Computational Vision
U. Minn. Psy 5036
Daniel Kersten
Lecture 19: Motion Illusions & Bayesian models

Initialize

Spell check off

In[18]:=
```
Off[General::spell1];
```

In[19]:=
```
SetOptions[ArrayPlot, ColorFunction → "GrayTones",
  DataReversed → True, Frame → False, AspectRatio → Automatic,
  Mesh → False, PixelConstrained → True, ImageSize → Small];
SetOptions[ListPlot, ImageSize → Small];
SetOptions[Plot, ImageSize → Small];
SetOptions[DensityPlot, ImageSize → Small, ColorFunction → GrayLevel];
nbinfo = NotebookInformation[EvaluationNotebook[]];
dir = ("FileName" /. nbinfo /. FrontEnd`FileName[d_List, nam_, ___] :→ ToFileName[d]);
```

Outline

Last time

- Early motion measurement--types of models
- Functional goals of motion measurements
- Optic flow
  - Cost function (or energy) descent model
  - A posteriori and a priori constraints
  - Gradient descent algorithms
  - Computer vs. human vision and optic flow
    - -- area vs. contour

Today

Local measurements, neural systems & orientation in space-time

Representing motion, Orientation in space-time
- Fourier representation and sampling
- Optic flow, the gradient constraint, aperture problem

Neural systems solutions to the problem of motion measurement.
- Space-time oriented receptive fields

Motion phenomena & illusions

Neither the area-based nor the contour-based algorithms we've seen can account for the range of human motion phenomena or psychophysical data that we now have.
Look at human motion perception

Global integration

Sketch a Bayesian formulation--the integrating uncertain local measurements with the right priors can be used to model a variety of human motion results.

Orientation in space-time: Relating neuron responses to gradient-based motion models

In this section, we'll see how viewing motion measurement as detecting orientation in space-time is related to neurophysiological theories of neural motion selectivity.

Demo: area-based vs. contour-based models

Are the representation, constraints, and algorithm a good model of human motion perception?

The answer seems to be "no" in general. The representation of the input is probably wrong. Human observers often give more weight to contour movement than to intensity flow. Human perception of the sequence illustrated below differs from "area-based" models of optic flow such as the above Horn and Schunck algorithm. The two curves below would give a maximum correlation at zero--hence zero predicted velocity. Human observers see the contour move from left to right--because the contours are stronger features than the gray-levels. However we will see in Adelson's missing fundamental illusion that the story is not as simple as a mere "tracking of edges" --and we will return to spatial frequency channels to account for the human visual system's motion measurements. At the end of this lecture, we'll review a Bayesian model that integrates local motion information according to reliability, providing a theory that may explain a diverse set of motion illusions.

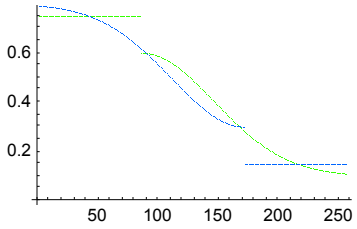
In[1]:=
```
size = 120;
Clear[y];
low = 0.2; hi = .75;
y[x_] := hi /; x<1
y[x_] := .5 Exp[-(x-1)^2]+.1 /; x >= 1
```

In[6]:=
```
ylist = Table[y[i],{i,0,3,3/255.}];
width = Dimensions[ylist][[1]];
```

In[8]:=
```
picture1 = Table[ylist,{i,1,width/2}];
picture2 = .9 - Transpose[Reverse[Transpose[picture1]]];
```

In[10]:=
```
g1 = ListPlot[picture1[[size/2]],PlotStyle->{Hue[.3]}];
g2 = ListPlot[picture2[[size/2]],PlotStyle->{Hue[.6]}];
Show[g1,g2]
```

Out[12]=



In[13]:=
```
ga1=ArrayPlot[picture1,Frame->False,Mesh->False,
                PlotRange->{0,1}, AspectRatio->Automatic];
ga2=ArrayPlot[picture2,Frame->False,Mesh->False,
                PlotRange->{0,1}, AspectRatio->Automatic];
```

In[15]:=
```
ListAnimate[{Show[ga1], Show[ga2], Show[ga1]}, 2,
 Paneled → False, AppearanceElements → None, AnimationRunning → False]
```

Out[15]=

There is a clear sense of motion of the edge, even though the signal inferred from an intensity, region-based integration of optic flow would produce little or no optic flow in that direction.

Spatial frequency dependence: Adelson's missing fundamental motion illusion

We first make a square-wave grating.

In[24]:=
```
realsquare[x_,y_,phase_] := Sign[Sin[x + phase]];
```

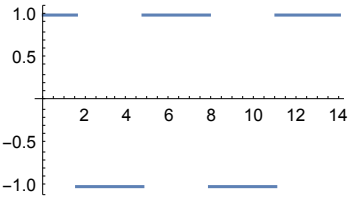
And make a four-frame movie in which the grating gets progressively shifted LEFT in steps of Pi/2. That is we shift the grating left in 90 degree steps.

In[25]:=
```
Table[DensityPlot[realsquare[x,y,i Pi/2],
    {x,0,14},{y,0,1}, Frame->False,
        Mesh->False,PlotPoints -> 60, Axes->None, PlotRange->{-4,4},ImageSize->Tiny],{i,1,
]
```

Out[25]=

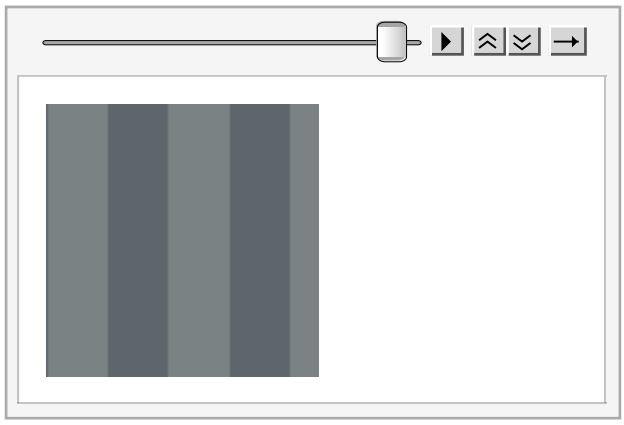
In[26]:=
```
Plot[realsquare[x, .5, Pi/2], {x, 0, 14}]
```

Out[26]=



In[27]:=
```
gsq = Table[
  ArrayPlot[Table[realsquare[x, y, i Pi/2], {y, 0, 14, .1}, {x, 0, 14, .1}],
   PlotRange → {-8, 8}, Frame -> False, ColorFunction -> "GrayTones",
        Mesh -> False, Axes -> None], {i, 1, 4, 1}
]; ListAnimate[gsq, AnimationRunning → False]
```

Out[27]=



A square wave can be decomposed into its Fourier components as:

realsquare(x) = (4/π)*{sin(x) + 1/3 sin(3x) + 1/5 sin(5x) + 1/7 sin(7x) + ...}

Now subtract out the fundamental frequency from the square wave

...leaving (4/π)*{1/3 sin(3x) + 1/5 sin(5x) + 1/7 sin(7x) + ...}

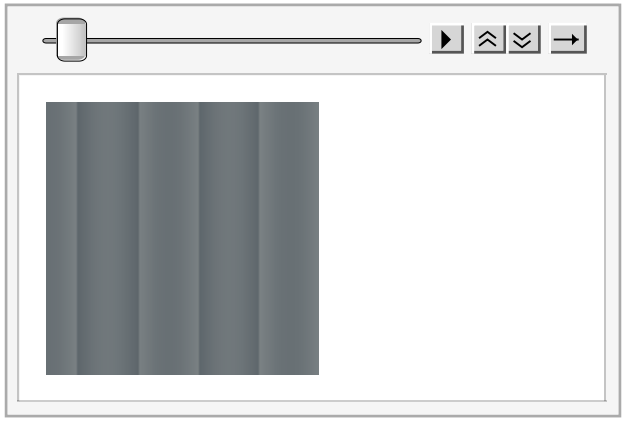
In[28]:=
```
realmissingfundamental[x_,y_,phase_] := realsquare[x,y,phase] - (4.0 / Pi) Sin[x + phase];
```

And make another four-frame movie in which the missing fundamental grating gets progressively shifted LEFT in steps of Pi/2. That is we shift the grating left in 90 degree steps, 1/4 th of a full cycle of 360 degrees.

