Computational Vision
U. Minn. Psy 5036
Daniel Kersten
Lecture 9: Neural spatial filtering

Initialize:

```
Off[General::spell1];
SetOptions[ArrayPlot, Mesh -> False, AspectRatio -> Automatic, PlotRange -> All,
  ColorFunction → "GrayTones", DataReversed → True, ImageSize → Tiny];
SetOptions[DensityPlot, Mesh -> False, AspectRatio -> Automatic,
  PlotRange -> All, ColorFunction → "GrayTones", ImageSize → Tiny];
```

Outline

Last time: Linear systems analysis applied to spatial resolution

1. The output or response of a linear system can be modeled as a matrix operation. Convolution is a special case.

If the system is shift-invariant, the matrix has the form of a convolution. Let r be the response, g the input image, and w the filter (or kernel). In one dimension a discrete convolution is:

$$r_i = \sum_{i'} w_{i-i'}\, g_{i'}$$

If the variables are continuous, then

$$r(x) = \int w(x - x')\, g(x')\, dx' = w(x) \otimes g(x) = g(x) \otimes w(x)$$

We can change the order because for the continuous case the convolution operator commutes. In two dimensions,

$$r(x, y) = \int_{-\infty}^{\infty} g(x - x', y - y')\, w(x', y')\, dx'\, dy'$$

In *Mathematica* **ListConvolution[w, g]** calculates discrete convolutions. In the discrete case, one has to deal with the boundaries of an image.

Convolution is used to model how the optics of the eye blur an image. But as we will see in this lecture, it can be used to model how neural image information gets transformed by networks of neurons in the eye and brain.

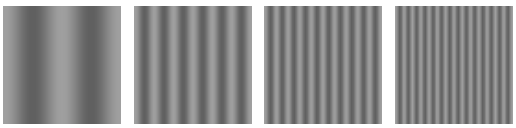
2. Some representations of images are better than others depending on the job of the model:

E.g. spatial frequency representation of images is good for modeling optical transformation.

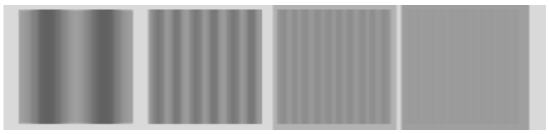
Sinewaves are the eigenfunctions of a shift-invariant system: Modeling can be done by convolution OR one can project the image onto the appropriate basis set providing the spectrum. Then scale each eigenvector by the product of spectrum with the MTF (i.e. eigenvalues of the system), and then add them all up.

3. *Spatial frequency analysis is used to characterize the modulation transfer function (MTF) of the human eye and the contrast sensitivity function (CSF) of the whole visual system (eye and brain).*
The MTF measures the ratio of the output contrast (landing on retina) to the input contrast as a function of spatial frequency, going from low to high:

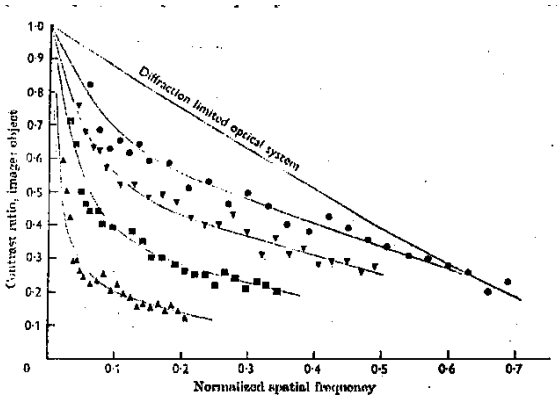
Input



Output



If the input image intensity is l(x) = a_f^{in} Cos[2 π (f x)] + b, contrast is defined as a_f /b. The modulation transfer function (ordinate in figure from Campbell and Green, below) of an optical system is the ratio of the output amplitude to the input: a_f^{out}/a_f^{in}



For small pupils (2 mm in diameter) and high spatial frequencies (approaching 50-60 cycles per degree), the human eye is diffraction limited. In other words, no matter how good the optics are, there is a physical limit to the detail that can be resolved--around 60 cycles/degree, the contrast of a 100% contrast grating drops to zero at the retina.
The spacing of the photoreceptors in the fovea is close to the "Nyquist limit"--specifically, around 120 cycles/degree. Recall that frequency = 1/period, and the period or spacing between foveal photoreceptors is about 0.008 degrees.

4. The MTF tells us how much contrast is lost due to the optics as a function of spatial frequency. The contrast sensitivity function (CSF) tells us how much contrast information is lost through optical and

neural processing as a function of spatial frequency. Sensitivity is the reciprocal of contrast at threshold. Why does the CSF look different than the MTF--i.e. why is the function bandpass? (CSF.gif)

Today: Linear systems and neural image processing

Linear systems models of neural processing

Single-channel spatial filtering

Multiple channel filters

Psychophysical experiments.

->Multi-resolution, and wavelet bases

->A model of the spatial filtering properties of neurons in the primary visual cortex

Understanding the material in this lecture will provide a basis for understanding current research in:

The search for the neural basis of image feature extraction for image representation and recognition

Computer vision models for edge detection, texture processing, ...

Models of human image discrimination performance, image quality metrics, ...

Tutorials and demos

Mathematica tutorial on convolutions

Mathematica tutorial on fourier analysis of images

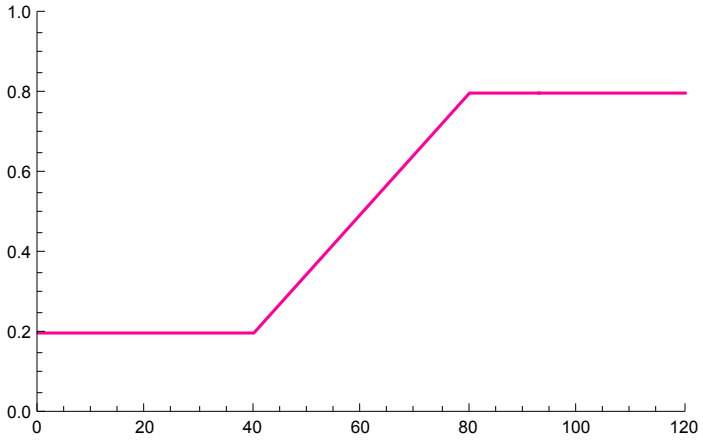
Single channel spatial frequency filtering

Mach bands & perception

Ernst Mach was a 19th century physicist and philosopher known today for a unit of speed and for "Mach's principle" of cosmology, Mach was also interested in sensory physiology and today is also known for several visual illusions. One illusion is called "Mach bands". He noticed that the brightness of a luminance ramp didn't look like one would predict simply from physical measurements of light intensity. Let's make some Mach bands.

```
lb = 40;
ub= 80;
size = 120;
Clear[y];
low = 0.2; hi = 0.8;

y[x_,lb_,ub_] := low /; x<lb
y[x_,lb_,ub_] :=
    ((hi-low)/(size/3)) x + (low-(hi-low)) /; x>=lb && x<ub
y[x_,lb_,ub_] := hi /; x>=ub
```

```
machg = Plot[y[x, 40, 80], {x, 0, 120},
  PlotRange → {{0, 120}, {0, 1}}, PlotStyle → {Hue[0.9]}]
```

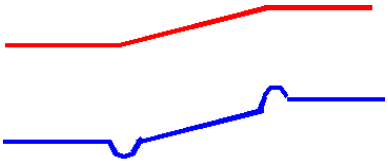


We'll now make a 2D gray-level picture with **ListDensityPlot** to experience the Mach bands for ourselves. **PlotRange** allows us to scale the brightness.

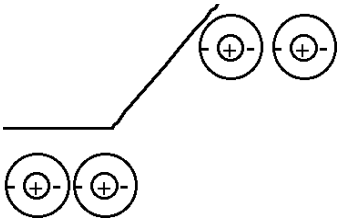
```
picture = Table[Table[y[i, lb, ub], {i, 1, size}], {i, 1, size/2}];
ArrayPlot[picture, Frame → False, PlotRange → {0, 1}]
```

There...what took Mach some effort to set up carefully now only requires a computer, some general purpose software, and a few lines of code.

What Mach noticed was that the left knee of the ramp looked too dark, and the right knee looked too bright. Objective light intensity did not predict apparent brightness.The red line below is proportional to the actual intensity. The blue line shows an informal sketch of apparent brightness.Why is this? Mach advanced an explanation in terms of lateral inhibition, an explanation that hasn't changed much in over 100 years. We'll return to it below. But first let's take a look at what is known about retinal anatomy and physiology.



Mach's explanation

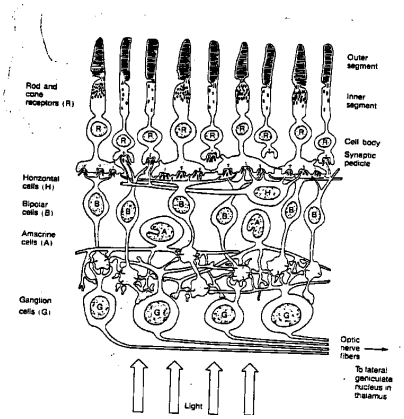


Physiological basis for spatial filtering

The neural basis? Lateral inhibitory filtering in sensory cells may be part of the answer. Found in vertebrates and invertabrates:

Limulus (horseshoe crab)--Hartline won the 1967 Nobel prize for this work that began in the 30's.

vertebrates: Frog - Barlow, and mammals: Cat --Kuffler. Below is a schematic representation of the mammalian retina.



