Motion

Motion Estimation

- Given a sequence of images we might ask
- What are the moving objects in the scene?
- What sort of motion are they undergoing?
- Where will they be in the future?
- To answer these questions we need to measure the motion

- There are many problems in motion estimation
- Often the motion is ambiguous
- Image sequences contain a lot of data - efficiency is a concern
- Many interesting tasks involve complex motion - e.g. facial expression analysis

Simple Techniques

Motion Difference

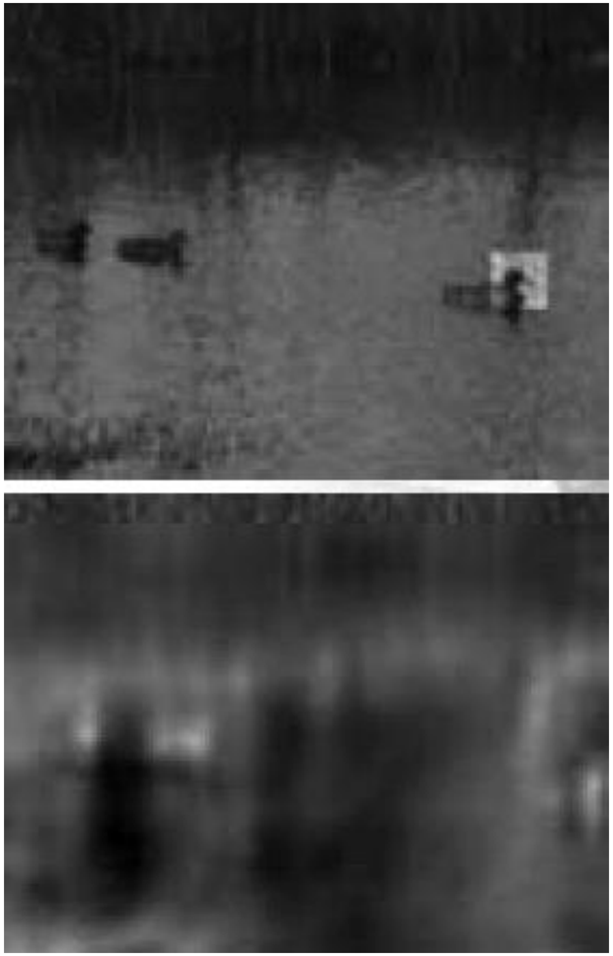
- Take two images from a sequence
- Compute the change in brightness at each pixel in the image
- Threshold

Background Models

- Find the average brightness at each pixel over a sequence
- Use the difference between the current frame and the average to find moving objects

Simple Techniques

- Area-based matching can also be used
- We take a template from the first image
- This is then compared to points in the second image to find corresponding regions
- This uses a ‘distance measure’ to compare patches



Motion Field and Optical Flow

The Motion Field

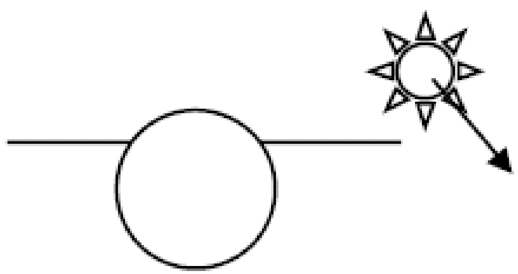
- "assigns a velocity vector to each point in the image"
- Tells us how the position of the *image* of the corresponding *scene point* changes over time
- Can be computed from the *scene* to tell us about the *image*

Optical Flow

- The "apparent motion of the brightness pattern" in an image
- Ideally it will be the same as the motion field, but this is not always the case
- Can be computed from the *image*, to tell us about the *scene*

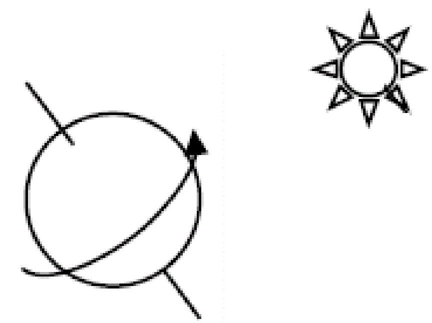
Optical Flow ≠ Motion Field

A Moving light



- The *image* changes so there is optical flow
- The *scene* objects do not move so there is no motion field

