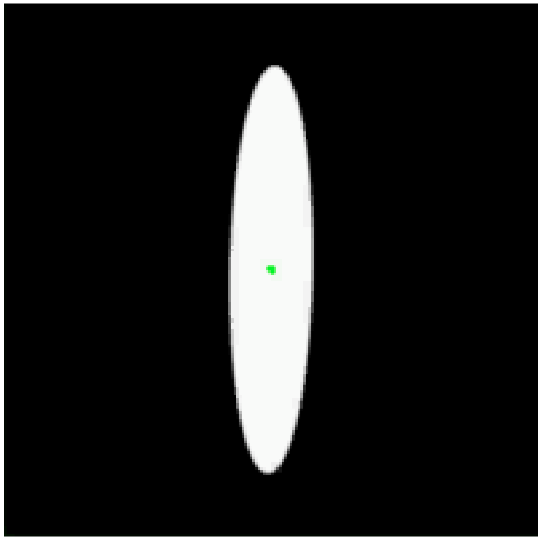
What is Motion?

As Visual Input:

- Change in the spatial distribution of light on the sensors.

Minimally, $dI(x,y,t)/dt \neq 0$



As Perception:

- Inference about causes of intensity change, e.g.

$I(x,y,t) \longrightarrow v_{OBJ}(x,y,z,t)$

Motion Field: Movement of Projected points

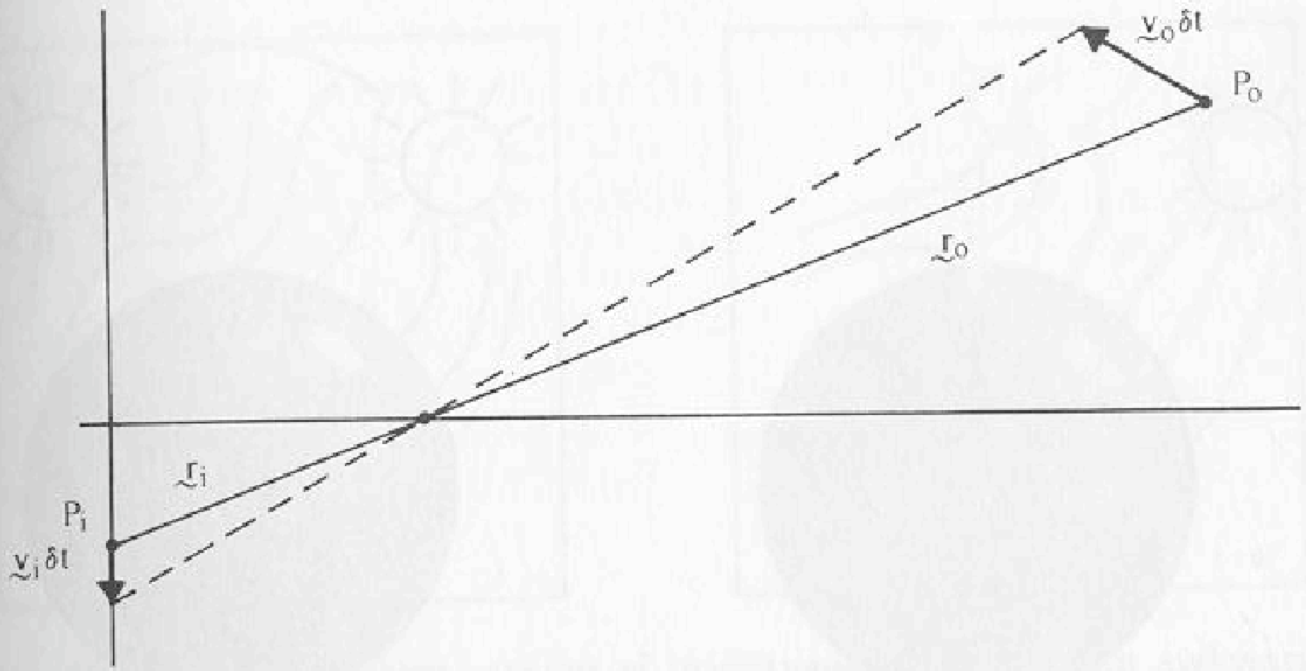
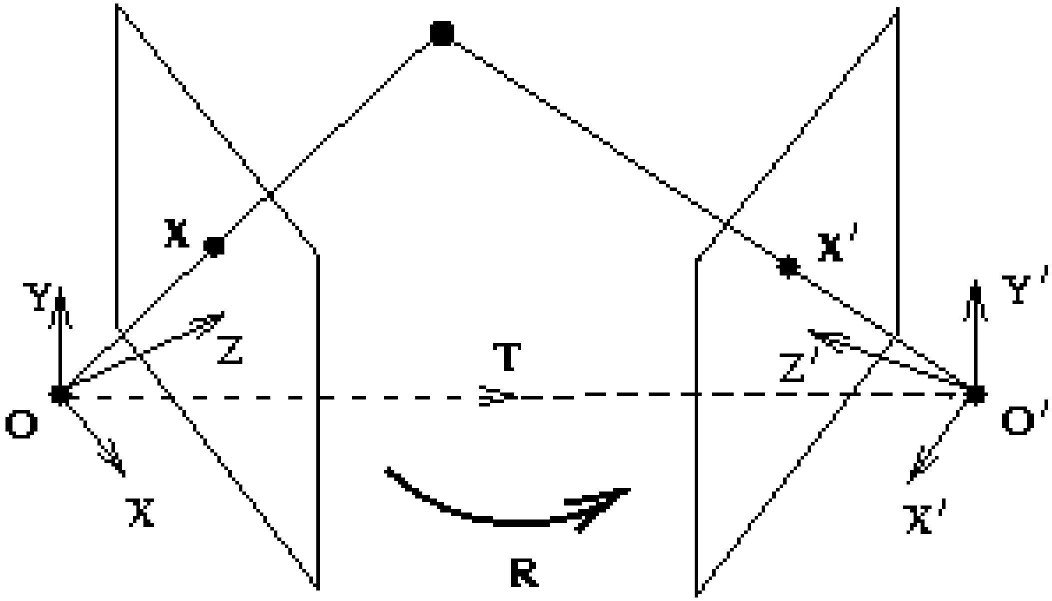


Figure 12-1. Displacement of a point in the environment causes a displacement of the corresponding image point. The relationship between the velocities can be found by differentiating the perspective projection equation.