The receptors are connected to horizontal and bipolar cells which in turn are connected to amacrine cells. One overall function of this circuitry is to provide gain control for intensity signals, with high-pass spatial and temporal filtering, so that the retinal output signals light levels relative to an average level, rather than signalling absolute level.The function of the retina may best be seen by a close examination of how its output cells transform image information. These cells are called ganglion cells and they send their axons out of the eye through the optic disk ("blind spot") to the lateral geniculate nucleus (LGN). The LGN has been crudely (and perhaps erroneously) likened to a relay station en route to cortex. Several types of ganglion cells, each with distinctive anatomy and function have been identified in cat and monkey (Enroth-Cugell and Robson, 1966; see Shapley and Perry, 1986 for a comparison with monkey ganglion cells). In cat, the two principle types are the X, Y cells. (There is parallel terminology,

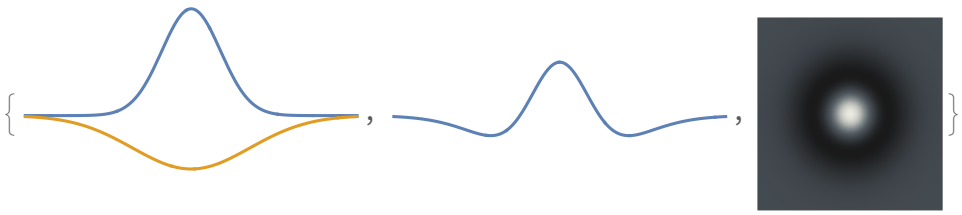
P and M cells, in the monkey literature, for parvocellular and magnocellular.). They code contrast into trains of action potentials (spikes) whose temporal frequency grows with contrast. In addition, these cells act as approximately circularly symmetric spatial-temporal band-pass filters, with small departures from linearity. What this means should become clear after we explain the idea of a receptive field.

If one measures the response of a ganglion cell to a uniformly illuminated screen, one typically finds a mean spike discharge rate (e.g. 50 spikes/sec). Then if a small light spot is positioned on the screen, the cell will increase its firing for some positions, and decrease its firing rate for other positions. A "receptive field" can be mapped out which shows the sensitivity of the cell to the light spots in various locations. One of the striking findings of the 1950's was the discovery that retinal cells almost uniformly show a concentric center-surround organization in which the center is excitatory and the surround inhibitory (or vice versa). The figures below show density plots where light areas represent where the cell increases firing to a spot, and the darker areas where it is inhibited

Three forms for the "mexican-hat" filter: w(x,y)

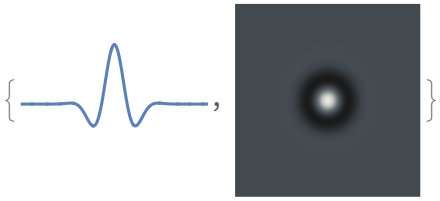
Difference-of-Gaussians (DOG)

```
DOG1[x_, y_, s1_, s2_] := 2 Exp[(-x^2 - y^2) / s1^2];
DOG2[x_, y_, s1_, s2_] := Exp[(-x^2 - y^2) / s2^2];

DOG[x_, y_, s1_, s2_] := 2 Exp[(-x^2 - y^2) / s1^2] - Exp[(-x^2 - y^2) / s2^2];

g1 = Plot[{DOG1[x, 0, 1, 2], -DOG2[x, 0, 1, 2]},
    {x, -4, 4}, ImageSize → Small, PlotRange → {-2, 2}, Axes → False];
g2 = Plot[{DOG[x, 0, 1, 2]}, {x, -4, 4}, ImageSize → Small,
    PlotRange → {-2, 2}, Axes → False];
g3 = DensityPlot[DOG[x, y, 1, 2], {x, -4, 4}, {y, -4, 4}, Mesh → False,
    PlotPoints → 64, PlotRange → {-1.5, 1.5}, ColorFunction → "GrayTones",
    ImageSize → Tiny, Axes → False, Frame → False];
{g1,
 g2,
 g3}
```



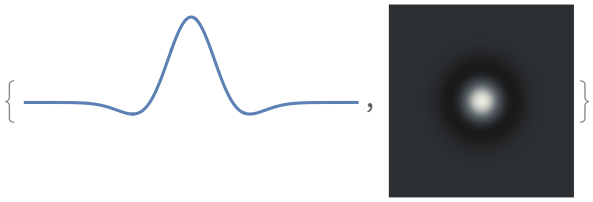
Radial Cosine gabor function

```
cgabor[x_, y_, fx_] := ℯ^(-x^2-y^2) Cos[2 π (fx Sqrt[x^2 + y^2])];
gg1 = Plot[cgabor[x, 0, 0.45], {x, -4, 4},
    PlotRange → {-1, 1}, ImageSize → Tiny, Axes → False, Frame → False];
gg2 = DensityPlot[cgabor[x, y, 0.45`], {x, -4, 4}, {y, -4, 4},
    Mesh → False, PlotPoints → 64, PlotRange → {-1.25, 1.25},
    ColorFunction → "GrayTones", ImageSize → Tiny, Axes → False, Frame → False];
{gg1,
 gg2}
```



∇^2G: Laplacian of a Gaussian

```
blur[x_, y_] := ℯ^(-x^2-y^2);
dx[x_, y_] := ∂u blur[u, v] /.u → x /.v → y
dy[x_, y_] := ∂v blur[u, v] /.u → x /.v → y
delsqG[x_, y_] := ∂{v,2} blur[u, v] + ∂{u,2} blur[u, v] /.u → x /.v → y
ggg1 =
  Plot[-delsqG[x, 0], {x, -4, 4}, Mesh → False, PlotRange → {-5, 5}, Axes → False];
ggg2 = DensityPlot[-delsqG[x, y], {x, -4, 4}, {y, -4, 4},
    Mesh → False, PlotPoints → 64, PlotRange → {-5, 5},
    ColorFunction → "GrayTones", ImageSize → Tiny, Frame → False];
{ggg1,
 ggg2}
```



Spatial filtering

If one thinks of the retina (or more precisely a small patch) as a having a whole array of identical linear ganglion cells, one could also interpret the receptive field as the analog of the point spread function of the optics. So now the above figure would represent the strength of the response of the array of ganglion cells at each location to a small spot of light.

This concentric antagonistic spatial organization is called lateral inhibition. There are both ON-center and OFF- center types of ganglion cells:

```
GraphicsRow[
 {Plot[-delsqG[x, 0], {x, -4, 4}, Mesh → False, PlotRange → {-5, 5}, Axes → False],
  Plot[delsqG[x, 0], {x, -4, 4}, Mesh → False, PlotRange → {-5, 5}, Axes → False]}]
```

How can we quantitatively model the collection of ganglion cell responses over an image? The answer is through convolution.

Let $r_{k,l}$ be the response (in spikes/sec) of a ganglion cell at x-y location (k,l). The average response, to a first approximation, is determined by the weighted sum of the input light intensities, $g_{i,j}$ at spatial location (i,j)

$r_{k,l}=\sum_{i,j} w_{k,l;i,j}\, g_{i,j}$

As with the PSF, we assume spatial homogeneity, and thus shift-invariance:

$r_{k,l}=\sum_{i,j} w_{k-i;l-j}\, g_{i,j}$. Or by suitable arrangement of rows and columns as matrix operation, **r = W.g**

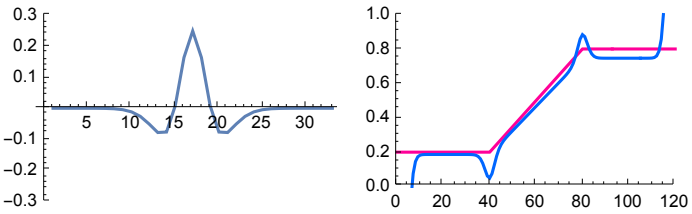
(Neural signals often show small-signal suppression, and high-signal saturation. We can describe this as a simple *point-wise* non-linearity, $\sigma()$, where σ is a sigmoid function (a function with an elongated "*S*" shape. We won't apply this non-linearity yet, and will assume we are usually in a linear regime).

As we have seen earlier, one can describe a linear convolution filtering system either in the spatial or spatial frequency domain. Thinking in frequency terms, a center-surround filter (as in the above figure) can also be interpreted as a band-pass filter, which suppresses low and high spatial frequencies, but leaves middle ones less affected. We return to this below.

So what about Mach bands?

We've seen Mach's qualitative explanation. In Mathematica, it is straightforward to convolve a DOG shaped filter with Mach's luminance ramp to predict a response.

```
LDOG = 0.25 Table[DOG[x, 0, 1/1.9, 2/2], {x, -4, 4, 0.25}]; (*This is the kernel*)
gm1 = ListPlot[LDOG, Joined → True, PlotRange → {-0.3, 0.3}];
r = ListConvolve[LDOG, picture[[1]], {17, -17}];
gm2 = Show[machg, ListPlot[r/0.1, Joined → True,
    PlotRange → {{0, 120}, {0, 1}}, PlotStyle → {Hue[0.6]}]];
GraphicsRow[
 {gm1,
  gm2}]
```



Note that we've scaled the output for comparison. Also, note that we have more positive than negative summed weights in the DOG filter. What would the response look like if we used a more realistic model of retinal center-surround cells, i.e. with no "d.c." response?