It is well-known that a low contrast square wave with a missing fundamental appears similar to the square wave (with the fundamental). (There is a pitch analogy in audition, although the analogy breaks down at the level of mechanism.) One reason is that we are more sensitive to sharp than gradual changes in intensity. If you look at the luminance profile with the missing fundamental, you would probably guess that the perceived motion for this sequence would appear to move to the left, as before. After all, it looks pretty much like the square wave. But it doesn't. Surprisingly, the missing fundamental wave appears to move to the right!

In[29]:=
```
gsqr = Table[
  ArrayPlot[
   Table[realmissingfundamental[x, y, i Pi / 2], {y, 0, 14, .1}, {x, 0, 14, .1}],
   PlotRange → {-8, 8}, Frame -> False, ColorFunction -> "GrayTones",
        Mesh -> False, Axes -> None], {i, 1, 4, 1}
]; ListAnimate[gsqr, AnimationRunning → False]
```

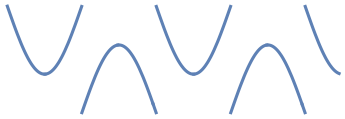
Out[29]=



Play the above movie. It typically appears to be moving to the right. You can generate movies with different contrasts by adjusting the PlotRange parameters.

In[30]:=
```
Plot[realmissingfundamental[x, 1, 0], {x, 0, 14}, PlotRange → {-2, 2}, Axes → False]
```

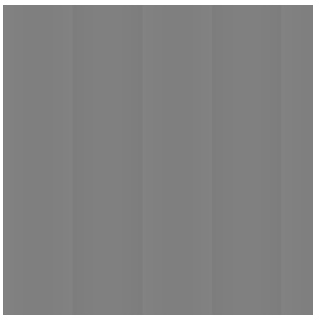
Out[30]=



Or you can change the contrast with the multiplier below. E.g. if the multiplier is: .01:

In[33]:=
```
Image[
 Table[.01 * (realmissingfundamental[y, x, 0]) + .5, {x, 0, 14, .03}, {y, 0, 14, .03}]]
```

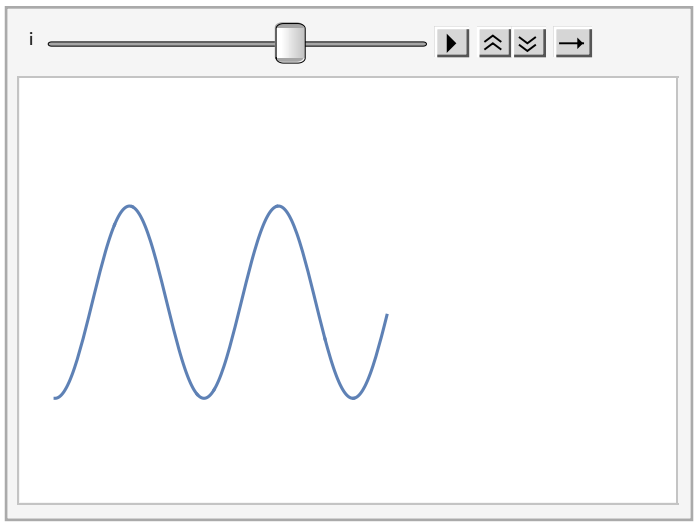
Out[33]=



If the visual system reconstructed the missing fundamental before computing motion direction we'd expect to see motion to the left. The missing fundamental moves towards the left as you can see by playing the movie below.

In[34]:=
```
Animate[Plot[Sin[x + i Pi/2],{x,0,14},
    PlotPoints -> 60, Axes->None,ImageSize->Small],{i,1,4,1},AnimationRunning->False]
```

Out[34]=



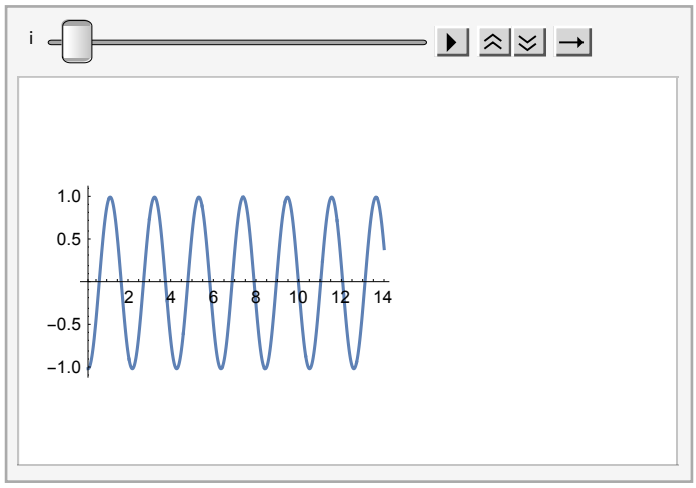
What in the stimulus does move to the right?

Why does the movement appear to be going to the right? Probably the best explanation comes from looking at the dominant frequency component in the pattern, which is the 3rd harmonic. It turns out that the third harmonic is jumping in 1/4 cycle steps to the right, even though the pattern as a whole is jumping in 1/4 cycle steps (relative to the missing fundamental) to the left, as shown in the figure below:

Make a movie with Plot[ ] that shows the third harmonic. Which way does it move?

In[36]:=
```
Animate[
    Plot[Sin[3 (x + i Pi/2)],{x,0,14},PlotPoints -> 60,ImageSize->Small],{i,1,4,1},Animati
]
```

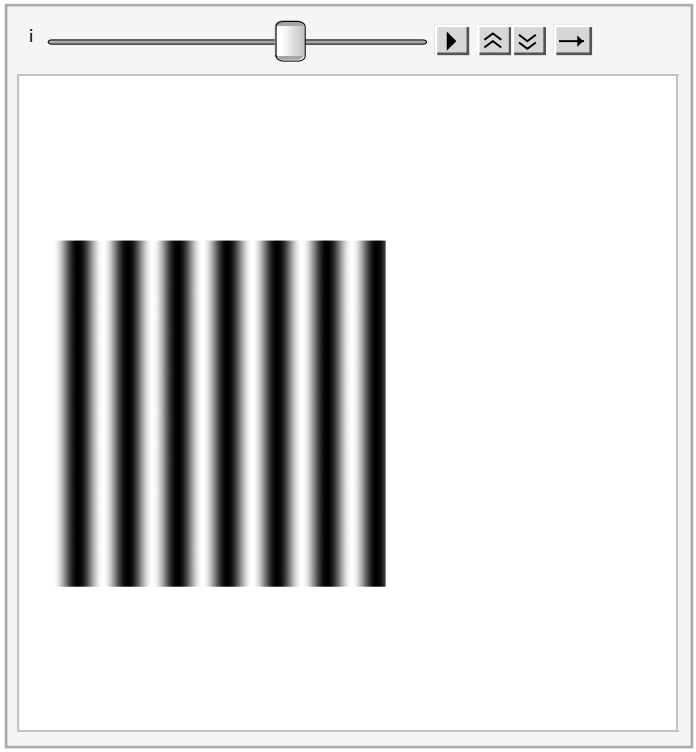
Out[36]=



And here is the movie with just the third harmonic. Which way does it appear to move?

