\mathrm{comb}(x/X, y/Y)$	$\mathrm{XY}\,\mathrm{comb}(uX, vY)$
$\cos(2\pi(u_o x + v_0 y))$	$\frac{1}{2}(\delta(u+u_0, v+v_0) + \delta(u-u_0, v-v_0))$
$\sin(2\pi(u_o x + v_0 y))$	$\jmath\frac{1}{2}(\delta(u+u_0, v+v_0) - \delta(u-u_0, v-v_0))$
$\mathrm{e}^{-\pi(x^2+y^2)}$	$\mathrm{e}^{-\pi(u^2+v^2)}$

Dual Fourier transform pairs

Space domain	Frequency domain
1	$\delta(u,v)$
$\mathrm{e}^{\mp \jmath 2\pi u_o x}\mathrm{e}^{\mp \jmath 2\pi v_0 y}$	$\delta(u-u_0, v-v_0)$
$AB\,\mathrm{sinc}(Ax, By)$	$\mathrm{rect}\left(\frac{u}{A}, \frac{v}{B}\right)$
$R\,\mathrm{jinc}(rR), r = \sqrt{x^2+y^2}$	$\mathrm{rect}(\rho/R),\ \rho = \sqrt{u^2+v^2}$
$UV\ \mathrm{comb}(Ux, Vy)$	$\mathrm{comb}(u/U, v/V)$

Can we derive expressions for Fourier transform pairs for the DFT? Explain.

Some important Fourier transform properties

Spectrum	$\lvert F(u,v)\rvert = \sqrt{R(u,v)^2 + I(u,v)^2}$
(magnitude)	$R(u,v)$: real part of $F(u,v)$
	$I(u,v)$: imaginary part of $F(u,v)$
Phase	$\phi(u,v) = \arctan\left(\frac{I(u,v)}{R(u,v)}\right)$
Conjugate symmetry	$F(u,v) = F^*(-u,-v)$
	$\lvert F(u,v)\rvert = \lvert F(-u,-v)\rvert$
Duality	If $g(x,y) \rightleftharpoons f(u,v)$
	then $f(x,y) \rightleftharpoons g(-u,-v)$
Translation	$f(x \pm x_0, y \pm y_0) \rightleftharpoons F(u,v)\mathrm{e}^{\pm \jmath 2\pi(ux_0+vy_0)}$
	$f(x,y)\mathrm{e}^{\mp \jmath 2\pi(u_0 x + v_0 y)} \rightleftharpoons F(u \pm u_0, v \pm v_0)$
Differentiation	$\frac{\partial}{\partial x} f(x,y) \rightleftharpoons \jmath\, 2\pi u F(u,v)$

Translation property in space

Notice that a translation of $f(x,y)$ changes only the phase of $F(u,v)$:

$$f(x \pm x_0, y \pm y_0) \rightleftharpoons F(u,v)\mathrm{e}^{\pm \jmath 2\pi(ux_0+vy_0)}$$

The magnitude spectrum is **invariant under translations of the image.**

$$\left| F(u,v)\,\mathrm{e}^{\pm \jmath 2\pi(ux_0+vy_0)} \right|$$

$$= \sqrt{F(u,v)\mathrm{e}^{\pm \jmath 2\pi(ux_0+vy_0)} \times F^*(u,v)\mathrm{e}^{\mp \jmath 2\pi(ux_0+vy_0)}} = \lvert F(u,v)\rvert$$

Can a linear shift invariant system move an image $f(x,y)$ around real space? ________.
Can it move its transform $F(u,v)$ around frequency space?

Translation property in frequency

A translation in frequency is equivalent to a modulation of the space signal by a complex exponential:

$$f(x,y)\,\mathrm{e}^{\mp \jmath 2\pi(u_0 x + v_0 y)} \rightleftharpoons F(u \pm u_0, v \pm v_0)$$

We can get some intuition for this by expanding the complex exponential:

$$\begin{aligned} & f(x,y)\,\mathrm{e}^{\mp \jmath 2\pi(u_0 x + v_0 y)} \\ & \quad = f(x,y)\Big[\cos(2\pi(u_0 x + v_0 y) \mp \jmath \sin(2\pi(u_0 x + v_0 y)\Big] \end{aligned}$$

Now we examine what happens to a single sinusoid within $f(x,y)$ when we perform a translation of its FT. For simplicity we choose the cosine function:

$$g(x,y) = \cos(2\pi(2x + 2y))$$

Translation property in frequency

This function has the FT:

$$G(u,v) = \frac{1}{2}\big[\delta(u+2, v+2) + \delta(u-2, v-2)\big]$$

We decide to translate $G(u,v)$ by $u_0 = 1$ and $v_0 = -3$:

$$G'(u,v) = G(u-1, v+2) = \frac{1}{2}\big[\delta(u+1, v-1) + \delta(u-3, v-5)\big].$$

Now we see what happens at the other end of the transform pair:

$$\begin{aligned} g'(x,y) &= \cos(2\pi(2x+2y))\,\mathrm{e}^{\jmath 2\pi(-1x + (-3)y)} \\ &= \frac{1}{2}\Big[\mathrm{e}^{\jmath 2\pi(2x+2y)} + \mathrm{e}^{-\jmath 2\pi(2x+2y)}\Big]\,\mathrm{e}^{\jmath 2\pi(-1x-3y)} \\ &= \frac{1}{2}\Big[\mathrm{e}^{\jmath 2\pi(x-y)} + \mathrm{e}^{-\jmath 2\pi(3x+5y)}\Big] \end{aligned}$$

Translation property in frequency

As expected, the frequency of the cosine is changed by the translation in the Fourier domain. The function $g'(x,y)$ is a complex function, since its FT $G'(u,v)$ is not conjugate symmetric.

An expression for $g'(x,y)$ can also be obtained using the well-known multiplication formulae:

$$\begin{aligned} \sin(\alpha)\cos(\beta) &= \frac{1}{2}\Big[\sin(\alpha-\beta) + \sin(\alpha+\beta)\Big] \\ \sin(\alpha)\sin(\beta) &= \frac{1}{2}\Big[\cos(\alpha-\beta) - \cos(\alpha+\beta)\Big] \\ \cos(\alpha)\cos(\beta) &= \frac{1}{2}\Big[\cos(\alpha-\beta) + \cos(\alpha+\beta)\Big] \end{aligned}$$

and Euler's identity:

$$\mathrm{e}^{\pm \jmath\theta} = \cos(\theta) \pm \jmath \sin(\theta)$$

Rotation property

$$\begin{aligned} x &= r\cos(\theta) & \qquad u &= \rho\cos(\phi) \\ y &= r\sin(\theta) & \qquad v &= \rho\sin(\phi) \end{aligned}$$

$$f(r, \theta + \theta_0) \leftrightharpoons F(\rho, \phi + \theta_0)$$

This tells us that a rotation of a 2D function by θ_0 will rotate the Fourier transform by **the same angle and in the same direction** (i.e., clockwise or counterclockwise).