Essential Matrix



$$\mathbf{a} \times \mathbf{v} = \begin{bmatrix} a_y v_z - a_z v_y \\ a_z v_x - a_x v_z \\ a_x v_y - a_y v_x \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

$$= \mathbf{A}\mathbf{v}$$

$$x' = \begin{bmatrix} R & t \\ 000 & 1 \end{bmatrix} x$$

$$\overrightarrow{O'x'} \bullet \left[\overrightarrow{Ox} - \overrightarrow{OO'} \right] = 0$$

$$x' \bullet \left(\vec{t} \times Rx \right) = 0$$

$$x' \bullet \left(\mathcal{E} x \right) = 0$$

$$x' \mathcal{E} x = 0$$

$$\mathcal{E} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} R$$

Differential Camera Motion

- For a small rotation around an axis □, rotation of the camera frame can be expressed:

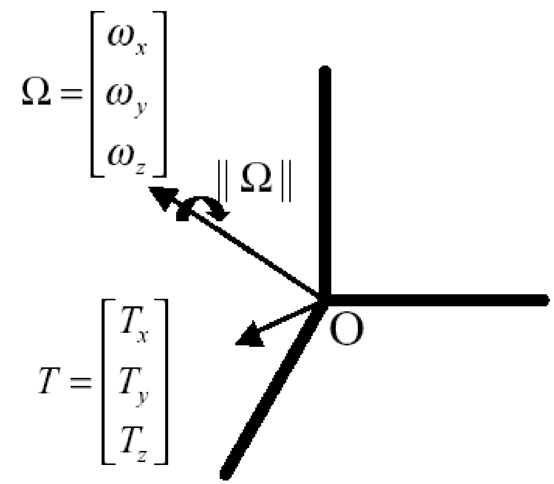
$$\vec{x}' = R\vec{x} + \vec{t}$$

$$\vec{x}' = \vec{x} + (\square \, \square \, \vec{x})dt + \vec{v}dt$$

$$= (I + dt[\square_{\square}])\vec{x} + \vec{v}dt$$

And motion of a point:

$$p' = p + \frac{dp}{dt}dt$$



Thus

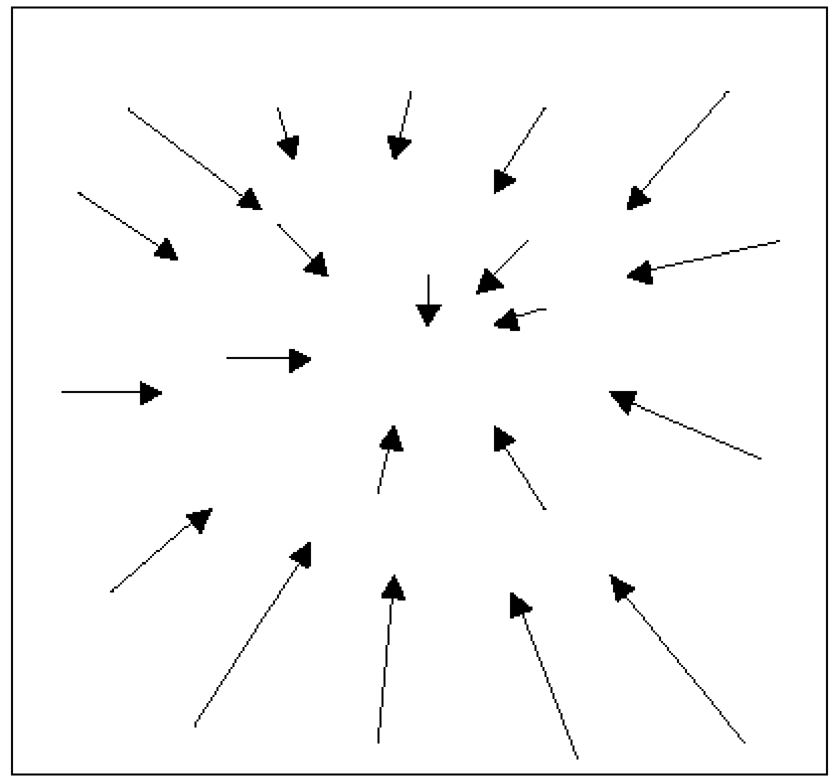
$$\boldsymbol{t} = \delta t \, \boldsymbol{v},$$

$$\mathcal{R} = \mathrm{Id} + \delta t \, [\boldsymbol{\omega}_\times]$$

$$\boldsymbol{p}' = \boldsymbol{p} + \delta t \, \dot{\boldsymbol{p}}.$$

Epipolar:

$$\boldsymbol{p}^T [\boldsymbol{v}_\times](\mathrm{Id} + \delta t \, [\boldsymbol{\omega}_\times])(\boldsymbol{p} + \delta t \, \dot{\boldsymbol{p}}) = 0$$