In[38]:=
```
Animate[
    DensityPlot[Sin[3 (x + i Pi/2)],
        {x,0,14},{y,0,1}, Frame->False,
        Mesh->False,PlotPoints -> 60, Axes->None, PlotRange->{-4,4}],{i,1,4,1},AnimationRu
]
```

Out[38]=



Adelson's demonstration implies that human motion measurement mechanisms are tuned to spatial frequency. How can the inferred biological mechanisms be pieced together to compute optic flow? We can construct the following rough outline. Assume we have, at each spatial location, a collection of filters tuned to various orientations (q) and speeds (s) over a local region. (For an algorithm for optic flow based on biologically plausible spatiotemporal filters see Heeger, 1987).

In this scheme, the optic flow measurements are distributed across the units, so if we wanted to read off the velocity from the pattern of activity, we would need some additional processing. For example, the optic flow components could be represented by the "centers of mass", measured by a "population vector" across the distributed activity. Because these measurements are local, we still have the aperture problem. We will look at possible biological solutions to this problem later.

A problem with this simple interpretation is that many V1 cells are known to be tuned to spatial and temporal frequency in such a way that the spatio-temporal filter is the product of the space and time filters. This means that there is a favored temporal frequency that is the same across spatial frequencies, so the filter will be tuned to different speeds depending on the spatial frequency.

Representing motion as orientation in space-time

Mathematica demo

In[39]:=
```
size = 32; x0 = 4; y0 = 4; pw = 12; xoffset = 1;
A1 = Table[Random[], {size}, {size}]; (*A2 = A1;*)
A2 = Table[Random[], {size}, {size}];
A2[[Range[y0, y0 + pw], Range[x0, x0 + pw]]] =
  A1[[Range[y0, y0 + pw], Range[x0 - xoffset, x0 + pw - xoffset]]];
grap1 = ArrayPlot[A1, Mesh → False];
grap2 = ArrayPlot[A2, Mesh → False];
```

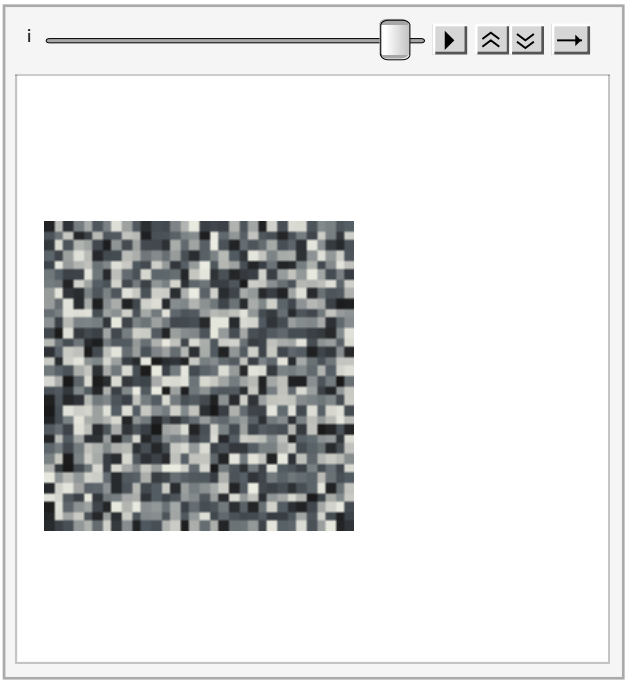
In[45]:=
```
ListAnimate[{grap1, grap2, grap1, grap2}, 4, AnimationRunning → False]
```

Out[45]=

Despite the dynamic background noise, your visual system can pick up on a regularity in the movie--a sub-group of pixels has a consistent displacement of luminance values across space and time. We can visualize this regularity by making a plot of intensity as a function of time and space, an "t-x" plot. Let's visualize it for the 8th row of an 8-frame noise image sequence, where the central square is moving from left to right, and the background is fixed.

In[53]:=
```
Animate[
          A2[[Range[y0, y0 + pw], Range[x0, x0 + pw]]] =
  A1[[Range[y0, y0 + pw], Range[x0 + i, x0 + pw + i]]];
     ArrayPlot[A2], {i, 8, 1, 1}]
```

Out[53]=

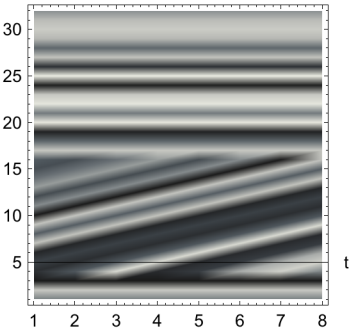


(Note in this case, the motion boundaries are consistent with motion seen through a “hole”, rather than of a square moving above the background. This distinction isn’t important to the key idea of motion as orientation in space-time, but does become important when considering motion parallax as a depth cue.)

In[55]:=
```
nframes = 8;
xt = {};
For[i=nframes,i>0,i--,
    A2[[Range[y0,y0+pw],Range[x0,x0+pw]]] =
    A1[[Range[y0,y0+pw],Range[x0+i,x0+pw+i]]];
    xt = Join[xt,{A2[[8]]}]
];
```

In[58]:=
```
ListDensityPlot[Transpose[xt], Mesh->False, Axes->True, AxesLabel->{"t","x"},ImageSize->Sm
```

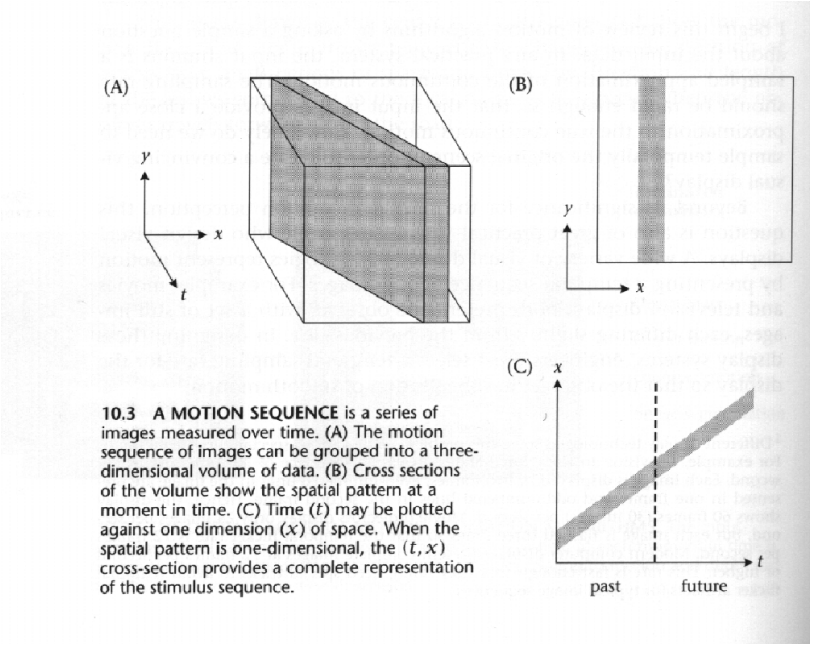
Out[58]=



The luminance of pixels from rows 4 to 16 move to the right resulting in a positively oriented intensity pattern in t-x space.

x-y-t space

In general, patterns are 2-D, so a space-time representation is three dimensional. The diagram below (from Wandell), shows a a representation for a bar moving from right to left in the x direction.