Convolution theorem

$$f(x,y) * h(x,y) \rightleftharpoons F(u,v)H(u,v)$$
$$f(x,y)h(x,y) \rightleftharpoons F(u,v) * H(u,v)$$

- The convolution theorem tells us that the output of a linear shift invariant system can be computed via multiplication of the transforms of the point-spread function and the input image. Taking the inverse transform of the result gives the output of the system.
- The dual transform pair (sometimes called the multiplication property) tells us that if we multiply two images, their Fourier transforms are convolved with each other.

Separability property

This extremely important property allows us to calculate Fourier transforms of **any dimension** using the one dimensional transform.

$$\begin{aligned}
\mathcal{F}_2\Big\{f(x,y)\Big\} &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\,\mathrm{e}^{-\jmath\,2\pi(ux+vy)}dx\,dy \\
&= \int_{-\infty}^{\infty} \mathrm{e}^{-\jmath\,2\pi ux} \int_{-\infty}^{\infty} f(x,y)\,\mathrm{e}^{-\jmath\,2\pi vy}\,dy\,dx \\
&= \int_{-\infty}^{\infty} \mathrm{e}^{-\jmath\,2\pi ux} F(x,v)\,dx \\
&= F(u,v)
\end{aligned}$$

Separability property for DFT

The separability property applies also to the DFT. Consider the M column $\times$ N row image $f[m,n]$:

$$\begin{aligned}
\mathrm{DFT}_2\{f[m,n]\} &= \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]\,\mathrm{e}^{-\jmath 2\pi(km/M+ln/N)} \\
&= \sum_{m=0}^{M-1} \mathrm{e}^{-\jmath 2\pi km/M} \sum_{n=0}^{N-1} f[m,n]\,\mathrm{e}^{-\jmath 2\pi ln/N} \\
&= \sum_{m=0}^{M-1} \mathrm{e}^{-\jmath 2\pi km/M} F[m,l] \\
&= F[k,l] \\
&= \mathrm{DFT}_1^{\text{columns}}\Big\{\mathrm{DFT}_1^{\text{rows}}\Big\{f[m,n]\Big\}\Big\}
\end{aligned}$$

It is easy to show that the separability property extends to any dimension.

Recipe for 2D DFT

1. Transform the M rows using an N point 1D DFT engine.
2. Transform the N columns of the result using an M point 1D DFT engine.

Recipe for 3D DFT

Consider an image of dimension $n_1 \times n_2 \times n_3$.

1. Transform the $n_1 \times n_2$ vectors parallel to the n_3 axis using an n_3 point transform.
2. Transform the $n_2 \times n_3$ vectors parallel to the n_1 axis of the result of (1) using an n_1 point transform.
3. Transform the $n_1 \times n_3$ vectors parallel to the n_2 axis of the result of (2) using an n_2 point transform.

Extension to higher dimensions is straightforward.

Sampling in 2D

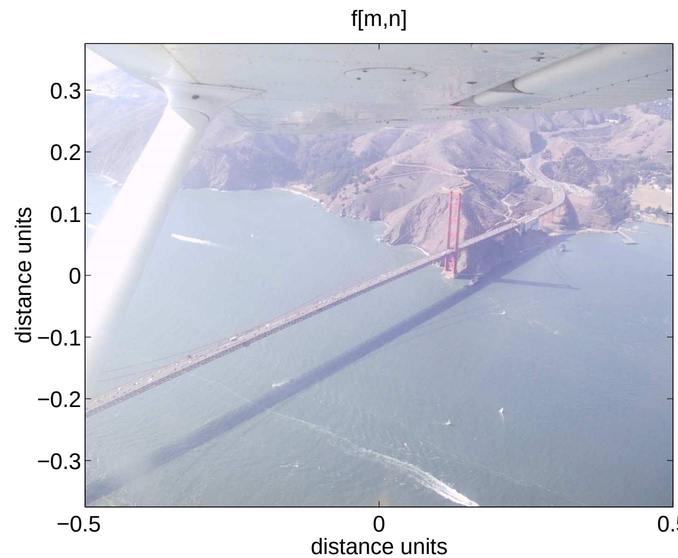
- Imagine we are in a plane over the Golden Gate bridge and we take a photo.
- Our digital camera is set to sample the scene at $N \times M = 1280 \times 960$ pixels.
- Let us denote the scene as $f(x, y)$ and define the width of the CCD to be 1 unit. The height is then 0.75 units.
- We assume that the pixels on the CCD are square.
- The sampling frequency in the x direction is:

$$u_s = 1280 \text{ samples/1 distance unit}$$

In the y direction it is:

$$\begin{aligned} v_s &= 960 \text{ samples/0.75 distance units} \\ &= 1280 \text{ samples per distance unit} = u_s \end{aligned}$$

2D sampling



2D sampling

- The sampling theorem tells us that the highest frequency that will be present in our digital image will be: ________
- The sampling period is $1/u_s = 1/v_s = 1/1280$ distance units.
- Our sampled image is given by:

$$f_s(x, y) = f(x, y) \times \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(x - k/1280, y - l/1280)$$

- The image $f_s(x, y)$ on the CCD is captured by an analog to digital converter, and becomes a 2D array of numbers $f[m, n]$.
- The scene $f(x, y)$ has a Fourier transform $F(u, v)$. What is the highest frequency present in $F(u, v)$?

____

- What is the FT of $f_s(x, y)$?

Sampling in 2D

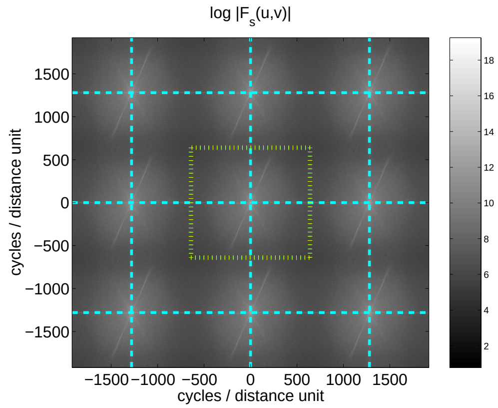
- Taking the FT of $f_s(x, y)$ and using the multiplication property of the FT we get:

$$F_s(u, v) = F(u, v) * \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(u - 1280k,\ y - 1280\,l)$$

So, $F_s(u, v)$ is just $F(u, v)$ rubber-stamped all over the u-v plane at intervals of 1280 cycles/distance unit, the sampling frequency.

2D sampling

The 9 periods of the log magnitude spectrum of $F_s(u,v)$ nearest the origin are:



The "rubber-stamp" is enclosed in a dotted line. The dashed lines mark the multiples of the sampling frequency that occur within the area plotted. The center of the stamp has been applied at the intersections of these lines.