► 1. Exercise: What happens (to perception and to the model) when the slope of the ramp increases, approaching that of a step function?

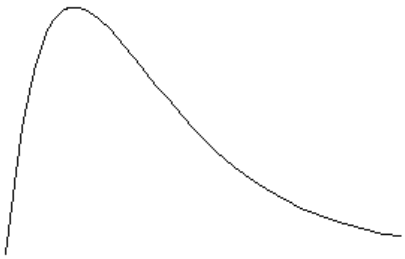
And what about the shape of the Contrast Sensitivity Function (CSF)?

At the end of the previous lecture, we described how your own contrast sensitivity function can be measured.

Can we use the center-surround, Mexican-hat, spatial filter to describe the form of your own contrast sensitivity function? (CSF.gif)

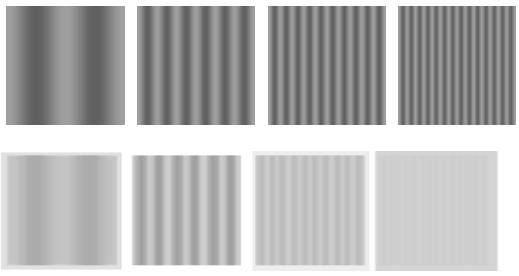
The previous lecture showed that the fall-off in sensitivity at high spatial frequencies could be due in part to optical losses. But there is also a fall-off at the low end. The peak is between 3 and 5 cycles/degree.

Contrast sensitivity as a function of spatial frequency has an inverted U-shape:

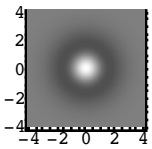


The visual system responds well to contrasts over a middle range, but not as well to low or high spatial frequencies. The visual system is sometimes said to behave as a *bandpass spatial filter*. because it lets

frequencies in a middle range of spatial frequencies pass through better than low or high frequencies.



A center-surround spatial filter can be used to model both the low and high frequency fall-offs. One theory (the "single-channel") theory can account for this if the size and spacings of the excitatory and inhibitory regions of the receptive field weighting function,



are well-matched to a middle frequency. In fact, the amplitude spectrum of the Fourier transform of a DOG shaped filter has the shape of an inverted U -- that is, the contrast amplitudes of low and high spatial frequencies are attenuated relative to the middle frequencies. The optimal frequency would have a periodicity close to that of the receptive field. I
If one uses the simple linear model, then the integral of the receptive field has to be zero (the positive and negative weights should balance each other out); otherwise, the filter will generate a "d.c." offset that is proportional to the mean light level. One of the striking properties of X and Y cat ganglion cells is that the insensitivity to mean absolute light level.

In 1992, Atick and Redlich explained the form of the CSF in terms of an efficient response given the statistical properties of natural images. This theory starts largely from first principles, and explains quantitatively how the CSF changes as a function of mean light level, from scotopic to phototopic conditions.

Although the single-channel theory works here, it doesn't account for other results that we will get to later. One theory that we now turn to is that human CSF is determined by the envelope of the contrast sensitivities of a family of cell types in primary visual cortex of the brain--an idea referred to as multiple channels. We'll start with a little history.

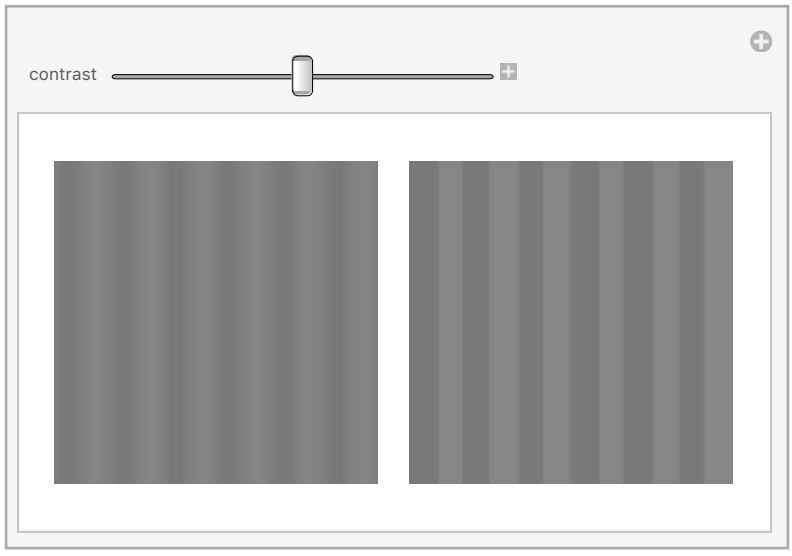
Multiple channel spatial filtering

Campbell and Robson -- 1968 -- Another classic

Contrast sensitivity for a cosine grating and for a square wave grating

Given a cosine grating and a square-wave grating of equal physical contrasts, which one is more detectable near threshold? Or would contrast sensitivity be the same? You could try lowering the lowcontrast variable until you can just barely see the square wave grating, but most displays have no more than 8 bits of graylevel resolution, so your sinewave and squarewave gratings may not be that different after quantization!

```
Grating[x_, y_, fx_, fy_] := Sin[2 Pi (fx x + fy y)];
Square[x_, y_, fx_, fy_] := Sign[Grating[x, y, fx, fy]];

tsine = Table[Grating[x, y, 0, 3], {x, -1, 1, .02}, {y, -1, 1, .02}];
tsquare = Table[Square[x, y, 0, 3], {x, -1, 1, .02}, {y, -1, 1, .02}];

Manipulate[
 GraphicsRow[{ArrayPlot[contrast * tsine, Frame → False, PlotRange → {-1, 1}],
   ArrayPlot[contrast * tsquare, Frame → False, PlotRange → {-1, 1}]}],
 {{contrast, .05}, 0, .1, .001}
]
```



Answer: Human contrast sensitivity is higher for a square wave than for a sine wave.

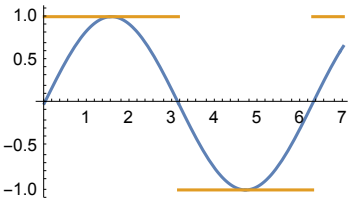
Campbell and Robson measured the CSF for sinusoidal and square-wave gratings. They showed that to be seen equally well (at threshold), one needed about 27% more contrast for a sine wave than for a

square wave. (This was for spatial frequencies higher than the peak frequency of the CSF). Or, putting it another way--if they are both at the same physical contrast, when a square-wave can just be seen, the sine-wave is invisible. Why?

Fourier series for a square-wave grating

Here is a plot of the relative intensity as a function of distance across the screen for the sine and square-wave gratings with the same physical contrast:

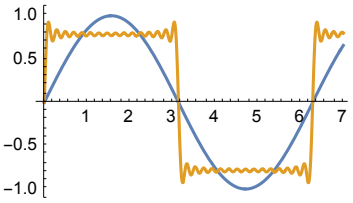
```
square1[x_] := Sign[Sin[x]];
Plot[{Sin[x], square1[x]}, {x, 0, 7}, ImageSize → Small]
```



Fourier showed that (almost) any periodic function could be synthesized from a sum of sine and cosine waves. The fourier series for a square-wave is: $\sum_{n=0}^{\infty} \frac{1}{2n+1} \sin[(2n+1)x]$. The first term in the series (n=0) is sin(x) and is called the fundamental component of the series. The rest of the components are harmonics. Let's plot up the sum of the first 13 terms together with a plot of the fundamental:

```
approxsquare[x_] := Sin[x] + (1 / 3) Sin[3 x] + (1 / 5) Sin[5 x] + (1 / 7) Sin[7 x] +
    (1 / 9) Sin[9 x] + (1 / 11) Sin[11 x] + (1 / 13) Sin[13 x] + (1 / 15) Sin[15 x] +
    (1 / 17) Sin[17 x] + (1 / 19) Sin[19 x] + (1 / 21) Sin[21 x] + (1 / 23) Sin[23 x] +
    + (1 / 25) Sin[25 x] + (1 / 27) Sin[27 x] +
    (1 / 29) Sin[29 x] + (1 / 31) Sin[31 x];
```

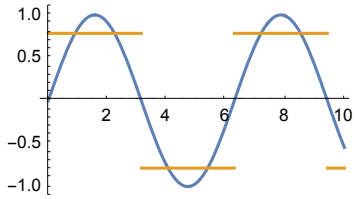
```
Plot[{Sin[x], approxsquare[x]}, {x, 0, 7}, ImageSize → Small]
```



The blue curve shows the fundamental component, which at its peak is clearly higher than the square wave at the same location.

In fact, Fourier analysis tells us that the contrast amplitude of the fundamental is $\pi/4$ (=1.27) greater than that of the square-wave:

```
realsquare[x_] := 1/4 π Sign[Sin[x]];
Plot[{Sin[x], realsquare[x]}, {x, 0, 10}, ImageSize → Small]
```



What does the psychophysics mean for neurophysiology?