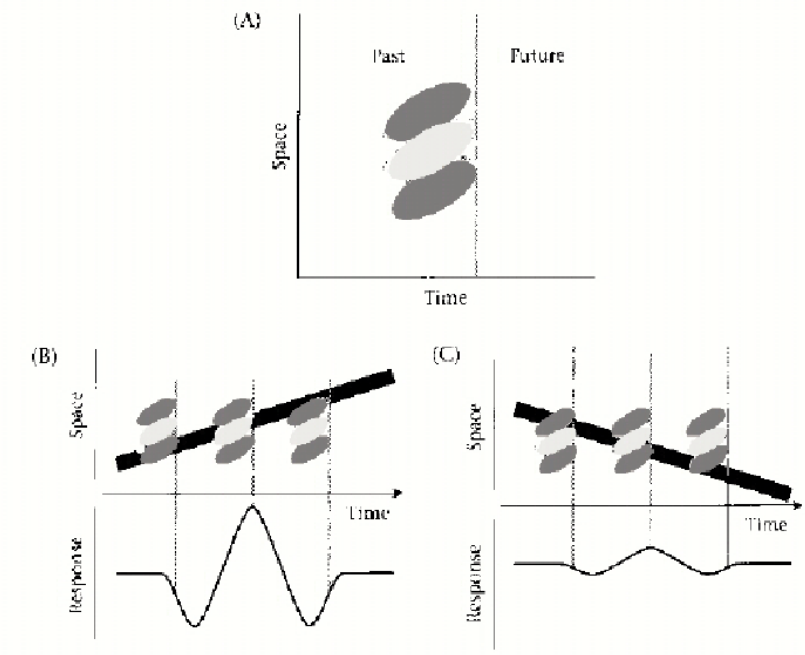


10.3 A MOTION SEQUENCE is a series of images measured over time. (A) The motion sequence of images can be grouped into a three-dimensional volume of data. (B) Cross sections of the volume show the spatial pattern at a moment in time. (C) Time (t) may be plotted against one dimension (x) of space. When the spatial pattern is one-dimensional, the (t, x) cross-section provides a complete representation of the stimulus sequence.

Neurophysiological filters

How might one construct spatio-temporal filters to estimate orientation in space-time?

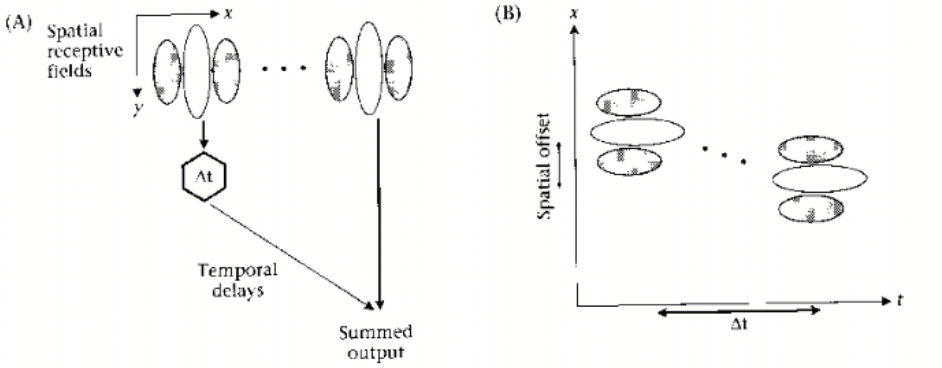
Space-time filters for detecting orientation in space-time



10.6 SPACE–TIME-ORIENTED RECEPTIVE FIELD. (A) The space–time receptive field of a neuron is represented on a (t, x) plot. The neuron always responds to events in the recent past, so the receptive field moves along the time axis with the present. The dark areas show an inhibitory region, and the light area shows an excitatory region. (B) The upper portion of the graph shows a (t, x) plot of a moving line and the space–time receptive field of a linear neuron. The neuron's receptive field is shown at several different moments in time, indicated by the vertical dashed lines. The common orientation of the space–time receptive field and the stimulus motion produce a large amplitude response, shown in the bottom half of the graph. (C) When the same neuron is stimulated by a line moving in a different direction, the stimulus motion aligns poorly with the space–time receptive field. Consequently, the response amplitude is much smaller.

From Wandell, "Foundations of Vision", 1995

A possible mechansim for building space-time filters from two spatial filters with a temporal delay



10.7 A METHOD FOR CREATING A SPACE–TIME-ORIENTED RECEPTIVE FIELD. (A) A pair of spatial receptive fields, displaced in the x direction, is shown at the top. The response of the neuron on the left is delayed and then added to the response of the neuron on the right. (B) The (t, x) receptive field of the output neuron in panel (A). The temporal response of the neuron on the left is delayed compared to the temporal response of the neuron on the right. The combination of spatial displacement and temporal delay yields an output neuron whose receptive field is oriented in space–time.

In panel A above, two matched oriented x-y filters pick out preferred spatial frequency components, one of the outputs is delayed relative to the other, and then the outputs are combined through summation. The summed output would show maximum firing rate for that particular spatial frequency, spatial offset between filters, and temporal delay. Panel B shows the interpretation of such a receptive field (as measured in the summed ouput) in space-time.

Wandell, "Foundations of Vision", 1995

Relationship of the gradient constraint -- the “optic flow equation”-- to oriented space-time filters

Let's pursue the analogy of edge detection in space to "edge detection" in space-time.

Recall the gradient constraint:

$$v_x \frac{\partial L}{\partial x} + v_y \frac{\partial L}{\partial y} + \frac{\partial L}{\partial t} = 0$$

v_x and v_y correspond to u and v used in the previous lecture. For simplicity, let’s consider just one spatial dimension x, and thus (t,x) space.

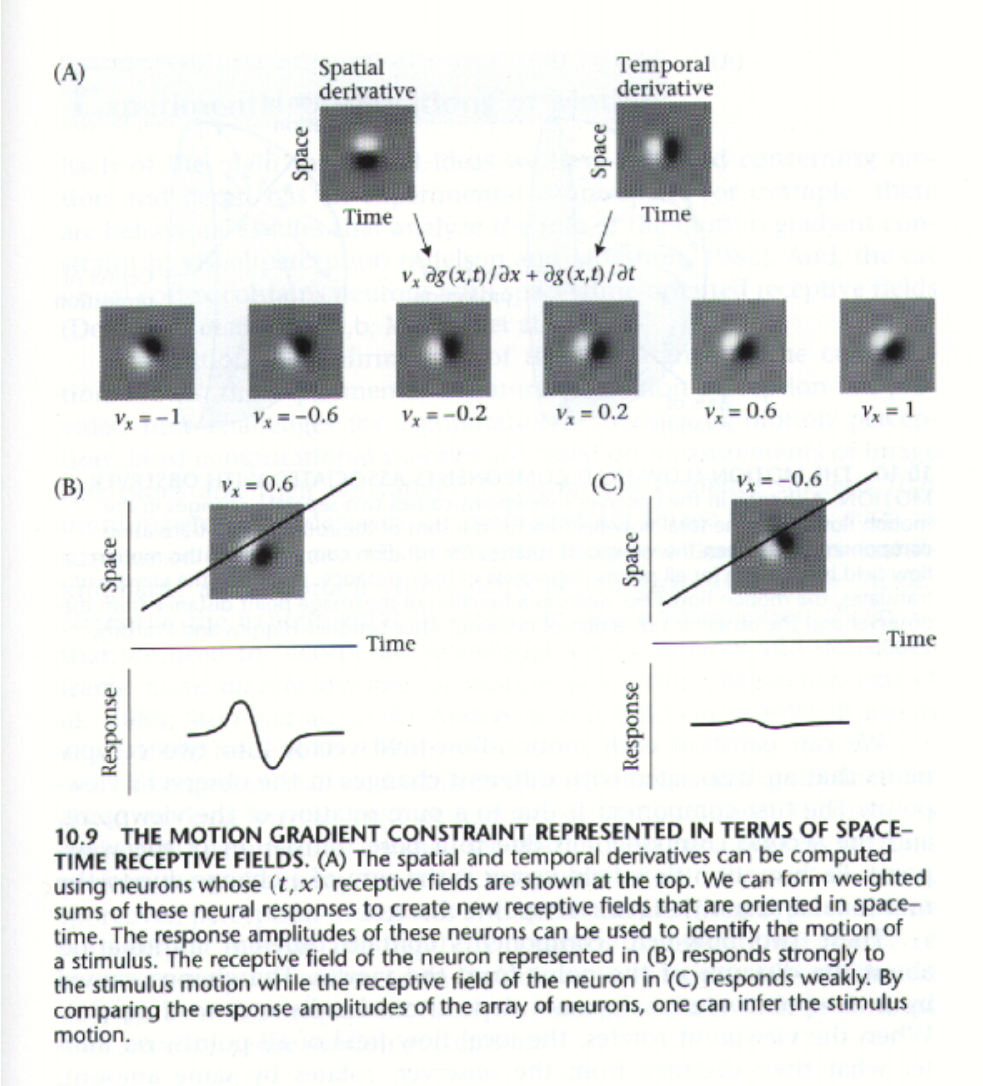
$$v_x \frac{\partial L}{\partial x} + \frac{\partial L}{\partial t} = 0$$