2D sampling

- Now what happens if we take the DFT of the 2D array $f[m,n]$?
- Taking the DFT is equivalent to sampling the spectrum $F_s(u,v)$ at a spacing of $u_s/N = 1280/1280 = 1$ cycle/distance unit in the u-direction and $v_s/M = 1280/960 = 4/3$ cycles/distance unit in the v-direction.
- This can be expressed as:

$$F[k,l] = F_s(u,v) \times \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(u - ku_s/N, v - lv_s/M)$$

- Taking the inverse FT gives:

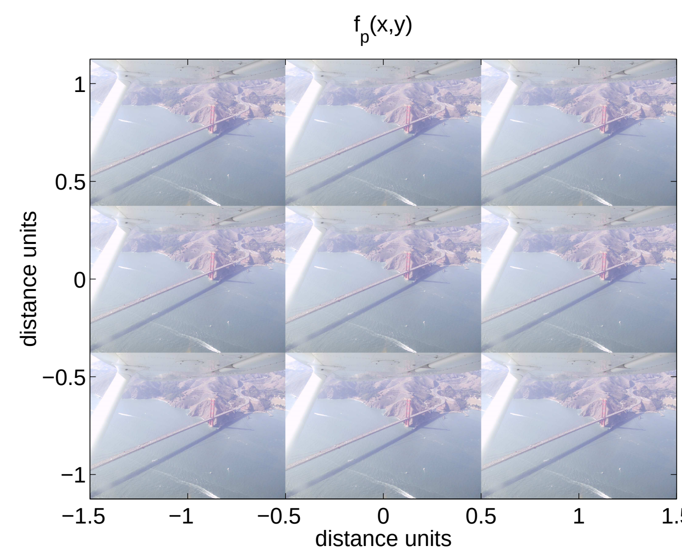
$$f_p(x,y) = f_s(x,y) * \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - nN/u_s, y - mM/v_s)$$

Sampling in 2D

- Thus, the function that the DFT will really give us the spectrum of **is not** $f_s(x,y)$ **or** $f(x,y)$, but $f_p(x,y)$, which is **periodic.**
- It repeats every $N/u_s = 1$ distance units along the x-axis and every $M/v_s = 960/1280 = 0.75$ distance units along the y-axis.

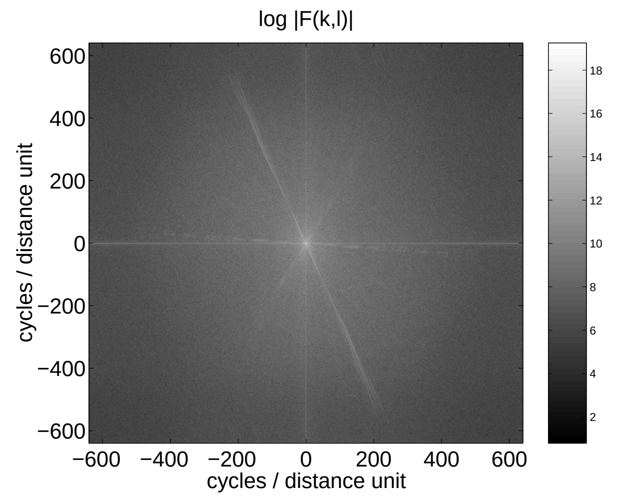
2D sampling

The 9 periods of $f_p(x,y)$ nearest to the origin are:



2D sampling

- When we apply the 2D DFT to the array of numbers $f[m, n]$, we get an array of the same size $F[k, l]$. We must always remember that this is actually **the first period of the transform of** $f_p(x, y)$.



Axes of the 2D DFT

- We always need to put meaningful axes of images of the DFT.
- The frequencies corresponding to each sample along the k axis begin at $-u_s/2 = -640$ and increase in steps of $u_s/N = 1280/1280 = 1$ cycle/distance unit until $(u_s/2 - u_s/N) = 639$ cycles / distance unit, is reached.
- For the the l axis, the corresponding frequencies begin at $-v_s/2 = -640$ and increase in steps of $v_s/M = 1280/960 = 4/3$ cycles/distance unit until $(v_s/2 - v_s/M) = 1280/2 - 1280/960 = 638\,\frac{2}{3}$ cycles / distance unit, is reached.

NOTE: When we are examining plots of the DFT, we will always assume that the DFT has been shifted so that zero frequency is at the center.

Windowing in 2D

- We can minimize the contribution of edge effects to the DFT of an image my multiplying the image by a window function.
- A 2D window can be made from a 1D window by taking the outer product of two 1D window vectors.
- Let $\mathbf{w}_N$ represent an N point 1D window vector.
- Let $\mathbf{w}_M$ represent an M point 1D window vector.
- If our image is N columns $\times$ M rows, then the 2D window will be given by:

$$\mathbf{W}_{N\times M} = \mathbf{w}_M \mathbf{w}_N^T$$

Windowing in 2D

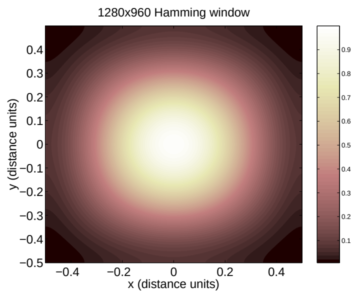
- For example, a 2×3 Hamming window is formed as follows:

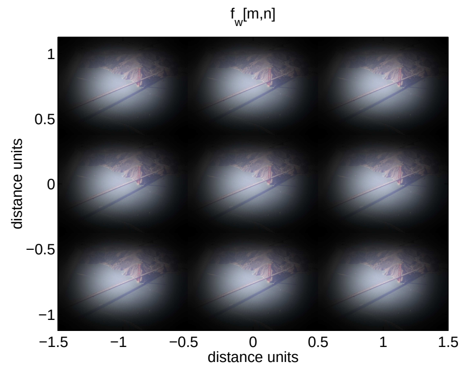
$$\mathbf{W}_{2\times 3} = \begin{bmatrix} 0.08 \\ 1 \\ 0.08 \end{bmatrix} \begin{bmatrix} 0.08 & 0.08 \end{bmatrix} = \begin{bmatrix} 0.0064 & 0.0064 \\ 0.0800 & 0.0800 \\ 0.0064 & 0.0064 \end{bmatrix}$$

Note: Not all 2D windows can be created from 1D windows.