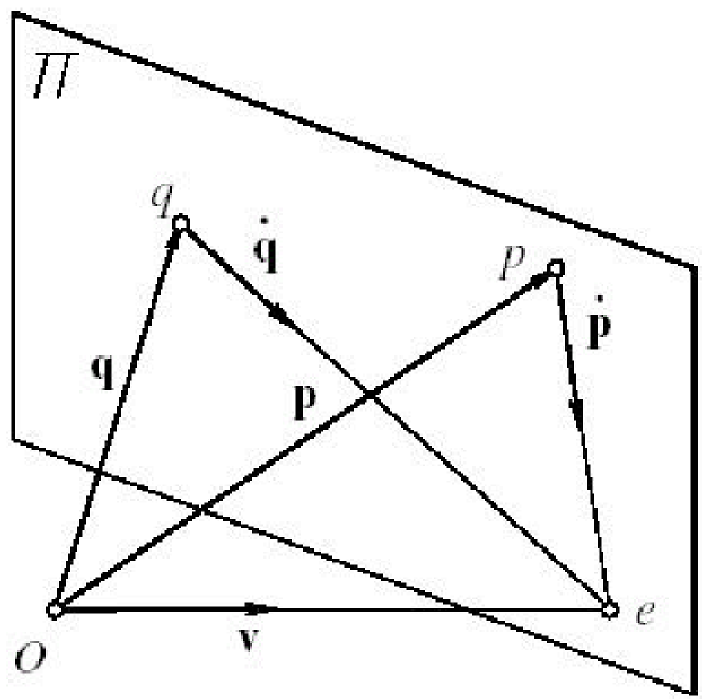


FIGURE 11.3: Focus of expansion: under pure translation, the motion field at every point in the image points toward the focus of expansion.

Basic Idea

- 1) Estimate point motions
- 2) use point motions to estimate camera/object motion
- Problem: Motion of projected points not directly measurable.
- -Movement of projected points creates displacements of image patches -- Infer point motion from image patch motion
  - Matching across frames
  - Differential approach
  - Fourier/filtering methods

Differential approach: Optical flow constraint equation

Brightness should stay constant as you track motion

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

1st order Taylor series, valid for small δt

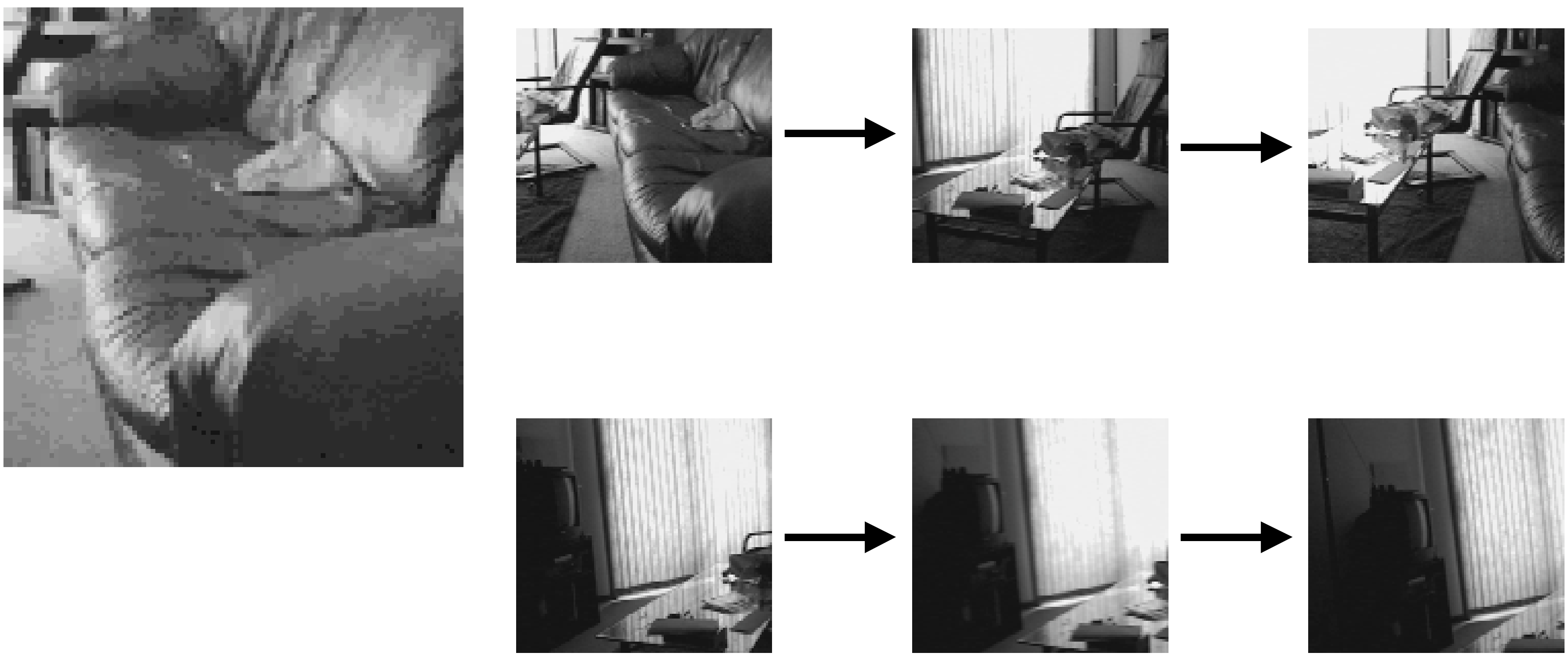
$$I(x,y,t) + u\delta t I_x + v\delta t I_y + \delta t I_t = I(x,y,t)$$