As we saw for edge detection, blurring reduces the effect of noise. So let image L(x,y,t) be blurred in

space and smeared in time, g(x,y,t).

$$v_x \frac{\partial g}{\partial x} + \frac{\partial g}{\partial t} = 0$$

The figure below from Wandell shows how adding weighted combinations of spatial and temporal derivatives, one can produce
a family of space-time oriented filters whose orientation is given by the speed v_x:



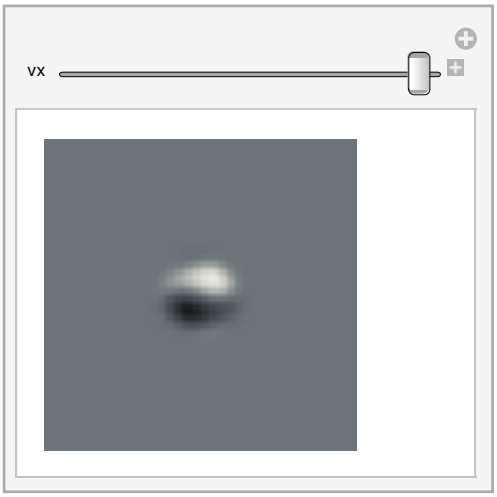
10.9 THE MOTION GRADIENT CONSTRAINT REPRESENTED IN TERMS OF SPACE–TIME RECEPTIVE FIELDS. (A) The spatial and temporal derivatives can be computed using neurons whose (t, x) receptive fields are shown at the top. We can form weighted sums of these neural responses to create new receptive fields that are oriented in space–time. The response amplitudes of these neurons can be used to identify the motion of a stimulus. The receptive field of the neuron represented in (B) responds strongly to the stimulus motion while the receptive field of the neuron in (C) responds weakly. By comparing the response amplitudes of the array of neurons, one can infer the stimulus motion.

The bottom panels B and C in Figure 10.9 from Wandell show how the response as a function of time has a greater amplitude for the motion filter that matches the pattern in t-x space.

Here's a bit of *Mathematica* that illustrates how v_x rotates the filter in space-time.

In[109]:=
```
grating[x_, t_, fx_, ft_, ϕ_, σ_] := Exp[-((x^2 + t^2) / σ)^2] * Sin[2 π (fx x + ft t) + ϕ];
fx = 1; ft = 0; ϕ1 = 0; σ = .20;

dgdx = Table[grating[x, t, fx, ft, ϕ1, σ], {x, -2, 2, .05}, {t, -2, 2, .05}];
fx = 0; ft = 1; ϕ1 = 0; σ = .20;
dgdt = Table[grating[x, t, fx, ft, ϕ1, σ], {x, -2, 2, .05}, {t, -2, 2, .05}];
Manipulate[ArrayPlot[vx * dgdx + dgdt], {vx, 0, Pi}]
```

Out[114]=



For the paper book, see:

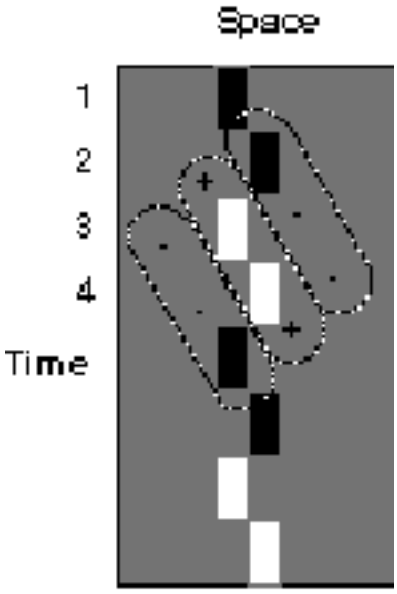
http://www.amazon.com/Foundations-Vision-Brian-Wandell/dp/0878938532/ref=sr_1_1?ie=UTF8&s=-books&qid=1226337031&sr=1-1

For the online material, see:

https://foundationsofvision.stanford.edu/chapter-10-motion-and-depth/

From theory to demos

If motion is measured using oriented excitatory-inhibitory space-time filters, one might be able to generate a continuous sense of motion by cyclying through a set of frames designed to excite these filters. See: http://www.georgemather.com



In[59]:=
```
mov2 = {
```




```
};
```

In[60]:=
```
ListAnimate[mov2, 15]
```

Out[60]=

- Try producing a sequence of frames by convolving a single input image with a oriented spatial (x-y) gabor function filter (with inhibitory side-lobes), then produce the next frame by phase-shifted from the filter used in the previous frame, and so forth. Try generating a sequence of 8 frames. (See Heeger, SIGGRAPH 1991).

Human motion perception: Rhombi and plaids

Aperture effects

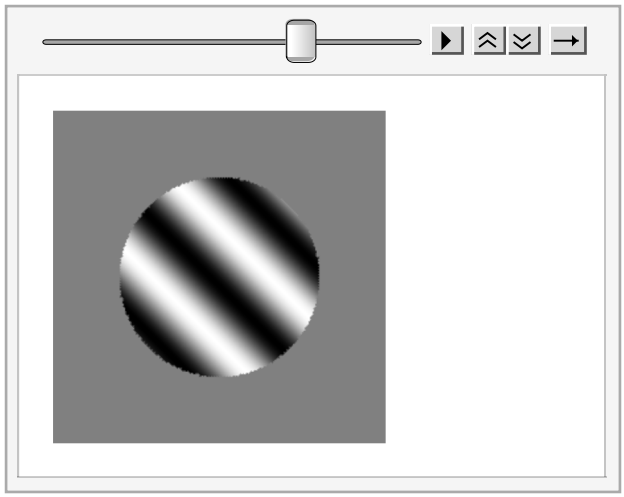
In[61]:=
```
niter = 8; width = 32;
theta1 = Pi/4.; contrast1 = 0.5;
freq1 = 4.; period1 = 1/freq1;
stepx1 = Cos[theta1]*(period1/niter); stepy1 = Sin[theta1]*(period1/niter);

grating[x_,y_,freq_,theta_] := Cos[(2. Pi freq)*(Cos[theta]*x + Sin[theta]*y)];
```

Circular aperture

In[67]:=
```
iseq=Table[DensityPlot[If[(x-0.5)^2+(y-0.5)^2<0.3^2,grating[x+i*stepx1,y+i*stepy1,freq1,th
];
```