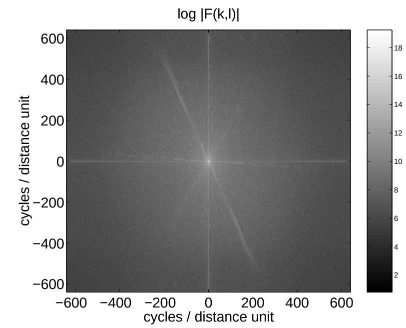
Windowing in 2D

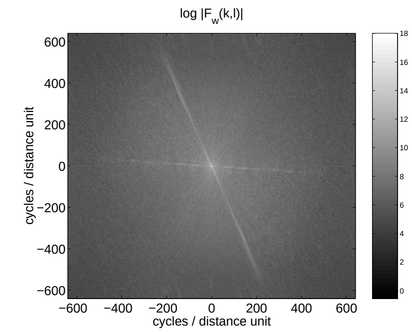
The 2D Hamming window $\mathbf{W}_{1280\times 960}$ is applied to a periodically extended version of $f[m,n]$:





Windowing in 2D





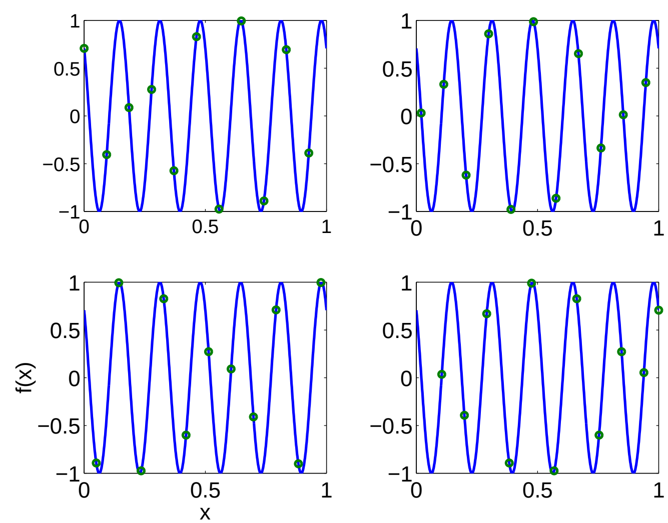
Left: Spectrum with rectangular window.
Right: Spectrum after Hamming window applied.

Note: The wideband vertical and horizontal stripes in the DFT are attenuated. These are directly related to the vertical and horizontal interperiod discontinuities, respectively. The oblique wideband spectral feature (due mainly to the bridge) is retained.

Maximum sampling rate in an image with square pixels

- When we sample a signal, the sampling theorem tells us we must sample at a rate higher than twice the highest frequency component present in the signal.
- Thus, we need to sample the highest frequency sinusoid slightly more than twice per period, or higher.

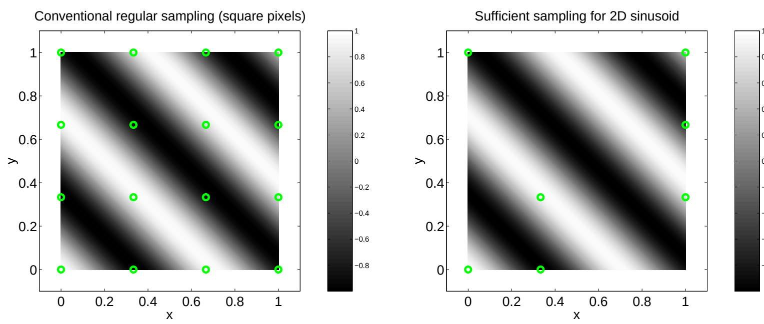
1D signals sampled slightly above Nyquist rate



In all 4 cases, the signal is sampled slightly more than twice per period.

Maximum sampling rate in an image with square pixels

In 2D, we must also sample the highest frequency component more than twice per period. We have a lot more flexibility though:



Because a 2D sinusoid does not vary along lines perpendicular to the direction it is traveling, we may sample it **anywhere** along such a line. The sampling on the right is just as good as that on the left, for this sinusoid.

Maximum sampling rate in an image with square pixels

- As a consequence, the maximum frequencies present in an image sampled using square pixels occur **along the diagonals**.
- A digital camera is thus able to capture the largest amount of fine detail along the diagonals.
- If an image contains $N = 1280$ columns and $M = 980$ rows, and we define the width of the image to be 1 distance unit, then the highest frequency present in the image is:

  $$\rho_{\max} = \sqrt{(u_s/2)^2 + (v_s/2)^2} = \sqrt{640^2 + 640^2} = 905.1 \text{ cycles/distance unit}$$

  and the highest frequency sinusoids that can be sampled are:

  $$\cos(2\pi\rho_m ax\,(x \pm y) + \phi)$$

  where $\rho_{\max} = 905.10$ cycles / distance unit.

Maximum sampling rate in an image with square pixels

Example: Below we have a 32×32 pixel image. Defining an image side as equal to 1 distance unit, we have sampling frequencies of $u_s = v_s = 32$ cycles / distance unit. The image contains:

$$\cos(2\pi 13(-x + y))$$

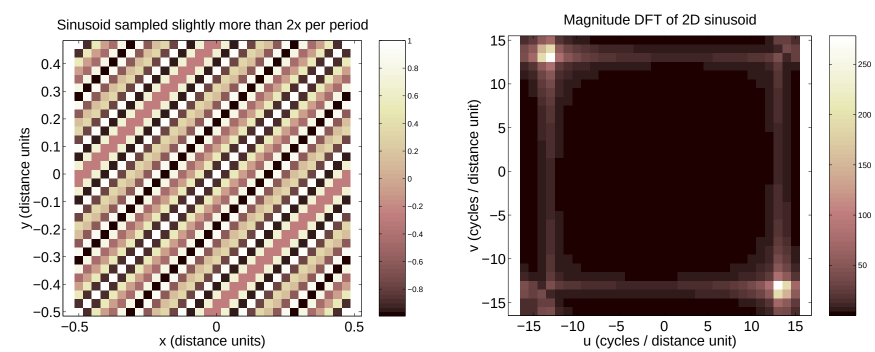
which has a frequency of magnitude:

$$\rho = \sqrt{13^2 + 13^2} = 13\sqrt{2} \approx 18.3848$$

Note that this frequency is **higher than** $u_s/2$ **and** $v_s/2$. However, since it is less than $\rho_{\max} = 16\sqrt{2}$, and travels along the major diagonal, it is properly sampled in a 32×32 image

Maximum sampling rate in an image with square pixels

The image is shown along with its magnitude DFT. Note the spectral peaks near $(u, v) = (-13, 13)$ and $(13, -13)$.



Why are the peaks not pure Kroneker delta functions?

Aliasing in 2D

- Aliasing occurs when an image is sampled at a frequency under twice that of the highest frequency component present.
- Aliasing is explained completely by the relationship between the sampling equation:

$$f_s(x,y) = f(x,y) \times \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(x - k/u_s, y - l/v_s)$$

and its transform:

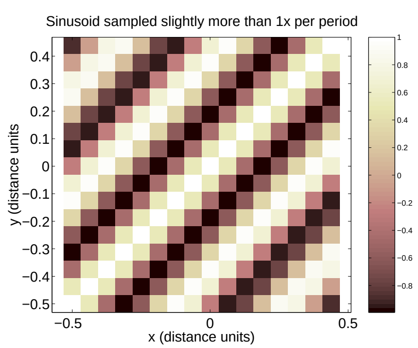
$$F_s(u,v) = F(u,v) * \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(u - ku_s,\, y - lv_s)$$

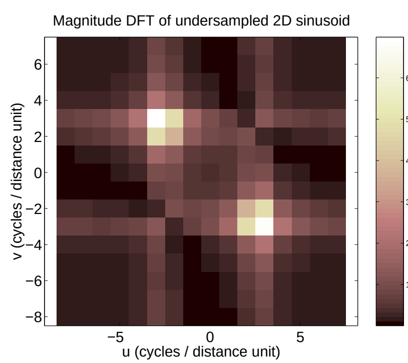
2D aliasing example

- We begin with the "high frequency" image used in the previous example that had a single $\rho = 13\sqrt{2}$ cycles / distance unit component.
- We downsample the image by taking every second sample along both axes.
- The new 16×16 image can contain components with a maximum frequency of $\rho < \rho_{\max} = 8\sqrt{2}$.
- Because $13\sqrt{2} \geq 8\sqrt{2}$, we expect the high frequency component to be aliased and appear as a lower frequency component.
- A sinusoid with horizontal frequency u_0 and vertical frequency v_0 will appear (if aliased) to have a horizontal and vertical frequencies of $(u_s - u_0)$ and $(v_s - v_0)$, respectively.