Constraint equation

$$\boxed{uI_x + vI_y + I_t = 0}$$

"BCCE" - Brightness Change Constraint Equation

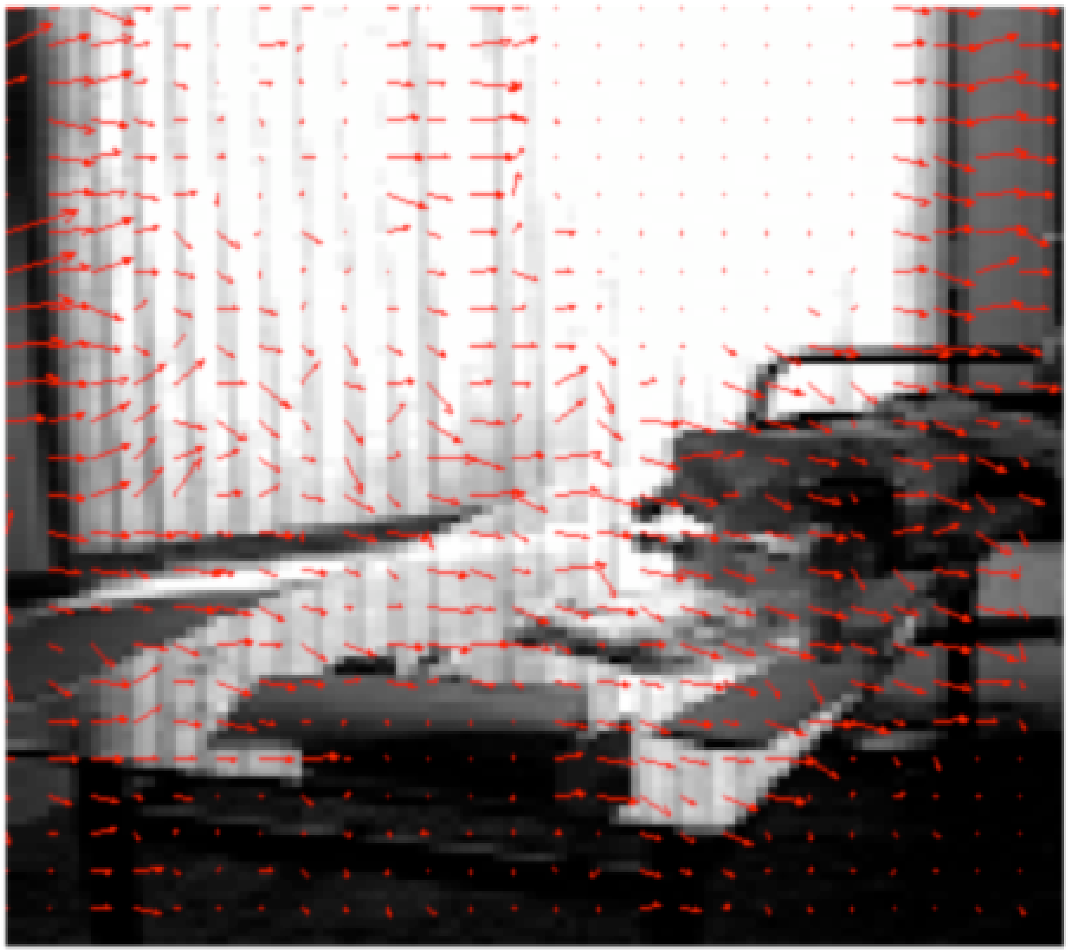
Image sequence from Egomotion



Local Translations

OpticFlow:
(Gibson,1950)
Assigns local image velocities v(x,y,t)