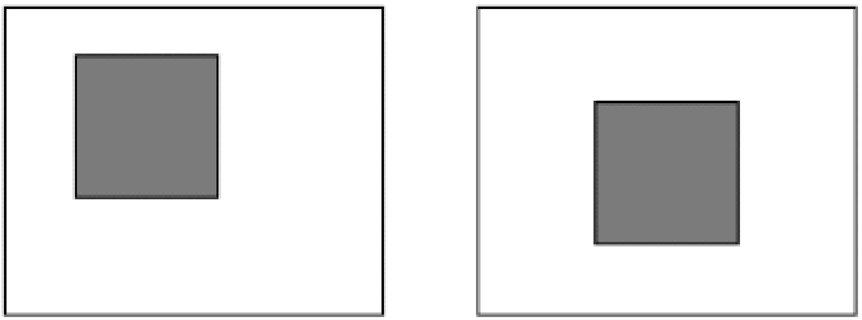
A Rotating Sphere



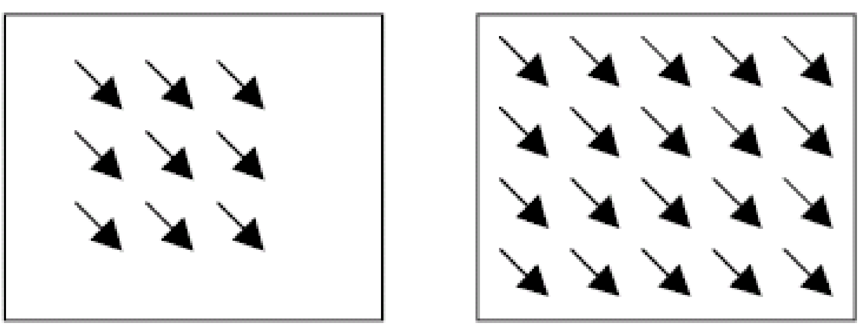
- The *scene* object moves, so there is motion field
- The *image* does not change, so there is no optic flow

Optical Flow is Ambiguous

- Consider the two images below:



- Two possible fields (of



So, optical flow

- Is not always what we want to compute
- Cannot be determined without ambiguity
- But it is all that we can compute from the images
- This means we need to make assumptions to find a *reasonable* flow field estimate

Brightness Constancy

Brightness constancy assumption:

$$I(x,y,t) = I(x+u, y+v, t+1)$$

- $I(x,y,t)$ is the brightness of the image at location (x,y) and time t
- (u,v) is the motion field at location (x,y) and time t
- This assumption is true apart from the effects of lighting (including shadows, reflections, and highlights)

Brightness Constancy

Another way to express brightness constancy is that

$$\frac{dI(x,y,t)}{dt} = 0$$

- This says that the image doesn't change over time - it just moves about

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0$$

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0$$

Brightness Constancy

$$\frac{\partial I(x,y,t)}{\partial x}u+\frac{\partial I(x,y,t)}{\partial y}v+\frac{\partial I(x,y,t)}{\partial t}=0$$

Image derivative in x direction

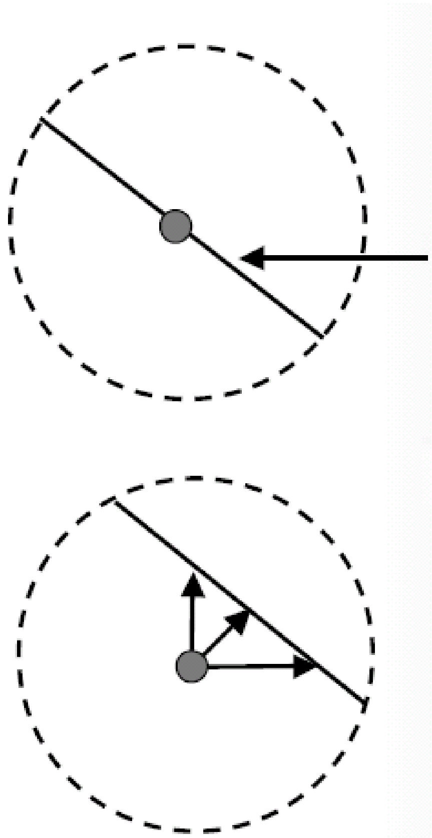
Image derivative in y direction

Image derivative in t direction

The Aperture Problem

There is no solution to the equation

- We can determine the component of flow in the same direction as the image intensity gradient
- We cannot determine the component of flow perpendicular to it
- This is the *aperture problem*