2D aliasing example

- Below is the downsampled image and its magnitude spectrum:

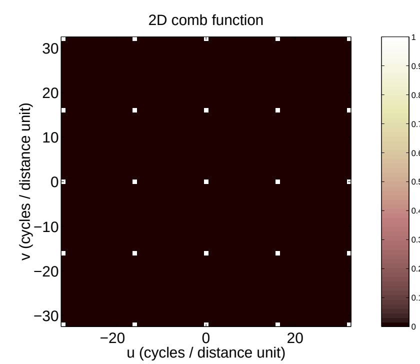


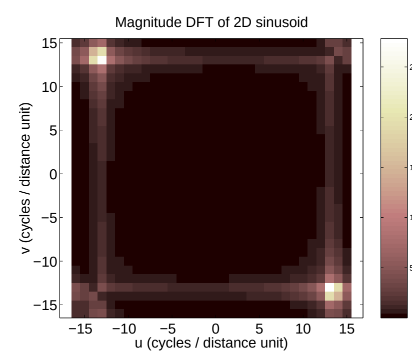


- We can clearly see that the frequency of the bands in the image is lower than that of the original image.
- Is aliasing a linear or a non-linear effect? ________________

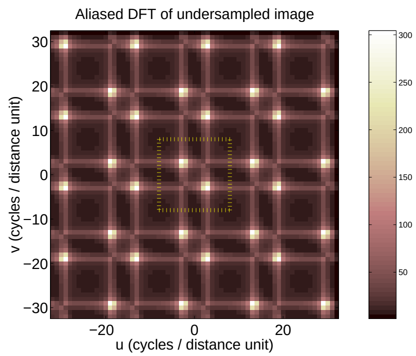
2D aliasing example

- How do we get a better understanding of what has happened? We will execute the instructions given by the FT of the sampling equation i.e., stamp the spectrum of the original image throughout frequency space by convolving the original spectrum with the comb function shown below:





2D aliasing example



- The dotted lines enclose the 1st period of the spectrum of the 16×16 downsampled image.
- This is the same spectrum that we found by DFTing the downsampled image in the previous slide.
- Conclusion: Aliasing is a predictable artifact of undersampling.

Filtering in the frequency domain

We will consider only frequency domain filters associated with linear spatially invariant systems.

$$f(x,y) \longrightarrow \boxed{h(x,y)} \longrightarrow g(x,y)$$

According to the convolution theorem, such filters can be implemented in continuous Fourier space as:

$$G(u,v) = H(u,v)\,F(u,v)$$

and in terms of DFTs as:

$$G[m,n] = H[m,n]\,F[m,n]$$

Filtering in the frequency domain

Advantages:

1. Much faster than convolution for large problems (when FFT is used to implement the DFT).
2. Implementation of filters with specific cut-off frequencies is more direct, flexible and intuitive.
3. Easily implemented in analog form using lenses.

Disadvantages:

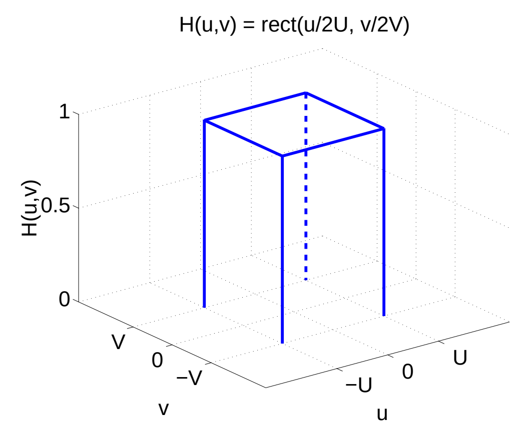
1. Requires assumption of periodicity in space. Consequently, the effects of edge discontinuities must be considered.
2. Filter must be conjugate symmetric or PSF will not be real.

The ideal low-pass filter

For rectangular images, the ideal lowpass filter is given by:

$$H(u,v) = \operatorname{rect}\left(\frac{u}{2U}, \frac{v}{2V}\right)$$

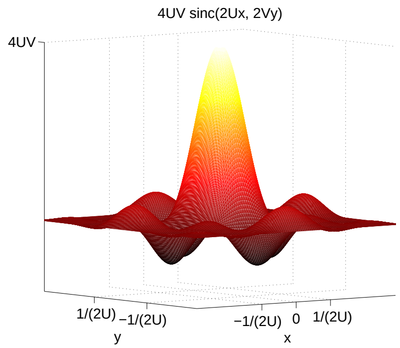
This filter has a cutoff frequency of (U,V).

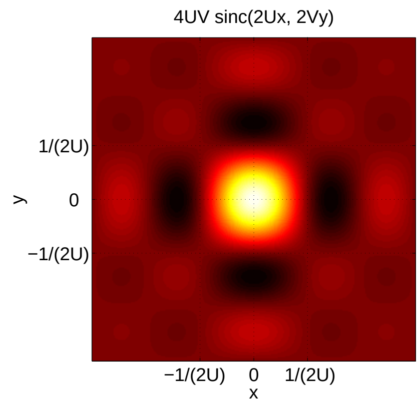


The ideal low-pass filter

The ideal LPF has a point-spread function given by:

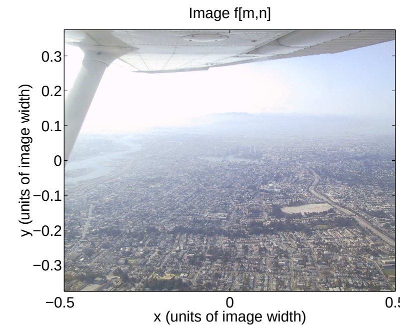
$$h(x,y) = 4UV\mathrm{sinc}(2Ux, 2Vy)$$

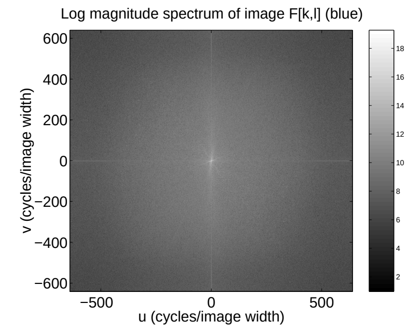




The ideal low-pass filter: Example

We will apply an ideal low-pass filter to image $f[m,n]$ having horizontal and vertical frequency cut-offs at 0.025 and 0.45 of the Nyquist rate, respectively.