Time ~100msec
Space ~1-10deg



Brightness constraint

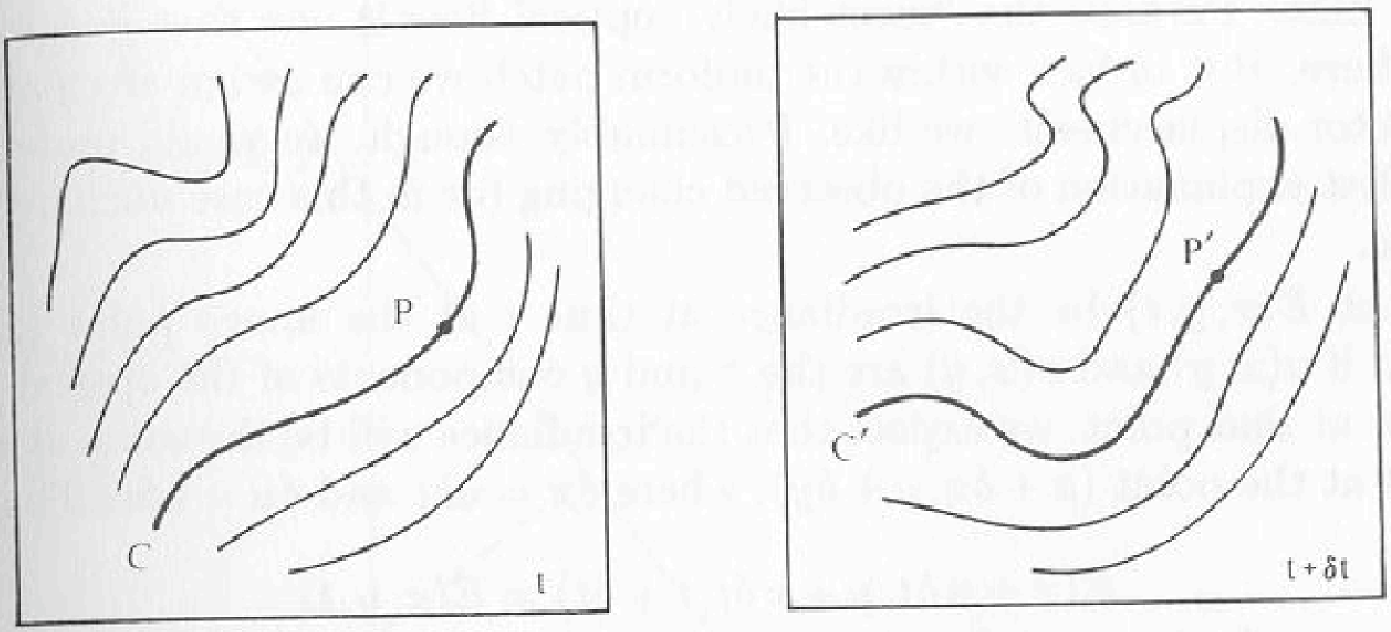


Figure 12-3. The apparent motion of brightness patterns is an awkward concept. It is not easy to decide which point P' on a contour C' of constant brightness in the second image corresponds to a particular point P on the corresponding contour C in the first image.

Photometric Motion

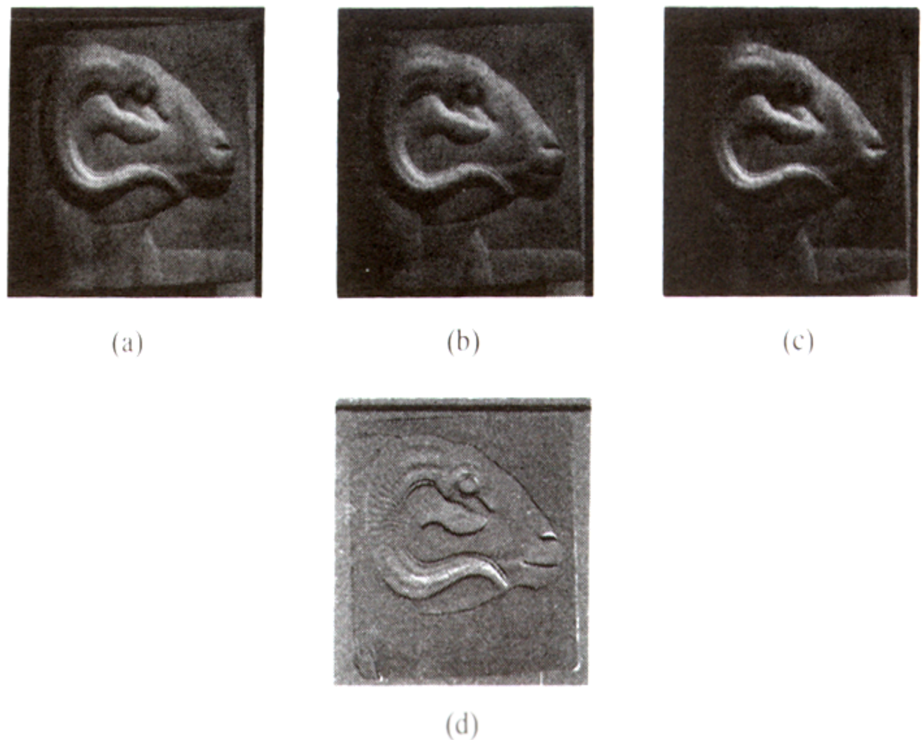
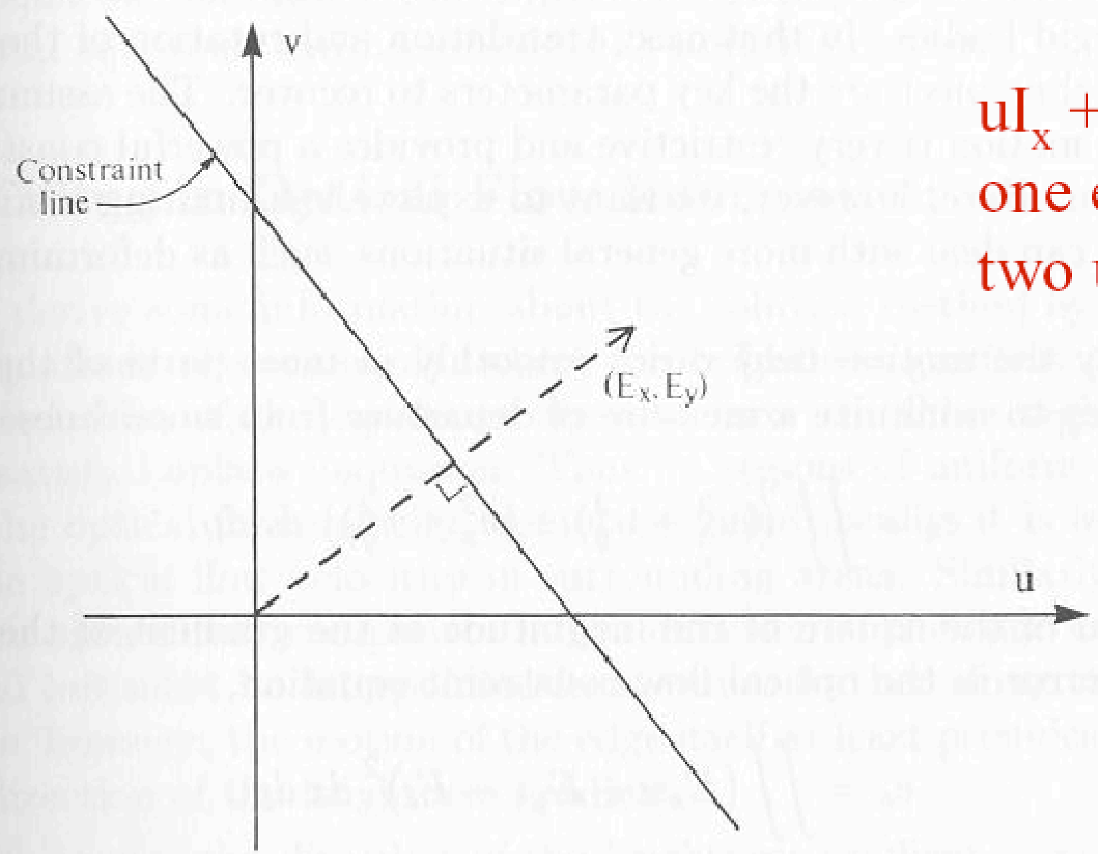