Flow Smoothness

We need another constraint to find a unique solution

- This is the constraint that the flow field is smooth
- Neighbouring pixels in the image should have similar optical flow

We want u and v to have low variation

- We can do this by trying to set

$$(u - \bar{u}) = 0, \ (v - \bar{v}) = 0$$

- So u and v are equal to the average of their neighbouring values

Squared Errors

We now have three error terms

- If we square them then the error is always positive, and we can look for a minimum
- A weighting term, λ, balances the influence of the brightness and smoothness errors

The squared error term is

- To minimise: take derivatives with respect to *u* and *v*, set to 0, then solve

$$\lambda\left(\frac{\partial I}{\partial x}u+\frac{\partial I}{\partial y}v+\frac{\partial I}{\partial t}\right)^2+(u-\bar{u})^2+(v-\bar{v})^2$$

Minimisation

$$e = \lambda\left(\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t}\right)^2 + (u - \overline{u})^2 + (v - \overline{v})^2$$

$$\frac{\partial e}{\partial u} = 2\lambda\frac{\partial I}{\partial x}\left(\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t}\right) + 2(u - \overline{u}) \quad = 0$$

$$\frac{\partial e}{\partial v} = 2\lambda\frac{\partial I}{\partial y}\left(\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t}\right) + 2(v - \overline{v}) \quad = 0$$

A Solution

Solving the two equations gives

$$u = \overline{u} - \lambda \frac{\partial I}{\partial x} \frac{\overline{u}\,\frac{\partial I}{\partial x} + \overline{v}\,\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t}}{1 + \lambda\left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right)}$$

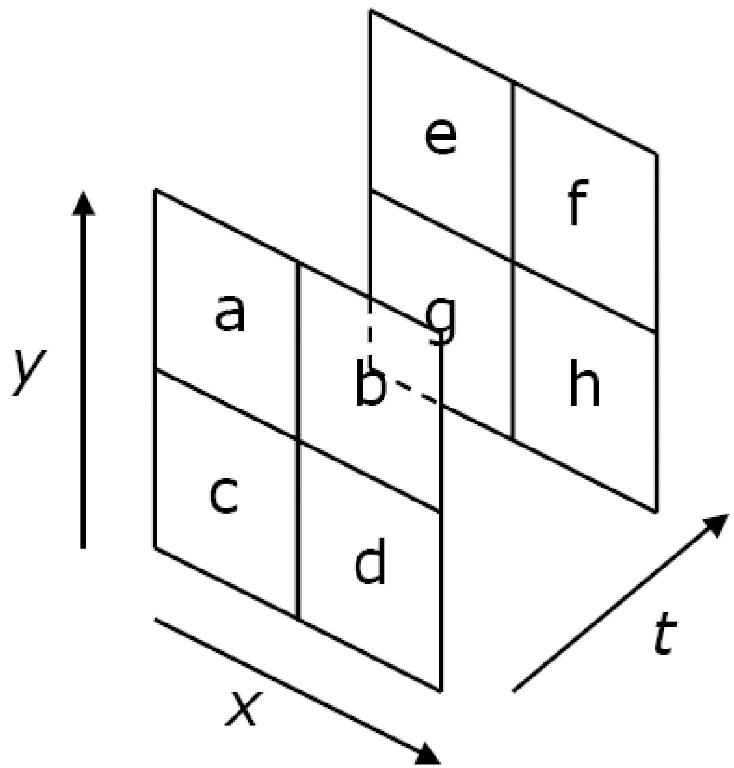
$$v = \overline{v} - \lambda \frac{\partial I}{\partial y} \frac{\overline{u}\,\frac{\partial I}{\partial x} + \overline{v}\,\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t}}{1 + \lambda\left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right)}$$