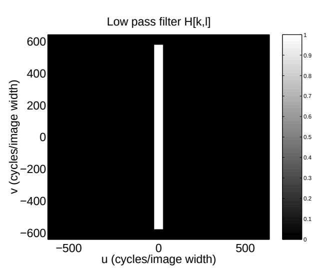


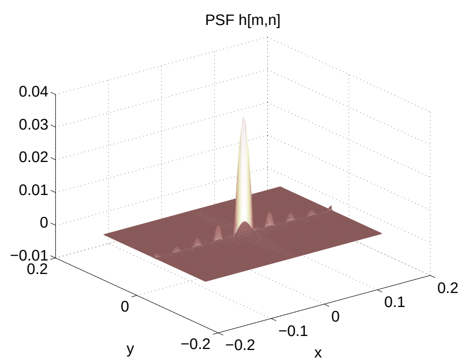


What can we expect the filtered image $g[m,n]$ to look like?

The ideal low-pass filter: Example

The filter and its PSF:

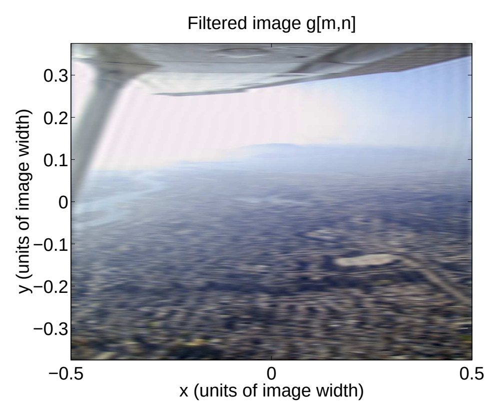




Compare the sinc components in the x and y directions.

The ideal low-pass filter: Example

The filtered image $g[m,n]$:



What do you notice about the smoothness of features in the x and y directions?

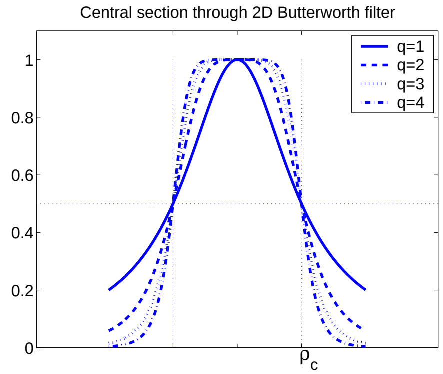
What do we see "ringing" near the strut?

The Butterworth low-pass filter

The transfer function of the 2D Butterworth filter is given by:

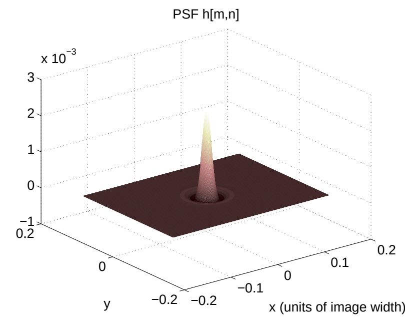
$$H(\rho) = \frac{1}{1 + (\rho/\rho_c)^{2q}}$$

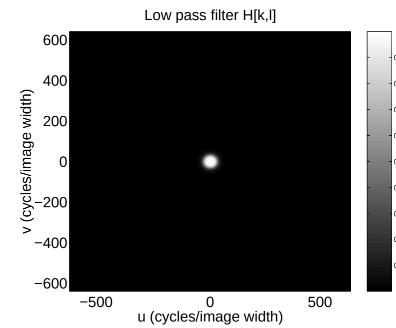
where $\rho = \sqrt{u^2 + v^2}$, ρ_c is the cut-off frequency and q is the order of the filter.



The Butterworth low-pass filter: Example

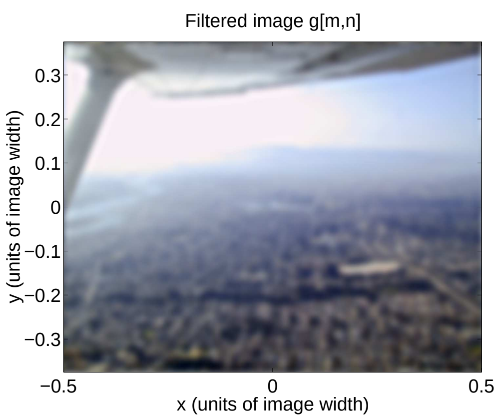
We will apply a Butterworth low-pass filter of order $q = 3$ to image $f[m,n]$. The filter has a radial cut-off at 0.025 of the horizontal Nyquist rate. Below are the PSF and DFT of the filter:





The Butterworth low-pass filter: Example

The resulting image $g[m,n]$:



How does the Butterworth PSF compare to the 2D sinc?

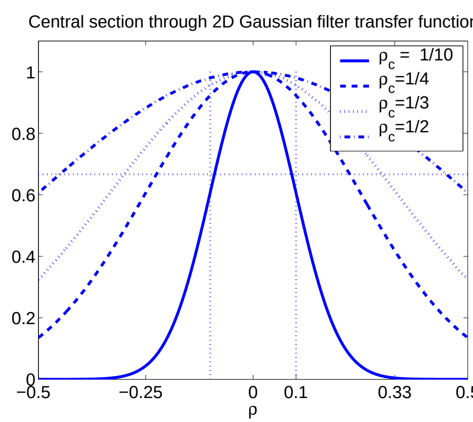
How does the image compare that obtained using the ideal filter?

The Gaussian low-pass filter

The transfer function of the 2D Gaussian filter is given by:

$$H(\rho) = \mathrm{e}^{-\rho^2/2\rho_c^2}$$

where $\rho = \sqrt{u^2 + v^2}$ and ρ_c is the cut-off frequency.

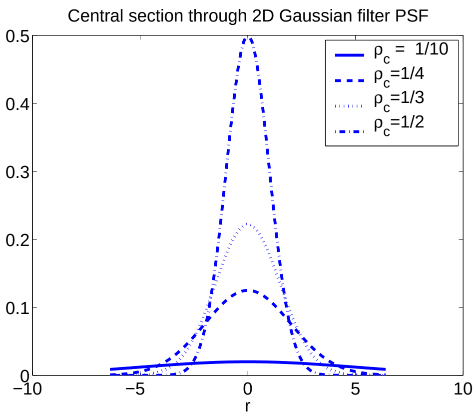


The Gaussian low-pass filter

The PSF is also a 2D Gaussian:

$$h(r) = 2\rho_c^2 \, \mathrm{e}^{-2\rho_c^2 r^2}$$

where $r = \sqrt{x^2 + y^2}$.