Fig. 2(a)–(c). Three frames of an image sequence of a rotating object; (d) difference between the first and last images. The geometric displacement between the first and last images has a mean of 0.44 pixels and a standard deviation of 0.52 pixels, whereas the photometric variation has a mean of 14.28 grey levels and a standard deviation of 18.87 grey levels.

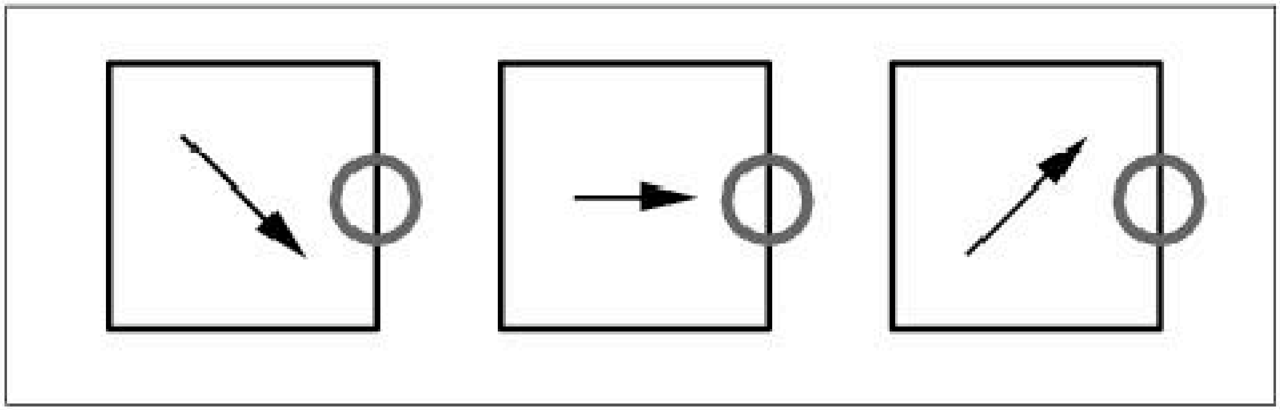
Brightness constancy constraint line



$uI_x + vI_y + I_t = 0$:
one equation,
two unknowns!

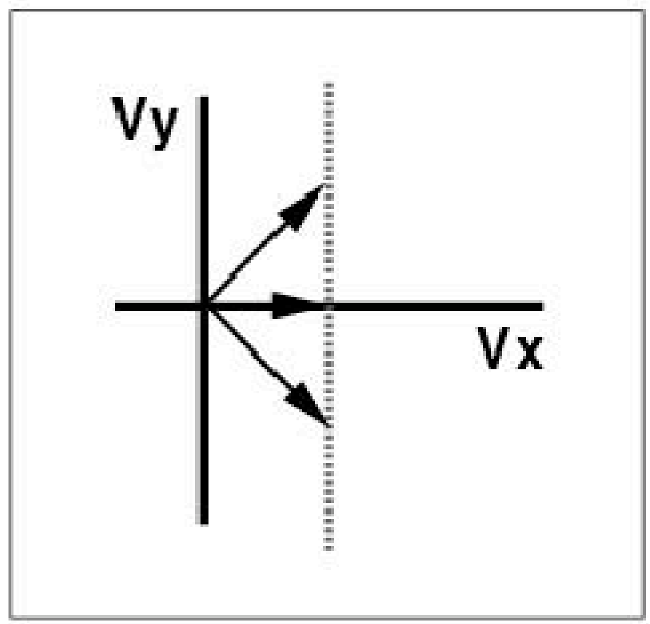
Figure 12-4. Local information on the brightness gradient and the rate of change of brightness with time provides only one constraint on the components of the optical flow vector. The flow velocity has to lie along a straight line perpendicular to the direction of the brightness gradient. We can only determine the component in the direction of the brightness gradient. Nothing is known about the flow component in the direction at right angles.