In[68]:=
```
ListAnimate[iseq]
```

Out[68]=

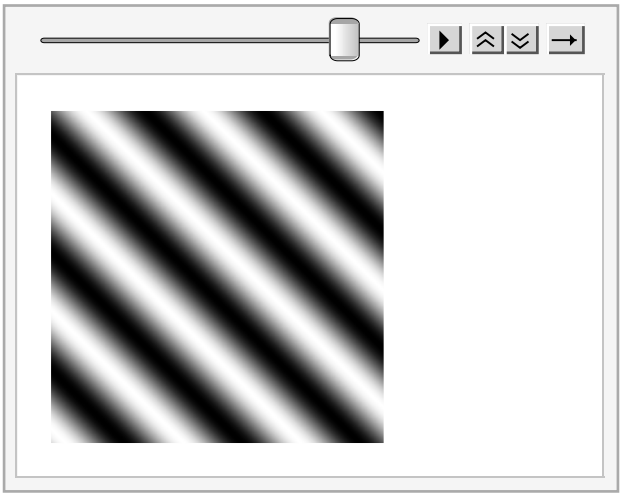


Square aperture

In[69]:=
```
iseqs=Table[DensityPlot[grating[x+i*stepx1,y+i*stepy1,freq1,theta1],{x,0,1},{y,0,1},PlotPo
];
```

In[70]:=
```
ListAnimate[iseqs, DisplayAllSteps → True, AnimationRate → 10]
```

Out[70]=

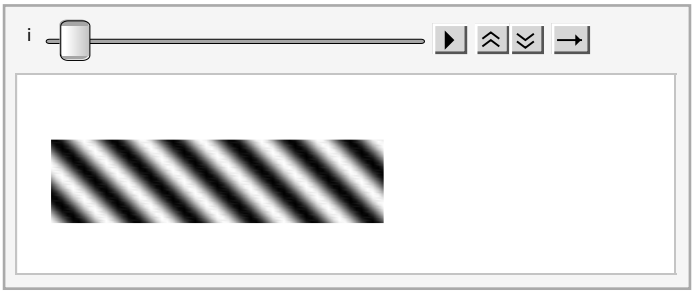


What do you see if you attend to the vertical boundaries? The horizontal boundaries?

Rectangular horizontal aperture

In[90]:=
```
Animate[DensityPlot[grating[x+i*stepx1,y+i*stepy1,freq1,theta1],{x,0,1},{y,0,.25},
        Mesh->False,Frame->None,PlotRange->{-2,2},PlotPoints->width,
        AspectRatio->Automatic],{i,1,niter},AnimationRunning->False]
```

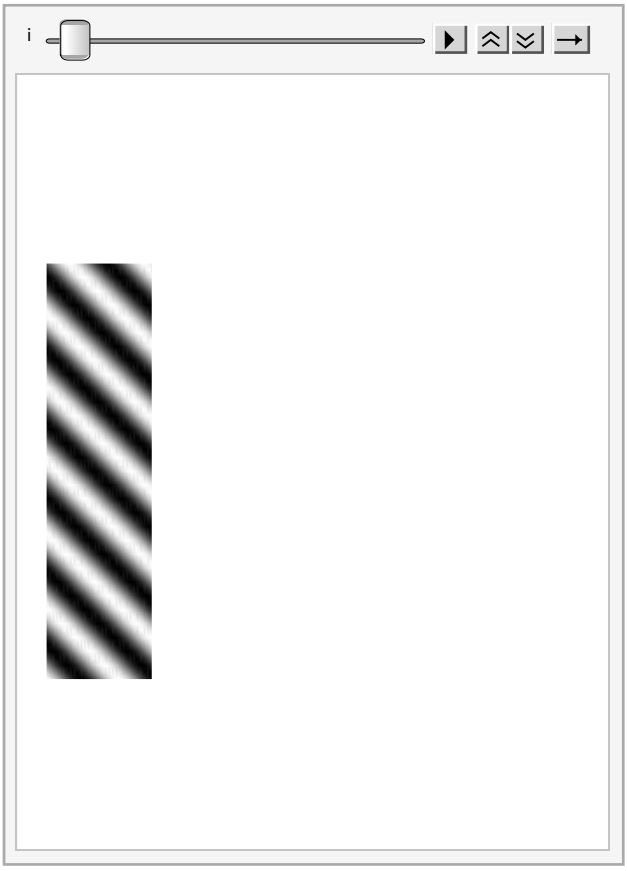
Out[90]=



Rectangular vertical aperture

In[72]:=
```
Animate[
    DensityPlot[grating[x+i*stepx1,y+i*stepy1,freq1,theta1],{x,0,.25},{y,0,1},
        Mesh->False,Frame->None,PlotRange->{-2,2},PlotPoints->width,
        AspectRatio->Automatic],{i,1,niter},AnimationRunning->False]
```

Out[72]=



The main point is that spatial integration of motion information takes into account flow at the boundaries. E.g. if boundary features at the vertical edge outweigh contributions from those at the horizontal edge, there will be a vertical bias, and so forth.

- Project idea: Try the above with stereo-defined apertures. Does stereo de-couple the integration of the boundary cues with the internal cues?

Moving rhombus illusions

http://www.cs.huji.ac.il/~yweiss/Rhombus/rhombus.html

Motion Plaids

Two overlapping (additive transparent) sinusoids at different orientations and moving in different directions are, under certain conditions seen as a single pattern moving with a velocity consistent with an intersection of constraints. Under other conditions, the two individual component motions are seen.

Adding two gratings, single frame

Plaid grating: Grating 1 + Grating 2

In[91]:=
```
GraphicsRow[{Image[ ], "+", Image[ ], "=", Image[ ]}]
```

Out[91]=

Trying using Psychopy's python demo via the menu: Demos/stimuli/plaid.py

Initialize parameters

Alternatively, play with the *Mathematica* code below:

In[92]:=
```
niter = 32; width = 16;
theta1 = Pi/4.; contrast1 = 0.25; theta2 = -1.2*Pi/2.; contrast2 = 0.75;

theta1 = Pi/4.; contrast1 = 0.75; theta2 = -Pi/4.; contrast2 = 0.75;

freq1 = 8.; period1 = 1/freq1; freq2 = 8.; period2 = 1/freq2;

stepx1 = Cos[theta1]*(period1/niter); stepy1 = Sin[theta1]*(period1/niter);

stepx2 = Cos[theta2]*(period2/niter); stepy2 = Sin[theta2]*(period2/niter);
(*stepx = Min[stepx1,stepx2]; stepy = Min[stepy1,stepy2];*)

grating2[x_,y_,freq_,theta_,contrast_] := .25*contrast*Cos[(2. Pi freq)*(Cos[theta]*x + Si

plaid = Table[
    Image[Table[grating2[x+i*stepx1,y+i*stepy1,freq1,theta1,contrast1]+
    grating2[x+i*stepx2,y+i*stepy2,freq2,theta2,contrast2],{x,0,1,.01},{y,0,1,.01}]],{i,1,
```

In[108]:=
```
ListAnimate[plaid, 15, DisplayAllSteps → True, AnimationRunning → False]
```

Out[108]=

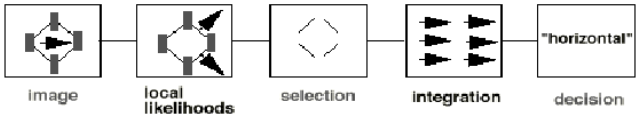
- Now try the above motion plaid with different motion directions and contrasts

There is a substantial history of research into understanding how both human and neural mechanisms integrate motion information in plaids. Movshon et al. (1986) provides perhaps the earliest examples of an interdisciplinary approach combining computational theory, perceptual observations, and neuron

responses (in MT and V1) to understand motion processing.