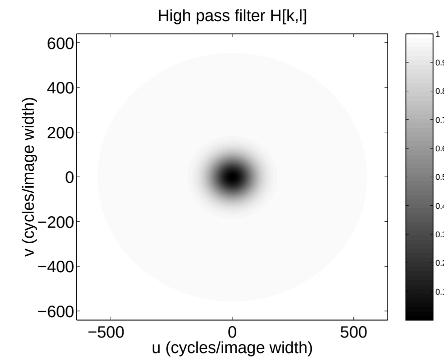


The Gaussian filter: Example

We will apply a Gaussian high-pass filter to image $f[m,n]$. The filter has a radial cut-off at 0.15 of the horizontal Nyquist rate. We make a high-pass filter from the low-pass filter prototype using:

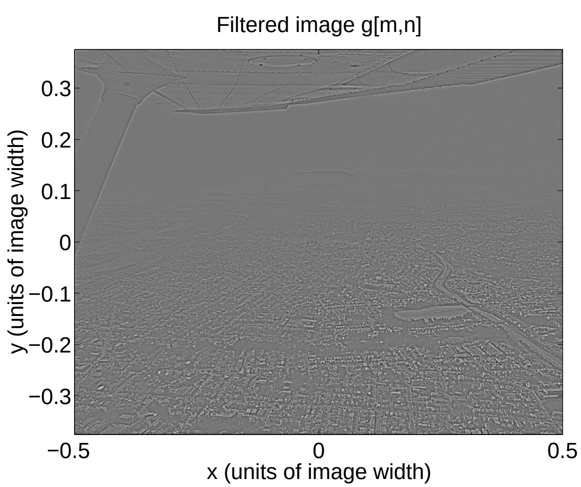
$$H(u,v) = 1 - \mathrm{e}^{-\rho^2/2\rho_c^2}$$

Below is the DFT of the filter:



The Gaussian high-pass filter: Example

The resulting image $g[m,n]$ is:



Note how high-pass filtering enhances edges, but makes smooth regions invisible.

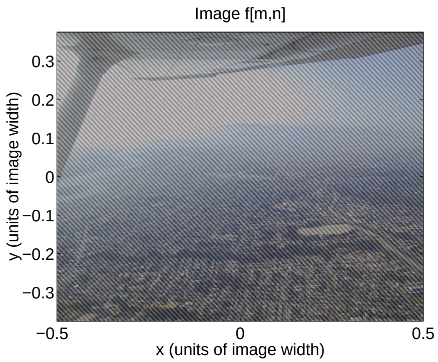
Buliding bandpass and bandstop Fourier domain filters

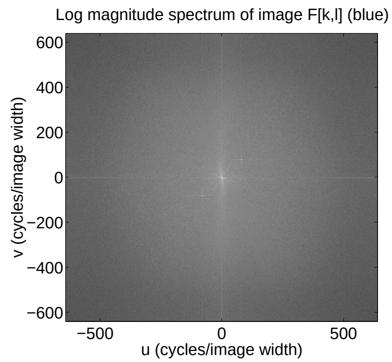
Let $H_l(u,v)$ denote a prototype lowpass filter. Then:

- $H_h(u,v) = 1 - H_l(u,v)$ is a highpass filter.
- $H_{bp}(u,v) = H_h(u,v) + H_l(u,v)$ is a bandpass filter if the lowpass cutoff is larger than the highpass cutoff.
- $H_{bs}(u,v) = 1 - H_{l1}(u,v) + H_{l2}(u,v)$ is a bandstop filter if the cutoff of H_{l1} is larger than that of H_{l2}.

Butterworth bandstop filter: Example

We will try to remove the 80 cycles / 1280 pixel sinusoidal noise from the image $f[m,n]$ using two 10th order Butterworth filters:





Butterworth bandstop filter: Example

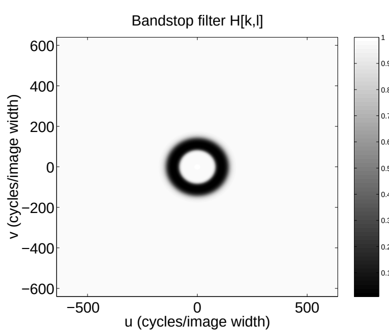
- We choose filter cutoffs by finding the radius of 80 cycle noise in an image sampled at 1280 cycles / image width:

$$\rho_c = 80\sqrt{2} = 113.14$$

- Based on this value of ρ_c, we choose $\rho_{l1} = 153.14$ and $\rho_{l2} = 73.14$ giving a stopband of width 80 cycles.

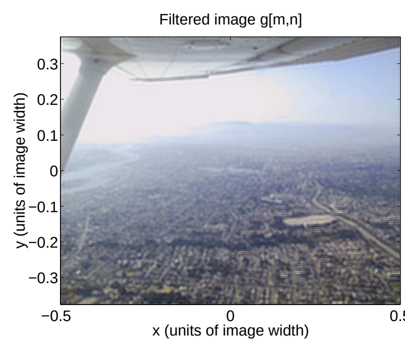
Butterworth bandstop filter: Example

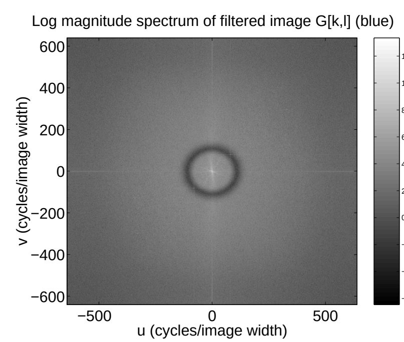
The filter transfer function is:



Butterworth bandstop filter: Example

The resulting image and its spectrum appear below.



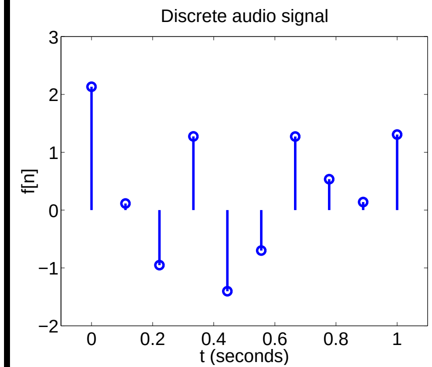


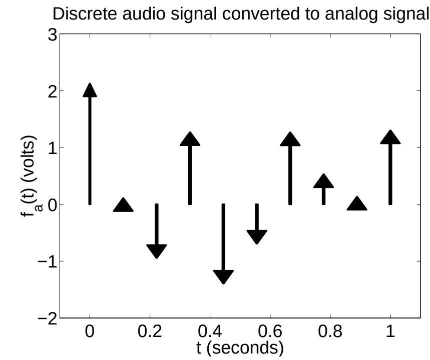
The ideal reconstruction filter

- We sometimes need to convert digital images to continuous analog images.
- For example, the discrete image on the LCD display inside a computer projector must be converted to a continuous image on the projection screen. Failure properly reconstruct the continuous image will lead to a projected image that appears “blocky”.
- To reconstruct a continuous image, we must use an analog filter to retain only a single period of the Fourier transform of the discrete image.
- First, we will review reconstruction filtering in 1D.