Campbell and Robson's conclusion was that the human visual system was analyzing the square-wave grating in terms of its spatial frequency components. They explained the CSF as the envelope of the sensitivites of more narrowly tuned spatial frequency-selective neurons in the visual cortex:

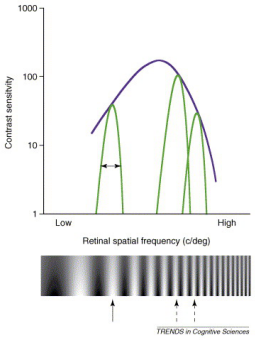
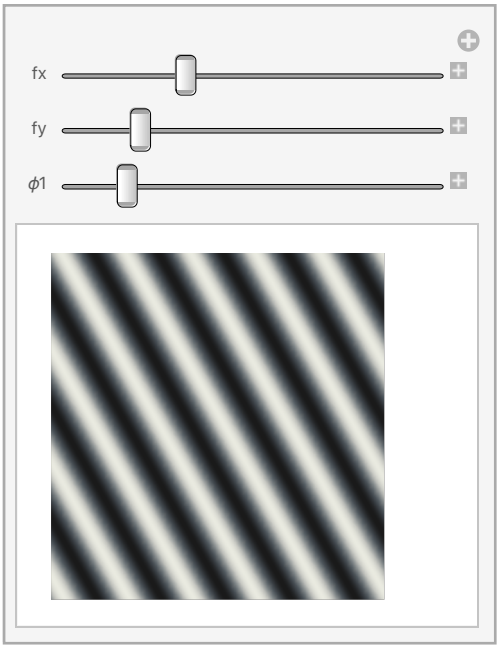


Figure from: Sowden, P. T., & Schyns, P. G. (2006). Channel surfing in the visual brain. *Trends in Cognitive Sciences*, *10*(12), 538–545.

Many found it irresistible to make the conjecture that: *The visual cortex represents complex images in terms of their projections onto the set of sinusoidal grating basis functions* I.e. dot products with weights whose pattern is determined by $\{\cos[2\pi(f_x x + f_y y) + \phi]\}$, or using complex variable notation, with members of the set $\{e^{i2\pi(f_x x + f_y y)}\}$. (Recall that these are eigenfunctions of a linear shift-invariant system).

What would this mean in terms of neurons? One interpretation would be that there are neurons in the visual cortex whose receptive field weights match those of sinusoidal gratings, and whose activities represent the magnitude of the projections. So rather than a circularly symmetric center-surround receptive field (as for the ganglion cell), the receptive field weights (for a range of cells of one orientation) would look like one of these:

```
grating[x_, y_, fx_, fy_, ϕ_] := Sin[2 π (fx x + fy y) + ϕ];
fx = 1 × 1/1; fy = 1/3; ϕ1 = 0;

Manipulate[
 DensityPlot[grating[x, y, fx, fy, ϕ1], {x, -2, 2}, {y, -2, 2}, PlotPoints → 64,
  Mesh → False, Frame → False, PlotRange → {-1, 1}, ColorFunction → "GrayTones",
  ImageSize → Small], {fx, .1, 4}, {fy, .1, 4}, {ϕ1, 0, 2 * Pi}]
```



where various combinations of $\{f_x, f_y, \phi\}$ produce gratings of any spatial frequency, orientation, and phase. Try some variations on the above.

► 2. Exercise: What is orientation θ as a function of f_x and f_y? What is radial frequency as a function of f_x and f_y?

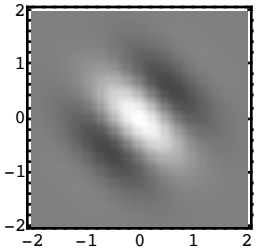
These would be the weights for a collection of cells at just one location. What image pattern would produce the biggest response? Assuming that the patterns are all normalized, the pattern that matches the receptive field weight would give the biggest response.

You might argue that this is a pretty big stretch from a simple psychophysical result. You'd be right. What are the neurophysiological data?

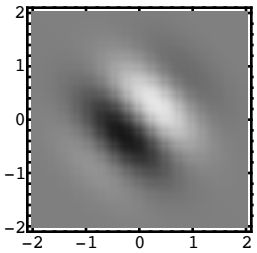
The neurophysiology of neurons in the striate cortex: Hubel & Wiesel

Hubel & Wiesel (1968) characterized the receptive fields of neurons in the mammalian visual cortex in terms of several different classes. Some cell types, called simple cells behaved like "bar detectors", and others like "edge detectors". Their sensitivities are illustrated in the figures below.

"Bar detector" receptive field



"Edge detector" receptive field



Conclusion-"global" fourier representation of images NOT consistent with neurophysiology

Although neurons in primary visual cortex showed spatial frequency and orientation selectivity, their receptive fields are localized in space. This suggested that a spatial frequency channel would be represented by a population of neurons with the same orientation and spatial frequency selectivity, but aggregated over space. In the next section we see how to model neural receptive fields that are localized in space and spatial frequency.

V1 receptive fields: localization in space and spatial frequency

Quick overview from retinal to primary visual cortex: As an image strikes the retina, light energy is converted by each receptor to an electrical signal. The discrete nature of retinal sampling suggests a pixel-like representation of the image. The retina further processes the incoming image signal, and the "neural image" as represented by ganglion cell outputs is bandpass filtered, but these filters are approximately circularly symmetric. The axons of the ganglion cells terminate in the lateral geniculate nucleus, where the receptive fields also show lateral inhibition, and are circularly symmetric. Like ganglion cells, the positive and negative weights tend to cancel out.

Measurements of the responses of single neurons in the primary visual cortex also behave as spatial filters; but they are systematically selective for both spatial frequency *and* orientation, and like ganglion cells, their receptive fields have a limited spatial extent, as in the bar and edge detectors above.

How can the receptive fields be modeled?

"Gabor filters": Localization in frequency vs. localization in space (see Appendix)

A single point or pixel representation is said to be localized in space. The amplitude spectrum of a single point is broadband, i.e. it has all spatial frequencies. In contrast, a sinewave grating is global extending indefinitely over space. But its amplitude spectrum is a narrowband, in fact infinitely narrow--just a point in frequency space.

Is there a compromise for localization in space and frequency?

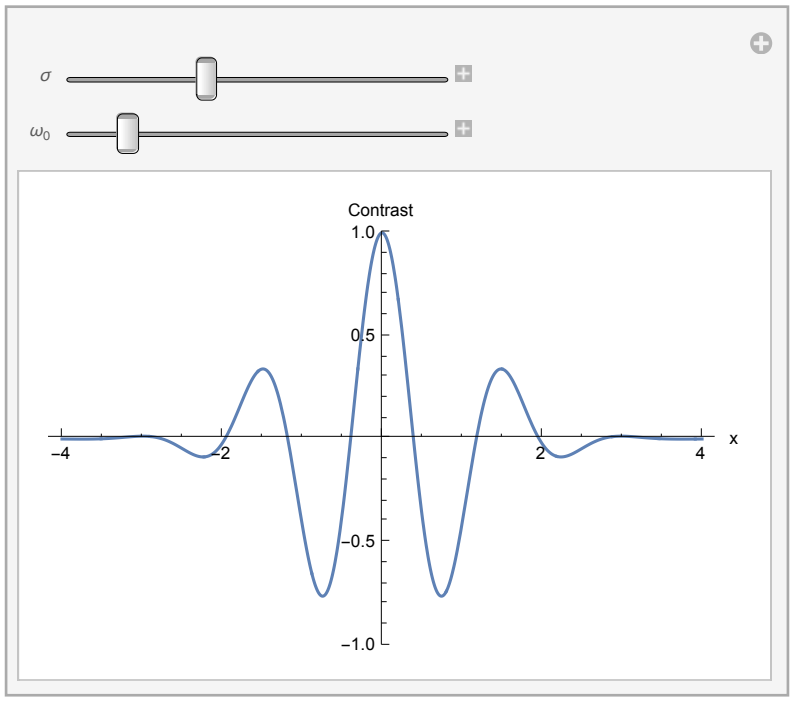
The "Gabor functions":

{Exp[-(x / σ)^2] Cos[ω_0 x], Exp[-(x / σ)^2] Sin[ω_0 x]} are said to be "well-localized" in both space and time.

Dennis Gabor showed that of all basis functions, "Gabor functions" achieved an optimal compromise (according to criteria he specified) in achieving simultaneous compaction in both frequency and space (or time).

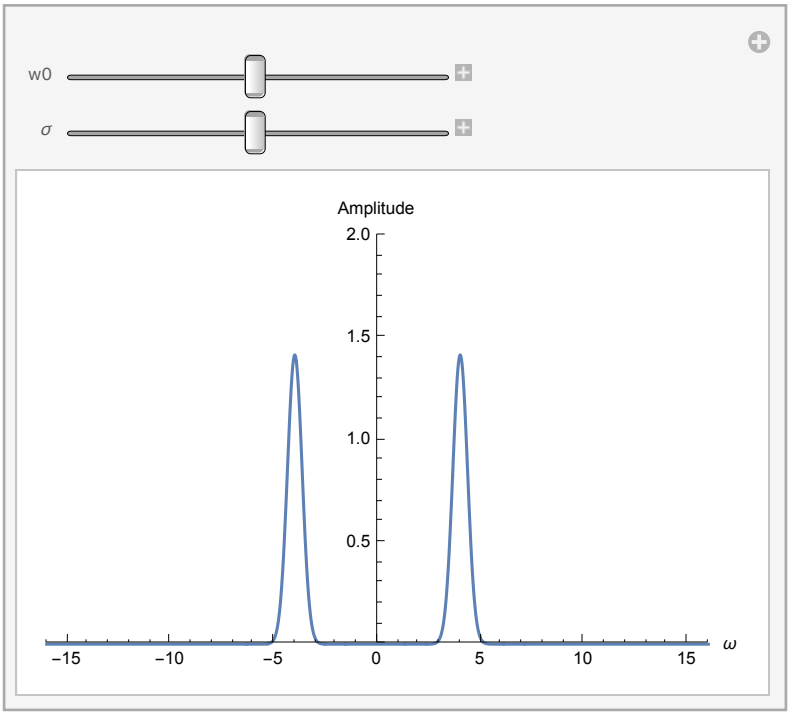
Plot a one-dimensional gabor function:

```
σ = 1; ω0 = 4;

Manipulate[
 Plot[e^-(x/σ)^2 Cos[ω0 x], {x, -4, 4}, PlotRange → {-1, 1}, AxesLabel → {"x", "Contrast"}],
 {{σ, 1}, .1, 4}, {{ω0, 4}, .1, 32}, ImageSize → Tiny]
```