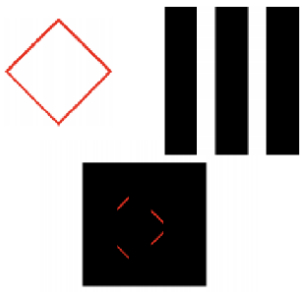
Bayesian model for integrating local motion measurements

General problem



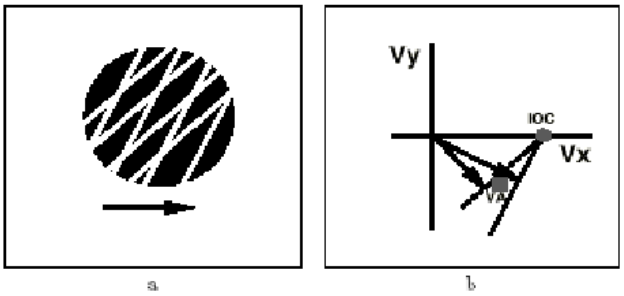
The above figure illustrates the problems of selection and integration. Below we assume the selection problem has been solved, leaving the question of how to integration the local ambiguous measurements into a single velocity vector for the group.

Lorenceau & Shiffrar's demo:



Intersection of constraints revisited

Grating plaids sometime seen as coherent, other times as two overlapping transparent gratings moving separately.



Further, the intersection of constraints model doesn't always predict perception. Sometimes the perceived motion is in the direction of the average of the two aperture vectors.

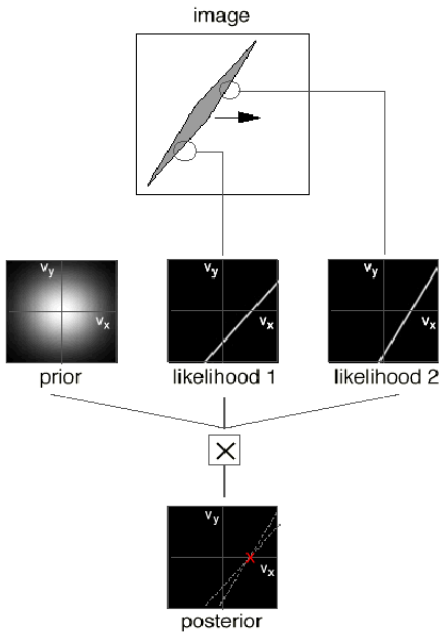
Both intersection of constraints and vector average models have been proposed to account for motion perception in different regimes, i.e. different combinations of plaid component parameters. The next section shows how a number of motion effects, including plaids and the above moving rhombus percepts can be explained by one model.

Bayes model for integration

Yuille, A., & Grzywacz, N. (1988).

Following Weiss Y, Simoncelli EP, Adelson EH (2002) (Motion illusions as optimal percepts. Nat Neurosci 5:598-604), we can get a graphical view of how local information can be combined probabilistically.

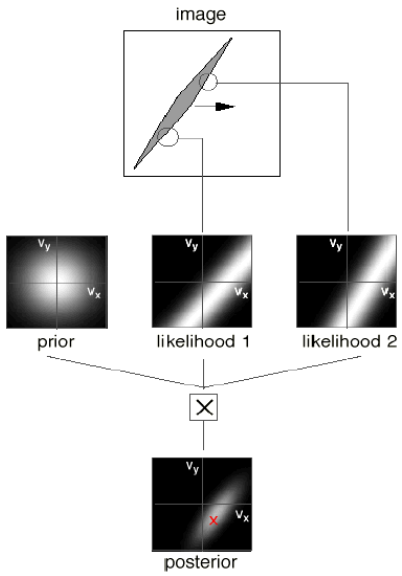
Probabilistic interpretation of intersection of constraints



The plot illustrates the calculation of the posterior:

$p(v_x,v_y|$ perpendicular component1, perpendicular component 2) $\propto$ p(perpendicular component1 | v_x)p(perpendicular component 2 | v_x) $p(v_x,v_y)$

Probabilistic interpretation with noisy measurements



Key ideas

A. Information for motion direction and speed comes from two types of sources: 1) the data, which involves many measurements of local velocity that produce a likelihood of (v_x v_y) for each measurement. (Two are shown above). These likelihoods have various degrees of uncertainty (i.e. variance in the possible values of (v_x v_y)) that depend on image signal-to-noise ratio, e.g. contrast. 2) Prior assumptions that assume the speeds are slow, i.e. a probability distribution of (v_x v_y) with a mean of zero.

B. A Bayes optimal solution multiplies the prior and likelihoods to obtain the posterior. The important qualitative idea is that estimates based on this posterior effectively weight information from the data and prior according to reliability. So if there is more certainty in the measurements (e.g. high contrast), this will bias the estimates of (v_x v_y) towards the intersection of constraints and away from the prior. If it is hard to see the motion, then the estimates should be biased towards the prior, i.e. slower.

- Exercise: Assume Gaussian distributions and prove that the maximum a posteriori estimate of a parameter given two measurements is given by:

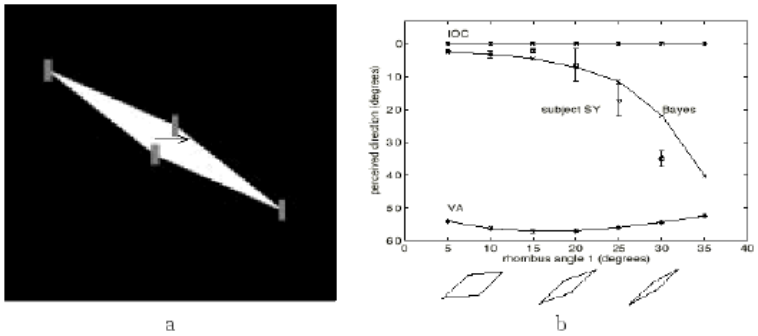
$$v = v_1 r_1/(r_1 + r_2) + v_2 r_2/(r_1 + r_2)$$

Where v_1 and v_2 are the best estimates based on measurements 1 and 2 separately, and $r_i = 1/\sigma_i^2$, *i.e.* the reciprocal of the variance of each. The math for this is identical to that for cue integration (*e.g.*, see Lecture 6, and Ernst, *M.O.*, & Banks, *M.S.*(2002).Humans integrate visual and haptic information in *a* statistically optimal fashion.Nature, 415 (6870), 429 – 433.)

Tests of theory

Rhombus experiment

The above figure shows how as the rhombus gets skinnier, the peak of the posterior moves towards the lower right quadrant of velocity space, consistent with psychophysics.



Aperture effects

Imagine a corrugated surface moving up, but viewed through apertures. For a circular aperture, information around the boundary is symmetric, so the bias for a certain direction balances out, leaving the interior velocity measurements to dominate.
For a rectangular aperture, corner information can provide a strong bias.