Reconstruction filtering in 1D

Reconstruction begins with the discrete signal $f[n]$. The first stage of the ideal digital-to-analog converter produces voltage values proportional to the sample values as $f_a(t)$.





Reconstruction filtering in 1D

- This process is described by the expression:

$$f_a(t) = \sum_{n=-\infty}^{\infty} f[n]\,\delta(t - n/u_s)$$

  where u_s is the original sampling frequency of the signal.
- Because of its discontinuities, the signal $f_a(t)$ has infinite bandwidth. If the discrete signal was sampled at u_s Hz, then by filtering out all frequency components of $f_a(t)$ greater than or equal to $u_s/2$, the original bandlimited signal can be obtained.
- A filter that will achieve this is the ideal lowpass filter with cutoff frequency $u_s/2$:

$$H(u) = \frac{1}{u_s}\mathrm{rect}(u/u_s)$$

Reconstruction filtering in 1D

- Taking the inverse Fourier transform of this filter yields the impulse response:

$$h(t) = \mathrm{sinc}(u_s t) = \mathrm{sinc}(t/T_s)$$

  where T_s is the sampling period.
- Application of this filter gives the reconstructed signal $f_r(t)$:

$$\begin{aligned} f_r(t) &= f_a(t) * \mathrm{sinc}(t/T_s) \\ &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f[n]\,\delta(\tau - nT_s)\,\mathrm{sinc}((t-\tau)/T_s)\,d\tau \\ &= \sum_{n=-\infty}^{\infty} f[n]\,\mathrm{sinc}(t - nT_s) \end{aligned}$$

Reconstruction filtering in 1D

The sinc filter $h(u)$ is called the ideal interpolation filter. What special properties of the sinc function make it ideal for interpolating sampled signals?

1. ______________________________
2. ______________________________
3. ______________________________

Can this filter be implemented digitally?

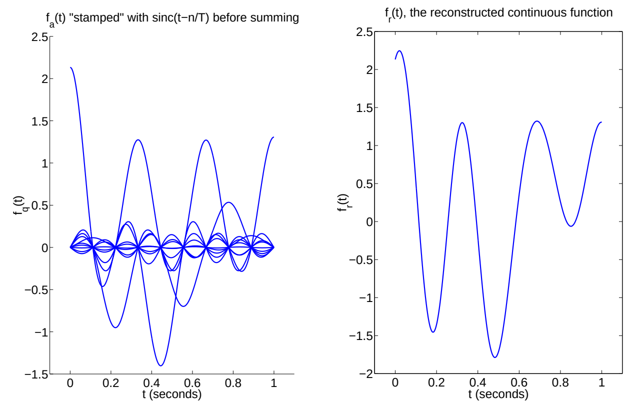
____

Is it practical for use in, say a CD player?

____

Reconstruction filtering in 1D

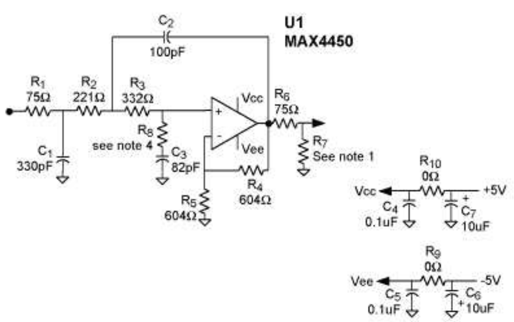
- The left figure shows the scaling and "stamping" of the sinc interpolation function before the sum is carried out.
- The right figure shows the result of the convolution of $f_a(t)$ with the interpolation function.



Why is the interpolation imperfect? ______________________________

Reconstruction filtering in 1D

- Reconstruction filters must be implemented as analog filters. Computers can only input and output discrete digital signals.
- The circuit below shows a 3 pole Butterworth reconstruction filter for use in a digital-to-analog converter:



Schematic diagram courtesy Maxim Integrated Products.

Reconstruction filtering in 2D

Reconstruction in 2D is completely analogous to 1D. For rectangular images, the ideal reconstruction filter is the ideal lowpass filter:

$$H(u,v) = \frac{1}{u_s v_s}\mathrm{rect}\left(\frac{u}{u_s}, \frac{v}{v_s}\right)$$

that has the PSF

$$h(x,y) = \mathrm{sinc}(u_s\, x, v_s\, y)$$

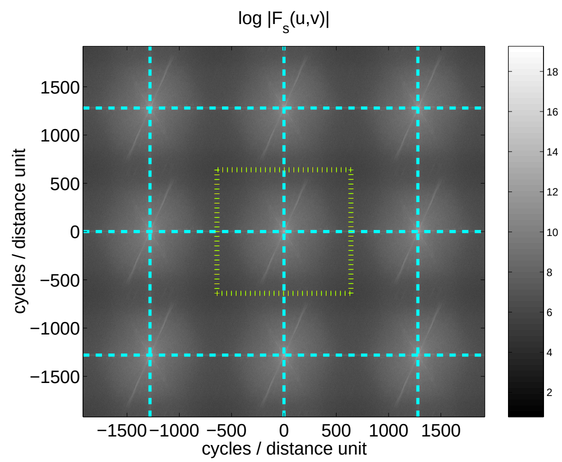
This filter selects out the period of the DSFT closest to the origin.

The 2D sinc function:

1. Contains all frequencies equally up to $u_s/2$ in the x direction and $v_s/2$ in the y direction.
2. Has a value of unity at the origin.
3. Has zero crossings at every non-zero multiple of the sampling periods.

It is the ideal interpolation function for rectangular images.

Reconstruction filtering in 2D



The dotted square endloses the part of the DSFT retained by the ideal reconstruction filter ($u_s = v_s = 1280$ cycles / image width). This is $F(u, v)$, the Fourier transform of the original continuous signal.

Reconstruction filtering in 2D

Most real images are acquired using circular lenses. As a result, all spatial frequencies are sampled equally up to $u_s = v_s = \rho_s$. Consequently, a radially symmetric reconstruction filter:

$$H(\rho) = \frac{1}{\rho_s} \mathrm{rect}\left(\frac{\rho}{\rho_s}\right)$$

that has the PSF

$$h(r) = \mathrm{jinc}(\rho_s r)$$

is sufficient. The jinc function:

1. Contains all radial frequencies equally up to $\rho_s/2$ in all directions.
2. Has a value of ≈ 0.785398 at the origin.
3. Has successive null circles at radii of $\approx 1.220, 2.233, 3.239, \ldots$

This filter is sufficient for the reconstruction of radially bandlimited rectangular images (i.e., uniform resolution images), but is not theoretically ideal.

Reconstruction filtering in 2D

- 2D reconstruction filters must be implemented as analog filters, as was the case in 1D
- Lens systems perform the analog filtering. The FT of the scene (left, imagine this might be the digital projector LCD) appears in the focal plane of the lens. A lowpass filter (opaque sheet with transparent central region) may be employed as reconstruction filter. The reconstructed image is projected onto the rightmost screen.