Plot the frequency spectrum of a gabor function:

```
σ = 1; ω = 4;

Manipulate[
 Plot[((1 + E^(w0 * σ^2 * w)) * Abs[σ]) / (E^((1 / 4) * σ^2 * (w0 + w)^2) * (2 * Sqrt[2])),
  {w, -16, 16}, PlotRange → {{-16, 16}, {0, 2}},
  AxesLabel → {"ω", "Amplitude"}], {{w0, 4}, .1, 8}, {{σ, 4}, .1, 8}]
```



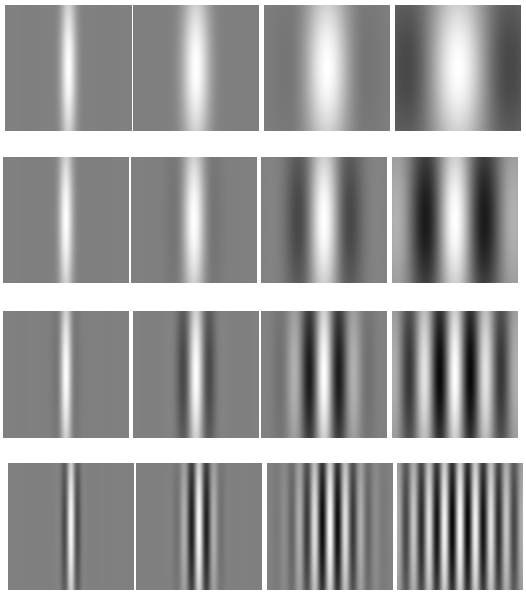
The spectrum is a pair of "blurred out" delta functions, i.e. not as precisely localized in frequency as a pure sinusoid, but nevertheless a gabor function's spectrum is focused around a particular frequency.

Psychophysical clues to human spatial image representations: What does the eye see best?

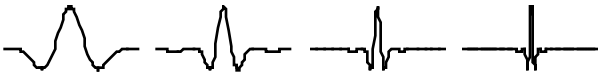
Human efficiency for detecting gabor patches

Burgess, Wagner, Jennings and Barlow (1981) combined the SKE observer and spatial frequency analysis of human vision to find out how efficiently humans detected patterns. They showed in a 1981 Science article that narrowly windowed sinusoids were detected with high efficiency (>70%) when added to static visual noise. Further, these targets were detected more efficiently than disks of light. You basically have all the tools to replicate the experiment of Burgess et al. You can compute d' for the ideal observer for signal-known-exactly patterns. And you can generate Gaussian-windowed sinusoids and add them to gaussian white noise. If you measure the percent correct, and convert that to d' for the human observer, you can calculate the absolute efficiency for human detection--and contribute to

answer the question of what the eye sees best.
Watson, Barlow & Robson (1983) found that that a 7 c/deg grating drifting at 4 Hz, (with a narrow gaussian envelope in space and time) was detected more efficiently than other patterns. Further, the quantum efficiency was very low (<0.05%).
Kersten (1984) measured efficiency for 1-d gratings (i.e. vertical) in temporal (1-d spatial) visual noise for various spatial frequencies and widths.



Peak efficiency was found for patterns of about the same shape, regardless of spatial frequency. The cross-sectional profiles for high efficiency patterns corresponded to the diagonals in the above graph and looked like:



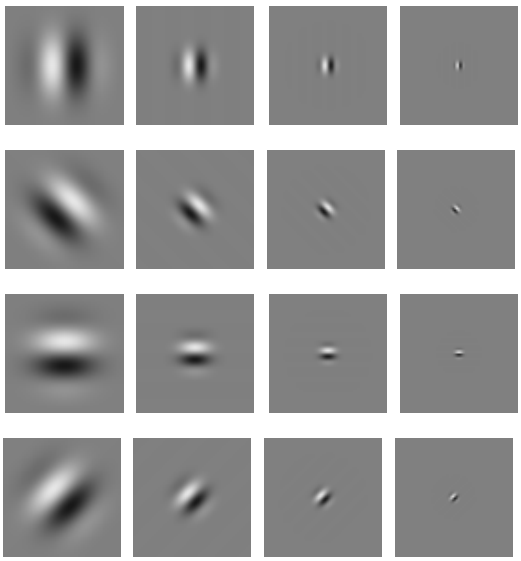
But this pattern of spatial integration might have been lucky, a consequence of using 1D stimuli and 1D dynamic noise. Recently, Morgenstern & Elder (2012) used 2D gratings in static noise, and found that efficiency for spatial integration continued uniformly out for more cycles.

Bottom-line: image coding in terms of scale and orientation: A model for human spatial image representation

When the filters have the same shape except for a change of scale ($x \rightarrow \alpha x$), they are called *self-similar*.

At each spatial location, project the image onto a collection of basis vectors (i.e. compute the dot product) that span a range of *spatial scales* and *orientations.* A subset of these basis images is illus-

trated below:



In general, these neural models of basis functions may be over-complete, and non-orthogonal. And there may be a range of phases. Above we show only the "sine-phase" or "edge-detectors" of Hubel and Wiesel.

The self-similar idea is important to vision because of the need for some kind of scale-invariance. Further, the self-similar aspect of these neural models bore a close resemblance to the emerging mathematical field of wavelet analysis (see Hubbard, 1998). The emphases are different--over-completeness may be important and vision does the projections in parallel (the serial algorithmic component of wavelet computation is integral to its mathematical interest), and the representation of orientation is important for neurocomputations.

Neural images: Gabor filtering of image of a face

A collection of spatial filters all of of the same type, but operating at different locations, is called a channel. A channel can be thought of as a a sub-population of neurons with similar properties but varying along a key dimension such as spatial frequency and/or orientation. One can use convolution to produce a "neural image" to represent the spatial distribution of activity in a topographic representation of the filtered image information in a subset of neurons that all share the same basic spatial filter template.

Mathematica has built-in functions to do convolutions directly on image type data structures. Here’s a kernel (recall, “kernel” is the filter to use in the convolution) with a vertical orientation preference.

```
ImageConvolve[ , {{-1, 0, 1}, {-2, 0, 2}, {-1, 0, 1}}] // ImageAdjust
```

But we can use results from linear systems theory, or Fourier theory more specifically, to do the convolution by multiplying the spectrum of the image with the spectrum of the gabor filter, and taking the inverse Fourier transform. In the previous lecture, we learned the concepts, here we apply them. We won't go into the details here (Fourier theory has some apparent complexities, like using basis functions that involve complex numbers), but the essential ideas remain the same.
The method is based on the result:

image*kernel = F^{-1}(F (image) F(kernel)),

where * means "convolution" and F is the fourier transform, and F^{-1} is its inverse.

The input image -- face

```
face = ImageData[ ];
```

```
size = Dimensions[face][[1]];
hsize = size / 2
```

```
32
```

Find the amplitude spectrum (spectrum) and the phase spectrum (phase) for the face picture. Chop[] sets small values to zero. Note that most of the contrast energy is near zero.

```
(*We use shift to align the filter with the image coordinates*)
shift[mat_,size_] :=
    Transpose[RotateRight[Transpose[RotateRight[mat,size]],size]]
```

Take the fourier transform of the image of the face:

```
faceft = Fourier[face];
facespectrum = Chop[Abs[faceft]];
facephase = Chop[Arg[faceft]];
```

Fourier: Argument face is not a non–empty list or rectangular array of numeric quantities.

Define two spatial filters, one even-symmetric (cosine-phase gabor), and one odd-symmetric (sine-phase gabor). We'll just use the sine-phase gabor to define the filter. You can try the other one.

```
cgabor[x_,y_, fx_, fy_,sig_] :=
N[Exp[(-x^2 - y^2)/(2 sig*sig)] Cos[
    2 Pi (fx x + fy y)]];

sgabor[x_,y_, fx_, fy_,sig_] :=
    N[Exp[(-x^2 - y^2)/(2 sig*sig)] Sin[
        2 Pi (fx x + fy y)]];
filter = Table[sgabor[i/32,j/32,4,4,1/16],
    {i,-hsize,hsize-1},{j,-hsize,hsize-1}];

filterft = Fourier[shift[filter, hsize]];
(*ArrayPlot[fface=Chop[t=InverseFourier[filterft faceft]],Mesh→False];*)
fface = Chop[t = InverseFourier[filterft faceft]];
Image[fface] // ImageAdjust
```

Note: This kind of operation is common in image processing filtering, not just neural modeling. *Mathematica* has a built-in function for the Laplacian of a Gaussian which can operate directly on an image type:

http://reference.wolfram.com/mathematica/ref/LaplacianGaussianFilter.html

Neural image? Or neural image representation?