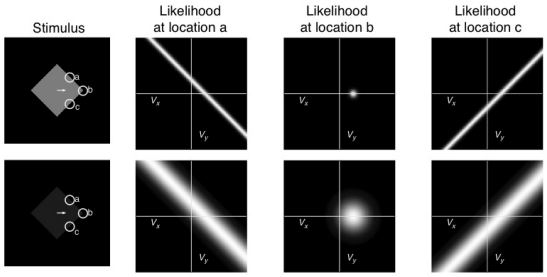
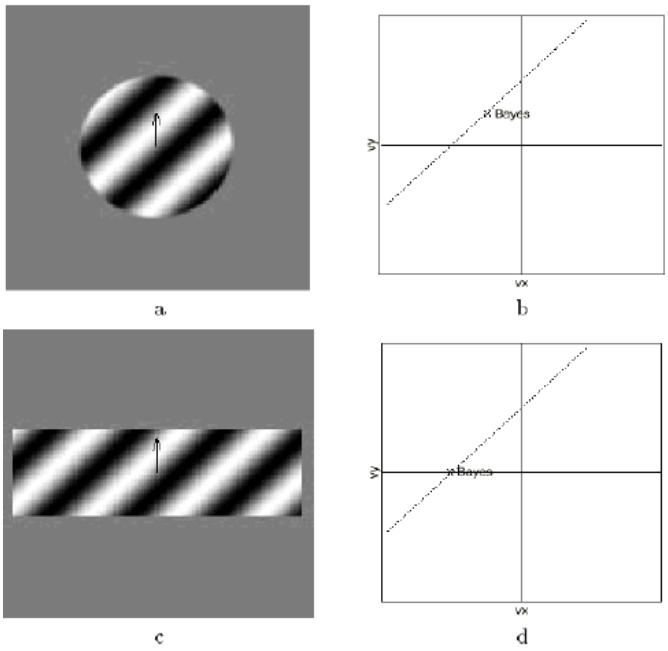
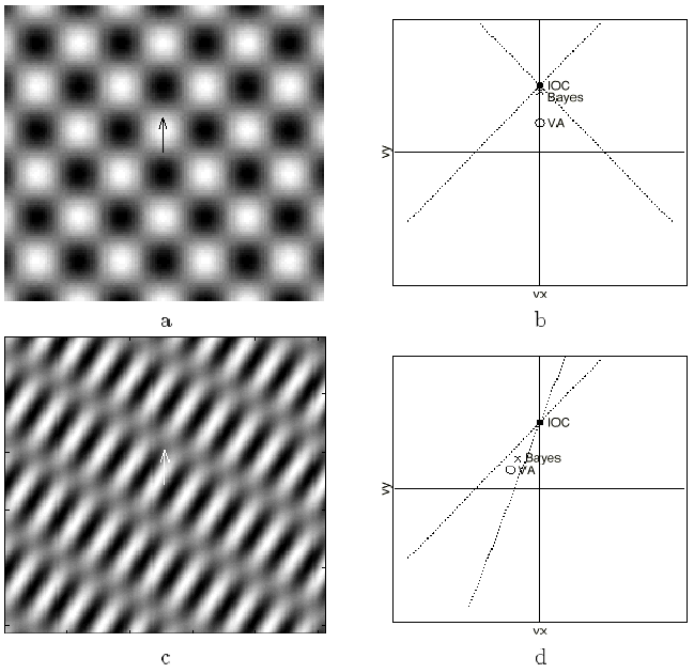


Fig. 3. Likelihood functions for three local patches of a horizontally translating diamond stimulus, computed using equation (4). Intensity corresponds to probability. Top, high-contrast sequence. Bottom, low-contrast sequence, with the same parameter σ. At edges, the local likelihood is a 'fuzzy' constraint line; at corners, the local likelihood peaks around the veridical velocity. The sharpness of the likelihood decreases with decreasing contrast.

So for the rectangular aperture below, there is more and overall stronger evidence for leftward motion.



Plaids



From Weiss and Adelson, 1998. Type I and I plaids. (Yo and Wilson, 1992)

Next time

In the next lectures we'll cover more intermediate-level perceptual processes.

Later we'll come back to motion perception, but in the larger context of what motion is used for: computing 3D layout, shape, and time to contact.

Appendix

Generalization of the Bayesian motion model to other types of motion stimuli

Requirements for generalizatoin:
Base likelihoods on actual image data
spatiotemporal measurements
Include "2D" features
E.g. corners
Rigid rotations, non-rigid deformations
Stage 1:local likelihoods
Stage 2: Bayesian combination
- Prior
slowness -- wagon wheel example, quartet example
smoothness - e.g. translating rigid circle

Overview Weiss, Simoncelli, Adelson models

Weiss Y, Simoncelli EP, Adelson EH (2002) Motion illusions as optimal percepts. Nat Neurosci 5:598-604 has a shorter version of the theory. And http://www-bcs.mit.edu/people/yweiss/intro/intro.html has a more complete algorithm that takes as input the actual image intensity values.

Dense to sparse:

$$\mathbf{v}(r) = \Phi(r)\theta$$

$$v_x(x, y) = \sum_{i=1}^{N/2} \theta_i G(x - x_i, y - y_i)$$

$$v_y(x, y) = \sum_{i=1+N/2}^{N} \theta_i G(x - x_i, y - y_i)$$

Likelihood:

$$L(v) \propto e^{-\sum_r w(r)(I_x v_x + I_y v_y + I_t)^2 / 2\sigma^2}$$

$$L_r(v) \to p(I \mid \theta) \propto \prod_r L_r(\theta)$$

Prior:

$$P(V) \propto e^{-\sum_r (Dv)^t(r)(Dv)(r)/2}$$

$$P(V) \to P(\theta)$$

Posterior:

$$P(\theta \mid I) \propto P(I \mid \theta) P(\theta)$$

Log posterior is quadratic in θ, –> linear estimator for θ

Weiss & Adelson, 1998

References

Adelson EH, Movshon JA (1982) Phenomenal Coherence of Moving Visual Patterns. Nature 300:523-525.

Adelson, E. H., & Bergen, J. R. (1985). Spatiotemporal Energy Models for the Perception of Motion. Journal of the Optical Society of America, 2(2), 284-299.

Anstis, S. M. and Rogers, B. J. (1986) Illusory continuous motion from oscillating positive-negative patterns: implications for motion perception. Perception 15, 627-640.

Barlow, H. B., & Levick, R. W. (1965). The Mechanism of Directional Selectivity in the Rabbit's Retina. Journal of Physiology, 173, 477-504.

Barlow, H. (1996). Intraneuronal information processing, directional selectivity and memory for spatio-temporal sequences. Network: Computation in NeuralSystems, 7, 251-259.

Braddick, O. J. (1974). A short-range process in apparent motion. Vision Research, 14, 519-527.

Hassenstein, B., & Reichardt, W. (1956). Functional Structure of a Mechanism of Perception of Optical Movement. Proceedings First International Congress on Cybernetics, 797-801.

He, S., & Masland, R. H. (1997). Retinal direction selectivity after targeted laser ablation of starburst amacrine cells. Nature, 389(6649), 378-82.
Gershman, S. J., Tenenbaum, J. B., & Jakel, F. (2015). Discovering hierarchical motion structure. Vision Research, 1–10. http://doi.org/10.1016/j.visres.2015.03.004
Hildreth, E. C. (1984). The Measurement of Visual Motion. Cambridge, MA: MIT Press.
Koch, C., Torre, V., & Poggio, T. (1986). Computations in the Vertebrate Retina: Motion Discrimination Gain Enhancement and Differentiation. Trends in Neuroscience, 9(5), 204-211.
Mather G, and Moulden B (1980) A simultaneous shift in apparent direction: evidence for a distribution-shift model of direction coding. Quarterly Journal of Experimental Psychology 32, 325-333.
Movshon JA, Adelson EH, Gizzi MS, Newsome WT (1986) The Analysis of Moving Visual Patterns. In: Pattern Recognition Mechanisms (Chagas C, Gattas R, Gross CG, eds), pp 117-151. Rome, Italy: Vatican Press.
Schrater PR, Simoncelli EP (1998) Local velocity representation: evidence from motion adaptation. Vision Res 38:3899-3912.
Schrater PR, Knill DC, Simoncelli EP (2000) Mechanisms of visual motion detection. Nature Neuroscience 1:64 - 68.
Van Santen, J. P. H., & Sperling, G. (1985). Elaborated Reichardt Detectors. Journal of the Optical Society of America, 2(7), 300-321.
Wandell, B. A. (1995). Foundations of Vision. Sunderland, Massachusetts: Sinauer.
Weiss, Y., & Adelson, E. H. (1998). Slow and smooth: a Bayesian theory for the combination of local motion signals in human vision (A.I. Memo No. 1624). M.I.T.
http://www-bcs.mit.edu/people/yweiss/intro/intro.html
Weiss Y, Simoncelli EP, Adelson EH (2002) Motion illusions as optimal percepts. Nat Neurosci 5:598-604.
Yuille, A., & Grzywacz, N. (1988). A computational theory for the perception of coherent visual motion. Nature, 333, 71-74.