But we need to know $\sim u$ and $\sim v$ to compute u and v

Iterative solution:

- Estimate u and v
- Then compute the averages, $\sim u$ and $\sim v$
- Then make a new estimate of u and v
- Then make a new estimate of $\sim u$ and $\sim v$
- etc…

Computing the Optical Flow



Gradients:

$dI/dx = (b+d+f+h) - (a+c+e+g)$

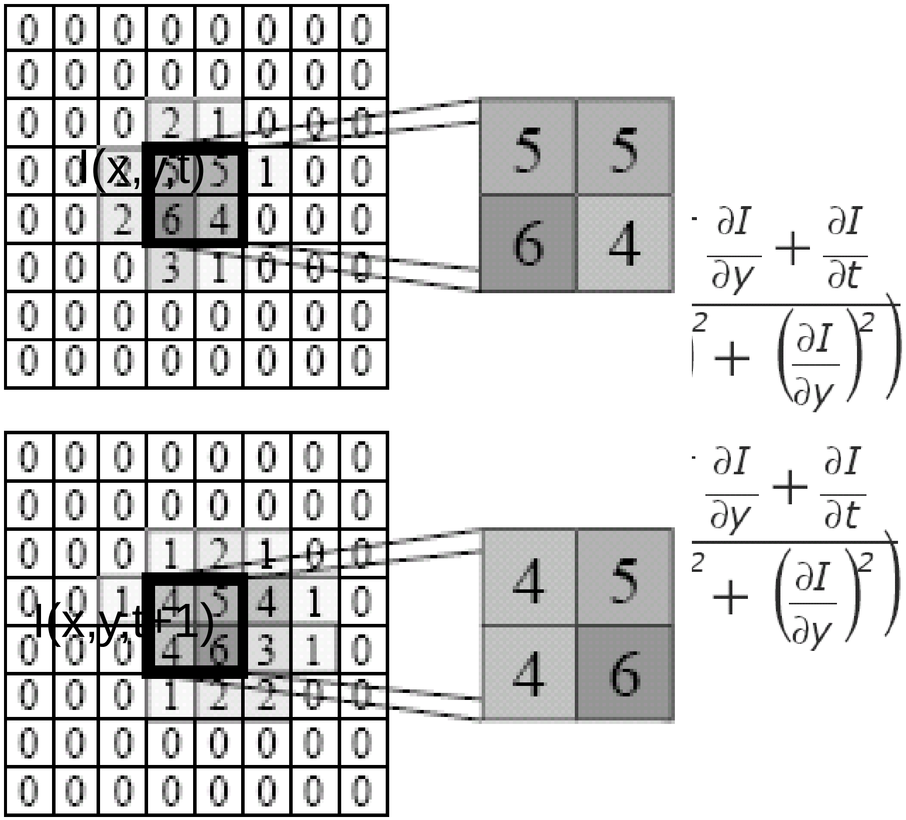
$dI/dy = (a+b+e+f) - (c+d+g+h)$

$dI/dt = (e+f+g+h) - (a+b+c+d)$

$$u = \overline{u} - \lambda \frac{\partial I}{\partial x} \frac{\overline{u}\frac{\partial I}{\partial x} + \overline{v}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t}}{1 + \lambda\left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right)}$$

$$v = \overline{v} - \lambda \frac{\partial I}{\partial y} \frac{\overline{u}\frac{\partial I}{\partial x} + \overline{v}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t}}{1 + \lambda\left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right)}$$

Computing the Optical Flow



dI/dx = (5+4+5+6)-(5+6+4+4)

dI/dy = (5+5+4+5)-(6+4+4+6)

dI/dt = (5+5+6+4)-(4+5+4+6)

Computing the Optical Flow

The algorithm is iterative

- We start with an initial estimate
- We refine it over a series of cycles
- We need an initial estimate
- We also need to know when to stop

Initialisation

- We can start with an estimate of u and *v* of 0 everywhere
- Stop when the results at iteration *n* and *n+1* are very similar
- This is when the algorithm converges
- Can we be sure it will?

Computing the Optical Flow