We can view the response activities of a family of receptive fields of neurons as representing a filtered neural image of the input image. Although useful, this view can be misleading when we start to think about function, for "who is looking at the image"?
Alternatively, thinking in terms of basis functions gives us another perspective. We can view the response activities of a family of receptive fields as a representation of the input image. If linear, an activity is the result of a projection of an image on to a basis function (receptive field weights). Given such a representation we can begin to ask questions like:
1. Is the neural basis set complete? Can any image be represented?
2. A closely related question is: Is any information lost? I.e. we do the inverse transformation, can the original input be reconstructed?
3. Maybe the neural basis set is "over-complete"?
4. Are the neural basis functions orthogonal? Are they normal?
5. How should localized filters be placed in space?

How should localized filters be placed in spatial coordinates?

The Nyquist-Shannon sampling theorem says that when we discretely sample a continuous signal whose highest frequency is f_c, we can restore the original continuous signal from the discrete one, if we have sampled the original at least as frequently as 2 x f_c. In other words, we should take samples at spacings no further apart than 1/2 x f_c.

The intuition is that to capture the highest frequency, you need to at least get a samples from the alternating bright and dark bars. That is, twice as frequently as the bright bars occur.

This general idea has implications for our spatially localized filters too. It suggests that the filters capturing the low-spatial frequency components don't need to be spaced as closely as those sampling higher spatial frequencies.
This has led to the idea that the neural populations in the primary visual cortex may be doing something similar. I.e. that the neurons whose activity represents the low-frequency filters are not only larger, but are more widely spaced. And that a full representation of the image can be captured by an "image pyramid", where the top levels represent information at the coarse scale, and bottom levels at a fine scale (See: Adelson, E. H., Simoncelli, E., & Hingorani, R. (1987) and Watson (1987)).
Mathematica provides a wavelet toolbox. E.g.

```
func[x_, {___, 1 | 2 | 3}] := ImageAdjust[ImageAdjust[ImageApply[Abs, x]], {0, 2}];
func[x_, wind_] := ImageAdjust[x];
```

```
dwd = DiscreteWaveletTransform[
```

```
, SymletWavelet[2], 4];
```

```
WaveletImagePlot[dwd, Automatic, func]
```

Next time

What is the computational use of a wavelet-like decomposition?
Edge detection?
    -> analysis of what vision needs to recognize objects, etc..
Image manipulations, useful for final projects
    Psychophysical techniques--"classification images", reverse correlation

Appendices

"Gabor filters": Localization in frequency vs. localization in space

Fourier transform

Often it is more convenient to work with the general continuous Fourier case (rather than the discrete). Then, we need to use complex numbers to represent the sinusoidal gratings, their amplitudes and phases. Complex numbers can be written in polar notation where the complex number has a length r = **Abs[z]** and an angle θ = **Arg[z]** that it makes with the real axis:

z = r (cos θ + i sin θ).

Using Euler's formula, $e^{i\theta}$ = cos θ + i sin θ,

we can put the length and angles back together:

z = Abs[z] Exp[I Arg[z]], (where i=I).

Given a function f(x), the fourier transform is given by:

$$\text{Fourier transform}: F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)\, e^{i\omega x}\, dx,$$

where $F(\omega)$ is in general a complex number. Arg[$F(\omega)$] gives the phase spectrum, which specifies how the gratings should be offset relative to each other, and Abs[$F(\omega)$] gives the amplitude spectrum, which specifies the amplitude of each grating.

$$\text{Inverse fourier transform}: f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega)\, e^{-i\omega x}\, d\omega$$

Localization in space: Fourier transform of a delta function is constant

The Dirac delta function is precisely localized in space,

```
FourierTransform[DiracDelta[x], x, ω]
```

$$\frac{1}{\sqrt{2\pi}}$$

but if you take its Fourier transform, you observe that it is a constant, $\frac{1}{\sqrt{2\pi}}$, i.e. the same value spread out over the whole frequency spectrum.

Localization in frequency: What is the Fourier transform of Cos[ω_0x]? ($\omega = 2\pi f$)

Cosine is spread out in space,

```
FourierCosTransform[Cos[ω₀ x], x, ω]
FourierCosTransform[Cos[x 4₀], x, 4]
```

but is precisely located in frequency.

Is there some compromise?

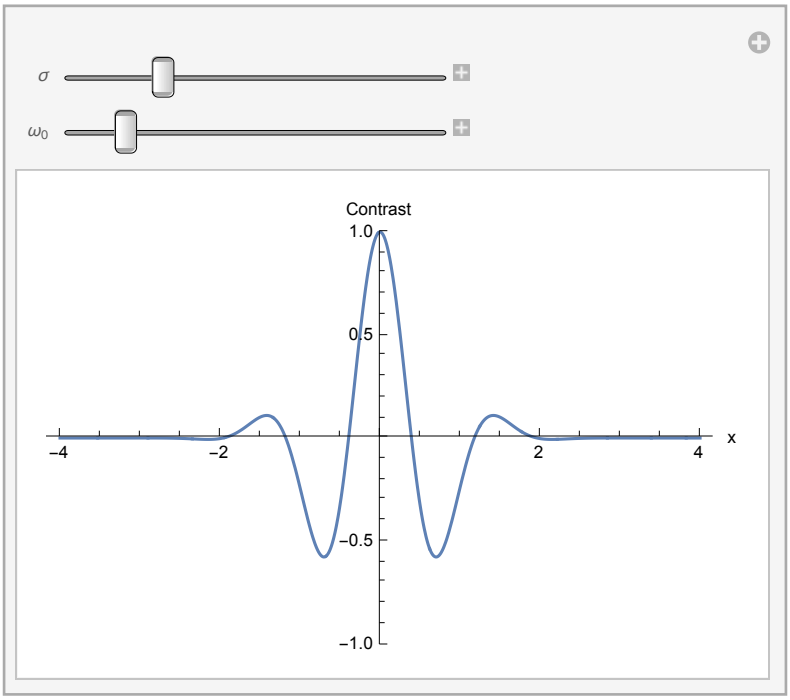
The "Gabor functions":

{Exp[-(x / σ)^2] Cos[ω₀ x], Exp[-(x / σ)^2 Sin[ω₀ x]} are said to be "well-localized" in both space and time.

Heisenberg had shown the uncertainty principle--simultaneous exact localization in space and momentum was not possible. Dennis Gabor showed that of all basis functions, "Gabor functions" achieved an optimal compromise (according to criteria he specified) in achieving simultaneous compaction in both frequency and space (or time).

Plot a gabor function:

```
σ = 1; ω₀ = 4;

Manipulate[
 Plot[e^-(x/σ)^2 Cos[ω₀ x], {x, -4, 4}, PlotRange → {-1, 1}, AxesLabel → {"x", "Contrast"}],
 {{σ, 1}, .1, 4}, {{ω₀, 4}, .1, 32}]
```



Plot the frequency spectrum of a gabor function:

```
Clear[σ, ω, w0]
```

```
FourierTransform[ℯ^-(x/σ)^2 Cos[w0 x], x, ω, FourierParameters → {0, 1}]
```

$$\frac{1}{2\sqrt{2}\sqrt{\frac{1}{\sigma^2}}}\left(1+\text{Cosh}\left[w0\,\sigma^2\,\omega\right]+\text{Sinh}\left[w0\,\sigma^2\,\omega\right]\right)\left(\text{Cosh}\left[\frac{1}{4}\sigma^2\,(w0+\omega)^2\right]-\text{Sinh}\left[\frac{1}{4}\sigma^2\,(w0+\omega)^2\right]\right)$$

```
FullSimplify[%]
```

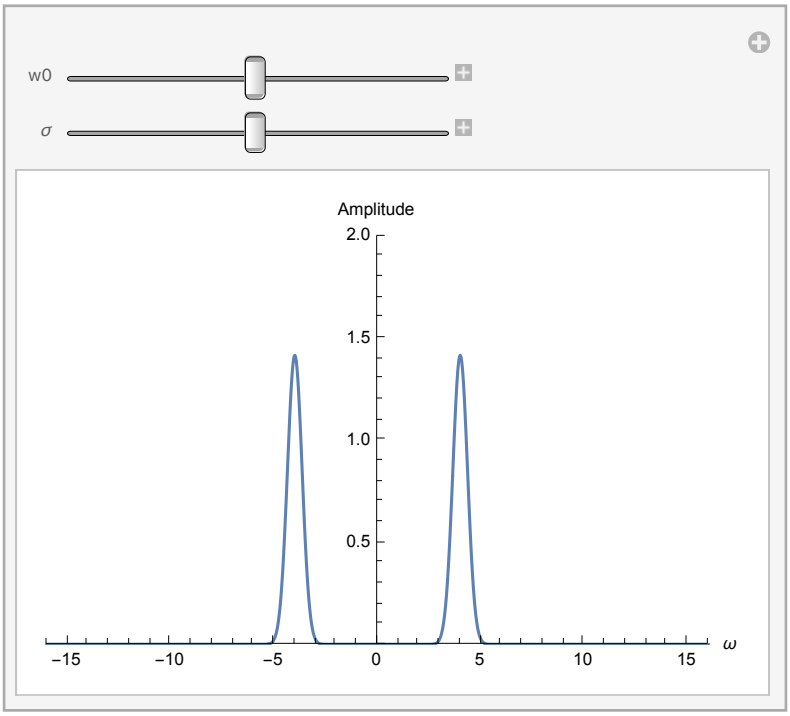
$$\frac{e^{-\frac{1}{4}\sigma^2\,(w0+\omega)^2}\left(1+e^{w0\,\sigma^2\,\omega}\right)}{2\sqrt{2}\sqrt{\frac{1}{\sigma^2}}}$$

Convert the above cell to InputForm:

```
((1 + E^(w0 * σ^2 * ω)) * Abs[σ]) / (E^((1 / 4) * σ^2 * (w0 + ω)^2) * (2 * Sqrt[2]))
```

$$\frac{e^{-\frac{1}{4}\sigma^2\,(w0+\omega)^2}\left(1+e^{w0\,\sigma^2\,\omega}\right)\,\text{Abs}[\sigma]}{2\sqrt{2}}$$

```
σ = 1; ω = 4;

Manipulate[
 Plot[((1 + E^(w0 * σ^2 * w)) * Abs[σ]) / (E^((1 / 4) * σ^2 * (w0 + w)^2) * (2 * Sqrt[2])),
  {w, -16, 16}, PlotRange → {{-16, 16}, {0, 2}},
  AxesLabel → {"ω", "Amplitude"}], {{w0, 4}, .1, 8}, {{σ, 4}, .1, 8}]
```



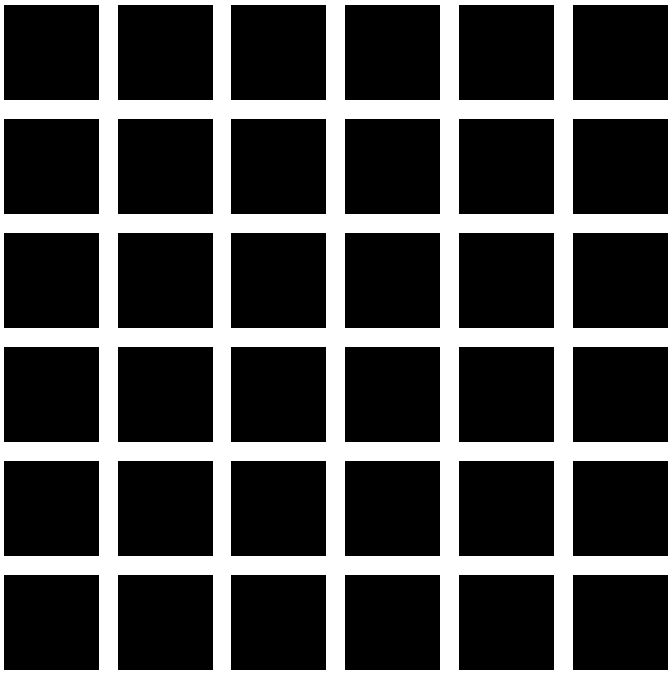
The spectrum is a pair of "blurred out" delta functions, i.e. not as precisely localized in frequency as a pure sinusoid, but nevertheless a gabor function's spectrum is focused around a particular frequency.

► 3. Exercise: Hermann grid

Below is the Hermann Grid. Notice the phantom dark spots where the white lines cross. Can you explain what you see in terms of lateral inhibition?

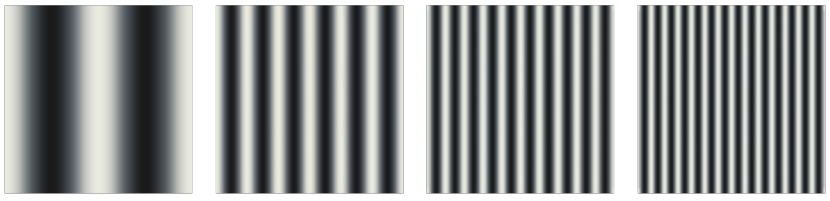
```
width = 5; gap = 1; nsquares = 6;

hermann = Flatten[Table[{Rectangle[{x, y}, {x + width, y + width}]},
{x, 0, (width + gap) * (nsquares - 1), width + gap},
    {y, 0, (width + gap) * (nsquares - 1), width + gap}], 1];
```

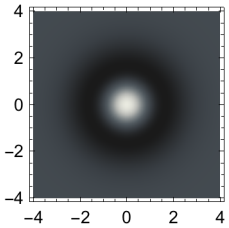
```
Show[Graphics[hermann, AspectRatio → 1]]
```



4 spatial frequencies & center-surround weighted filter

```
Grating[x_, y_, fx_, fy_] := Cos[2 π (fx x + fy y)];
g = Table[DensityPlot[0.25` Grating[x, y, fx, 0], {x, -1, 1}, {y, -1, 1},
    PlotPoints → 64, Mesh → False, Frame → False, PlotRange → {-1, 1}], {fx, 1, 7, 2}];
f = Table[DensityPlot[DOG[x, y, 1, 2], {x, -4, 4}, {y, -4, 4},
     Mesh → False, PlotPoints → 64, PlotRange → {-1, 1}], {fx, 1, 1}];
Show[GraphicsRow[{g[[1]], g[[2]], g[[3]], g[[4]]}]]
Show[GraphicsRow[{f[[1]]}]]
```





Gabor basis set

Basis set: Cartesian representation of Gabor functions:

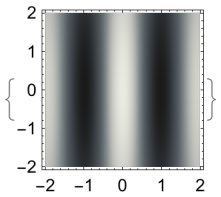
```
cgabor[x_,y_, fx_, fy_,s_] :=
Exp[-(x^2 +  y^2)/s^2] Cos[2 Pi(fx x + fy y)];
sgabor[x_,y_, fx_, fy_, s_] :=
Exp[-(x^2 +  y^2)/s^2] Sin[2 Pi(fx x + fy y)];
```

Various frequencies , vertical orientations, and fixed width

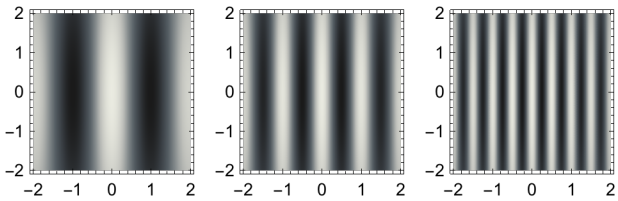
```
vtheta = Table[0, {i1,4}];
vf = {.5,1,2,4};
swidth = {4};
```

```
gfreq = Table[DensityPlot[cgabor[x,y,vf[[i]] Cos[ vtheta[[j]] ], vf[[i]] Sin[ vtheta[[j]]
```

```
gfreq[[1, 1]]
```




```
Show[GraphicsRow[{gfreq[[1, 1, 1]], gfreq[[2, 1, 1]], gfreq[[3, 1, 1]]}]]
```



Various widths , vertical orientations, and fixed center frequency

```
vtheta = Table[0, {i1,4}]; vf = {1};
swidth = {.25,.5,1,2,4};
```

```
gwidth = Table[DensityPlot[cgabor[x,y,vf[[i]] Cos[ vtheta[[j]] ],vf[[i]] Sin[ vtheta[[j]]
```

```
Show[GraphicsRow[{gwidth[[1, 1, 1]], gwidth[[1, 1, 2]], gwidth[[1, 1, 3]]}]]
```

Vertical orientations, and center frequencies of the basis set

```
vtheta = Table[0, {i1,4}];
vf = {.5, 1, 2, 4};
```

Various frequencies, but with the width, s, proportional to the reciprocal of spatial frequency. This maintains a constant bandwidth in octaves.

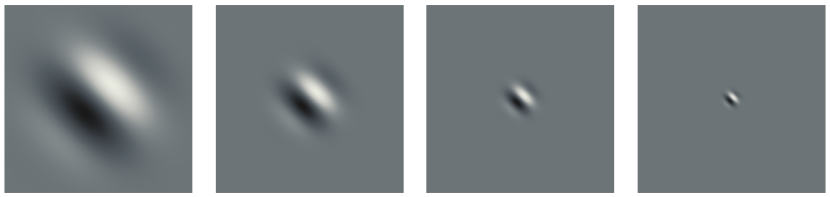
```
vtheta = Table[0, {i1,4}];
vf = {1/2,1,2,4};
swidth = {.25,.5,1,2,4};

gfixedoctave=Table[DensityPlot[cgabor[x,y,vf[[i]] Cos[ vtheta[[j]] ],
        vf[[i]] Sin[ vtheta[[j]] ], .5/vf[[i]] ], {x,-2,2}, {y,-2,2},
        PlotPoints->64, Mesh->False, Frame->False, PlotRange->{-1,1},DisplayFunction→Ident
        {i, 1, 4}, {j, 1, 1}];

Show[GraphicsRow[{gfixedoctave⟦1, 1⟧, gfixedoctave⟦2, 1⟧,
   gfixedoctave⟦3, 1⟧, gfixedoctave⟦4, 1⟧}], DisplayFunction → $DisplayFunction]
```

```
gfixedoctave2=Table[Plot[cgabor[x,0,vf[[i]] Cos[ vtheta[[j]] ],
        vf[[i]] Sin[ vtheta[[j]] ], .5/vf[[i]] ], {x,-2,2},
        PlotPoints->64, Axes->False,PlotRange->{-1,1},DisplayFunction→Identity],
        {i, 1, 4}, {j, 1, 1}];

Show[GraphicsRow[{gfixedoctave2⟦1, 1⟧, gfixedoctave2⟦2, 1⟧, gfixedoctave2⟦3, 1⟧,
   gfixedoctave2⟦4, 1⟧}], DisplayFunction → $DisplayFunction]
```

Various orientations, and center frequencies of the basis set

```
vtheta = Table[i1 Pi/4, {i1,4}];
vf = {.5, 1, 2, 4};

gorientsize = Table[DensityPlot[sgabor[x,y,vf[[i]] Cos[ vtheta[[j]] ],
        vf[[i]] Sin[ vtheta[[j]] ], .5/vf[[i]] ], {x,-2,2}, {y,-2,2},
        PlotPoints->64, Mesh->False,Frame->False,  PlotRange->{-1,1},DisplayFunction→Ident
        {i, 1, 4}, {j, 1, 4}];
```

```
Show[GraphicsRow[
  {gorientsize[[1, 1]], gorientsize[[2, 1]], gorientsize[[3, 1]], gorientsize[[4, 1]]}],
 DisplayFunction → $DisplayFunction]
```



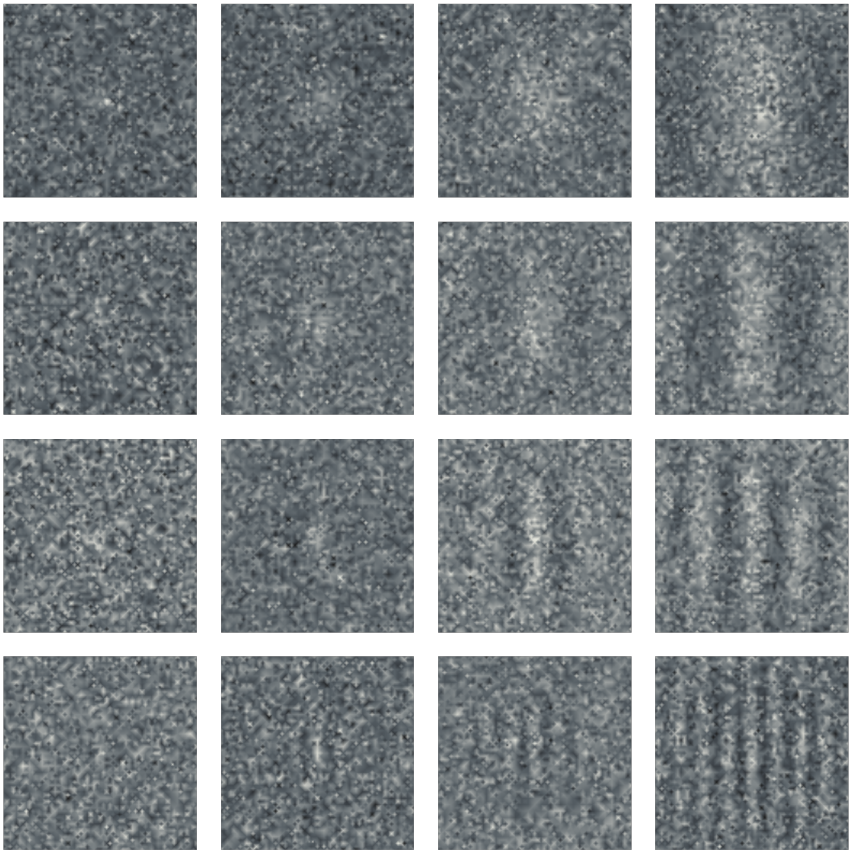
```
Show[GraphicsRow[
  {gorientsize[[1, 2]], gorientsize[[2, 2]], gorientsize[[3, 2]], gorientsize[[4, 2]]}],
 DisplayFunction → $DisplayFunction]
```

```
Show[GraphicsRow[
  {gorientsize[[1, 3]], gorientsize[[2, 3]], gorientsize[[3, 3]], gorientsize[[4, 3]]}],
 DisplayFunction → $DisplayFunction]
```

```
Show[GraphicsRow[
  {gorientsize[[1, 4]], gorientsize[[2, 4]], gorientsize[[3, 4]], gorientsize[[4, 4]]}],
 DisplayFunction → $DisplayFunction]
```

```
vtheta = Table[0, {i1,4}];
(*vf = {1/4,1/2,1,2};*)
vf = {1/4,1/2,1,2};
swidth = {.25,.5,1,2,4};
```

```
gwidthnoise = Table[DensityPlot[.5 * RandomReal[NormalDistribution[0, .4]] +
      .25 * cgabor[x, y, vf[[i]] Cos[ vtheta[[j]] ],
         vf[[i]] Sin[ vtheta[[j]] ], swidth[[k]] ], {x, -2, 2}, {y, -2, 2},
         PlotPoints → 16, Mesh -> False,
     Frame → False,  PlotRange -> {-1, 1}, ImageSize → Tiny],
         {i, 1, 4}, {j, 1, 1}, {k, 1, 4}];

GraphicsGrid[{{gwidthnoise⟦1, 1, 1⟧,
    gwidthnoise⟦1, 1, 2⟧, gwidthnoise⟦1, 1, 3⟧, gwidthnoise⟦1, 1, 4⟧},
   {gwidthnoise⟦2, 1, 1⟧, gwidthnoise⟦2, 1, 2⟧, gwidthnoise⟦2, 1, 3⟧,
    gwidthnoise⟦2, 1, 4⟧}, {gwidthnoise⟦3, 1, 1⟧, gwidthnoise⟦3, 1, 2⟧,
    gwidthnoise⟦3, 1, 3⟧, gwidthnoise⟦3, 1, 4⟧}, {gwidthnoise⟦4, 1, 1⟧,
    gwidthnoise⟦4, 1, 2⟧, gwidthnoise⟦4, 1, 3⟧, gwidthnoise⟦4, 1, 4⟧}}]
```



References

Adelson, E. H., Simoncelli, E., & Hingorani, R. (1987). *Orthogonal Pyramid Transforms for Image Coding.* Paper presented at the Proc. SPIE - Visual Communication & Image Proc. II, Cambridge, MA.

Atick, J. J., & Redlich, A. N. (1992). What does the retina know about natural scenes? *Neural Computation*, *4*(2), 196–210.

Burgess, A. E., Wagner, R. F., Jennings, R. J., & Barlow, H. B. (1981). Efficiency of human visual signal

discrimination. *Science, 214*, 93-94.
Daugman, J. G. (1988). An information-theoretic view of analog representation in striate cortex, *Computational Neuroscience*. Cambridge, Massachusetts: M.I.T. Press.
De Valois, R. L., Albrecht, D. G., & Thorell, L. G. (1982). Spatial frequency selectivity of cells in macaque visual cortex. *Vision Res, 22*(5), 545-559.
De Valois, R. L., Yund, E. W., & Hepler, N. (1982). The orientation and direction selectivity of cells in macaque visual cortex. *Vision Res, 22*(5), 531-544.
Enroth-Cugell, C., & Robson, J. G. (1966). The contrast sensitivity of retinal ganglion cells of the cat. Journal of Physiology (London), 187, 517-552.
Hubbard, B. B. (1998). *The world according to wavelets : the story of a mathematical technique in the making* (2nd ed.). Wellesley, Mass: A.K. Peters.
Hubel, D. H., & Wiesel, T. N. (1959). Receptive Fields of Single Neurons in the Cat's Striate Cortex. *J. Physiol., 148*, 574-591.
Hubel, D. H., & Wiesel, T. N. (1968). Receptive Fields and Functional Architecture of Monkey Striate Cortex. *J. Physiol.*, 215-243.
Gerstein, G. L., & Mandelbrot, B. (1964). Random walk models for the spike activity of a single neuron. Biophysical Journal, 4, 41-68.
Kersten, D. (1984). Spatial summation in visual noise. *Vision Research, 24*, 1977-1990.
Morgenstern, Y., & Elder, J. H. (2012). Local Visual Energy Mechanisms Revealed by Detection of Global Patterns. *Journal of Neuroscience*, *32*(11), 3679–3696.
Shapley, R., & Perry, H. H. (1986). Cat and monkey retinal ganglion cells and their visual functional roles. Trends in Neuroscience, 9(5), 229-235.
Silverman, M. S., Grosof, D. H., DeValois, R. L., & Elfar, S. D. (1989). Spatial-frequency organization in primate striate cortex*., 86*, 711-715.
Silverman, M. S., Grosof, D. H., De Valois, R. L., & Elfar, S. D. (1989). Spatial-frequency organization in primate striate cortex. *Proc Natl Acad Sci U S A, 86*(2), 711-715.
Simoncelli, E. P., Freeman, W. T., Adelson, E. H., & Heeger, D. J. (1992). Shiftable Multi-scale Transforms. *IEEE Trans. Information Theory, 38*(2), 587--607.
Sowden, P. T., & Schyns, P. G. (2006). Channel surfing in the visual brain. *Trends in Cognitive Sciences*, *10*(12), 538–545.
Watson, A. B., Barlow, H. B., & Robson, J. G. (1983). What does the eye see best? *Nature, 31,*, 419-422.
Watson, A. B. (1987). Efficiency of a model human image code. *Journal of the Optical Society of America, A, 4*(12), 2401-2417.
Watson, Andrew B. (2000) Visual detection of spatial contrast patterns: Evaluation of five simple models (2000), Optics Express 6(1), 12-33 (http://www.opticsexpress.org/oearchive/source/14103.htm).