Problem: Images contain many edges-- Aperture problem

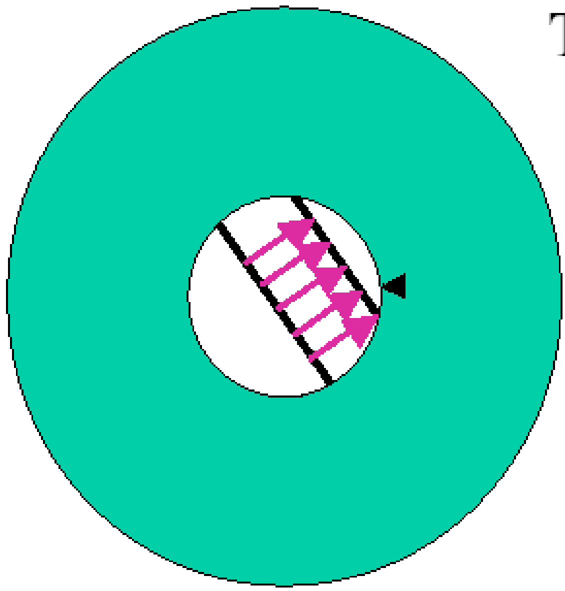


a

Normal flow:
Motion component in the direction of the edge



b

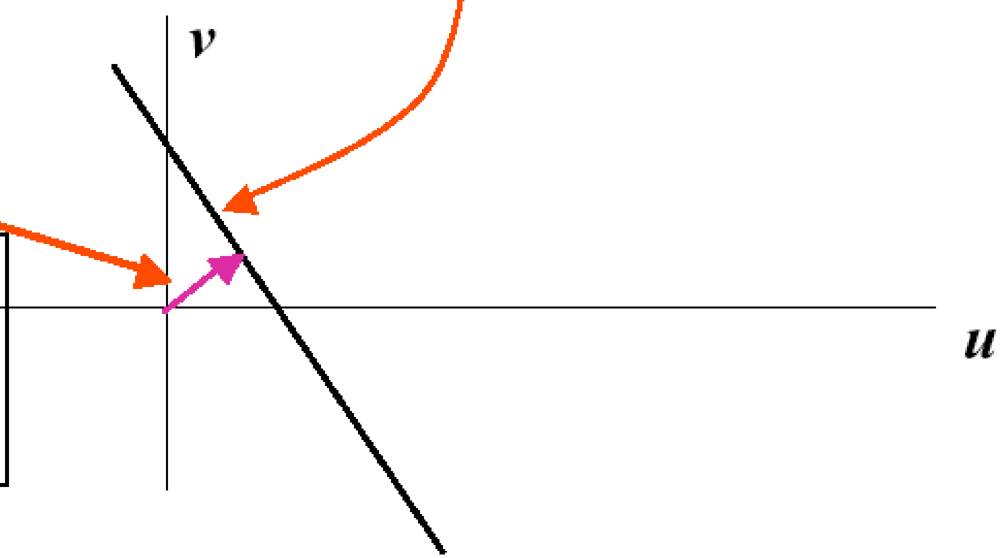


The gradient constraint:

$$I_x u + I_y v + I_t = 0$$

$$\nabla I \bullet \vec{U} = 0$$

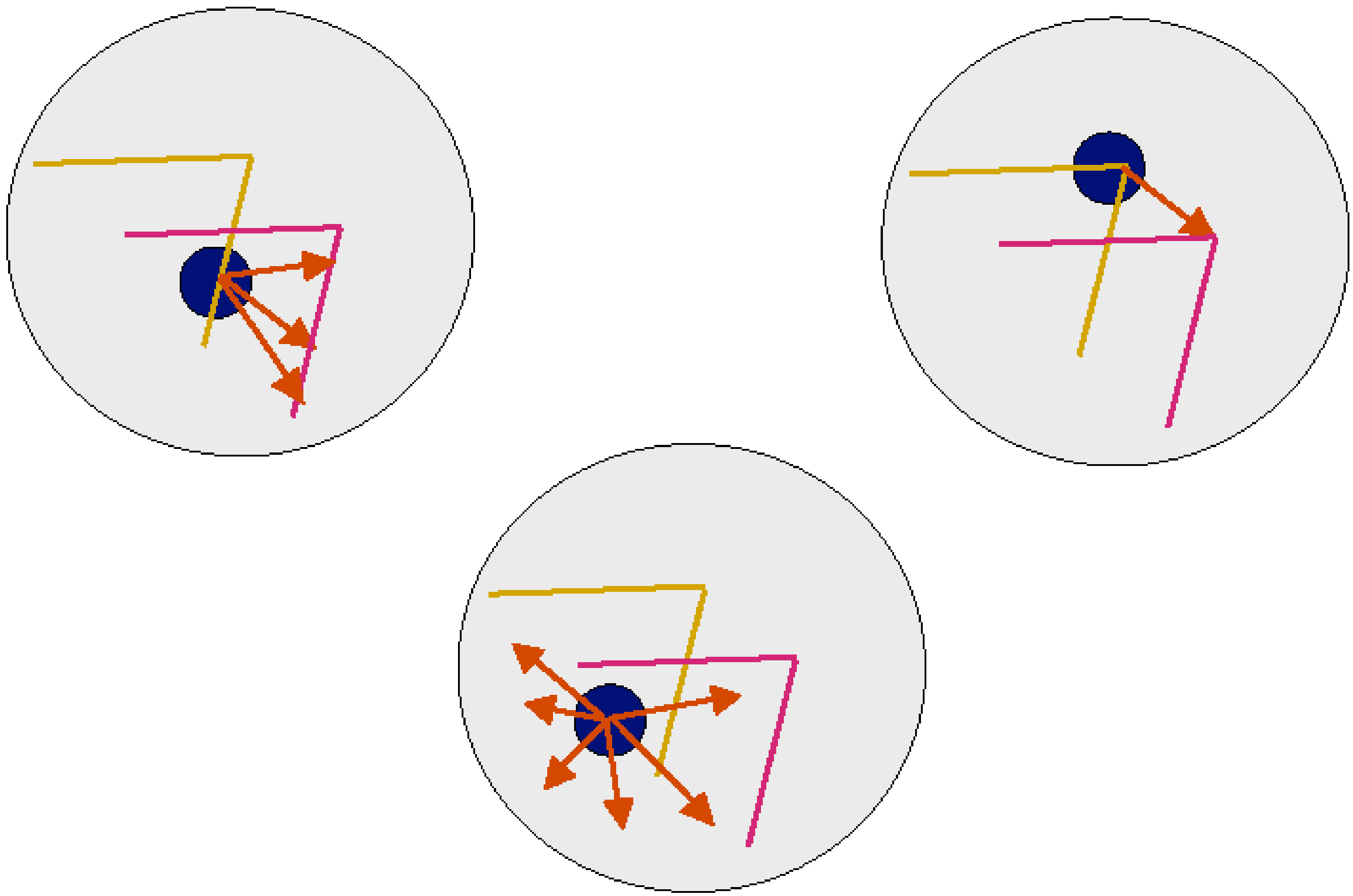
Defines a line in the *(u,v)* space



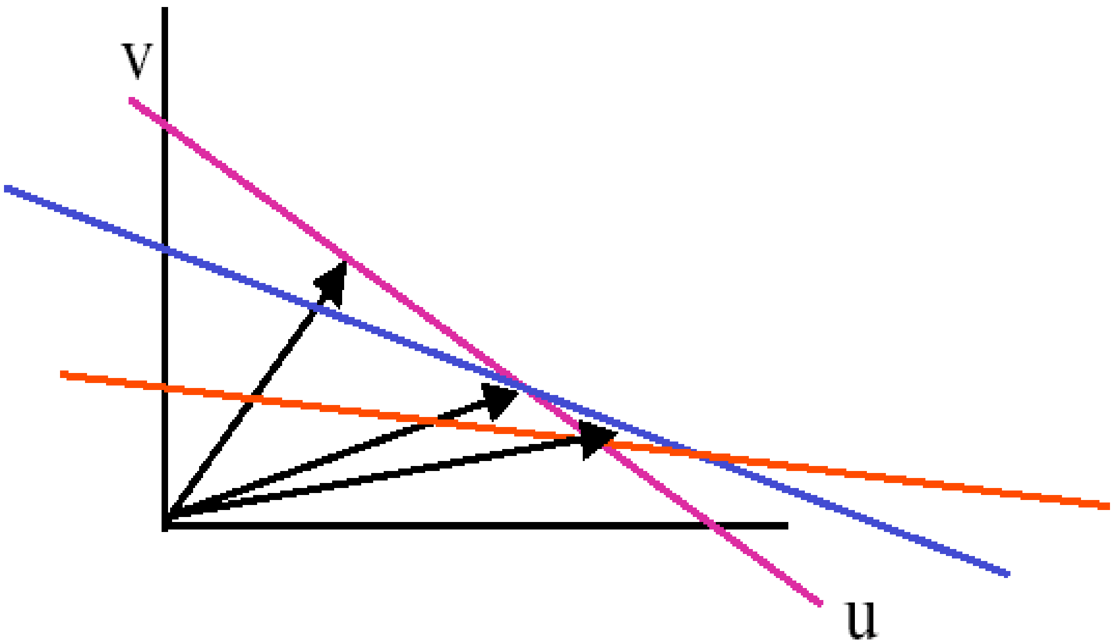
Normal Flow:

$$u_{\perp} = -\frac{I_t}{|\nabla I|}\frac{\nabla I}{|\nabla I|}$$

Local Patch Analysis



Combining Local Constraints



$$\nabla I^1 \bullet U = -I_t^1$$

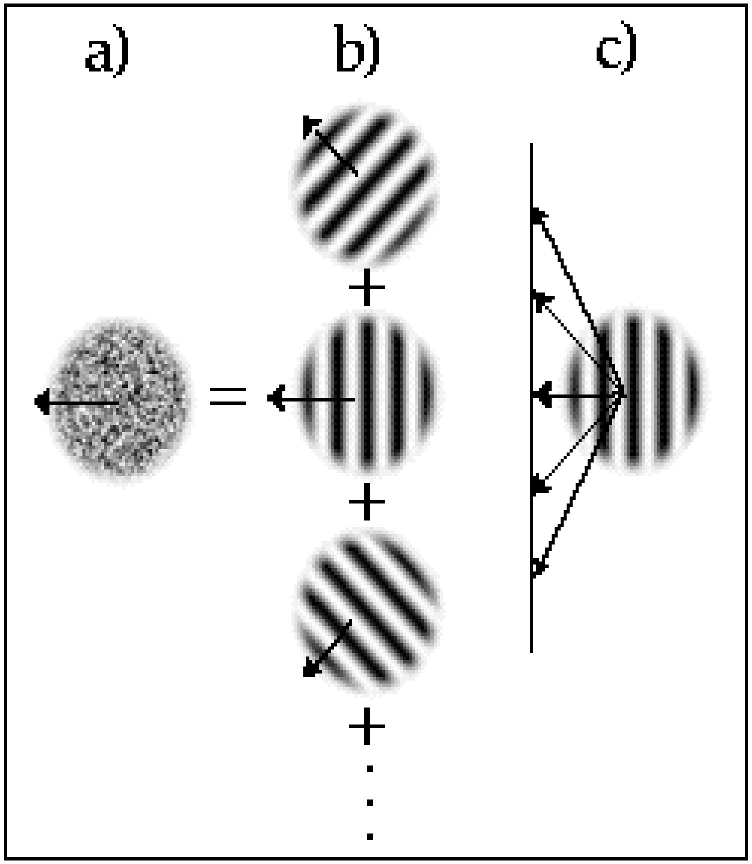
$$\nabla I^2 \bullet U = -I_t^2$$

$$\nabla I^3 \bullet U = -I_t^3$$

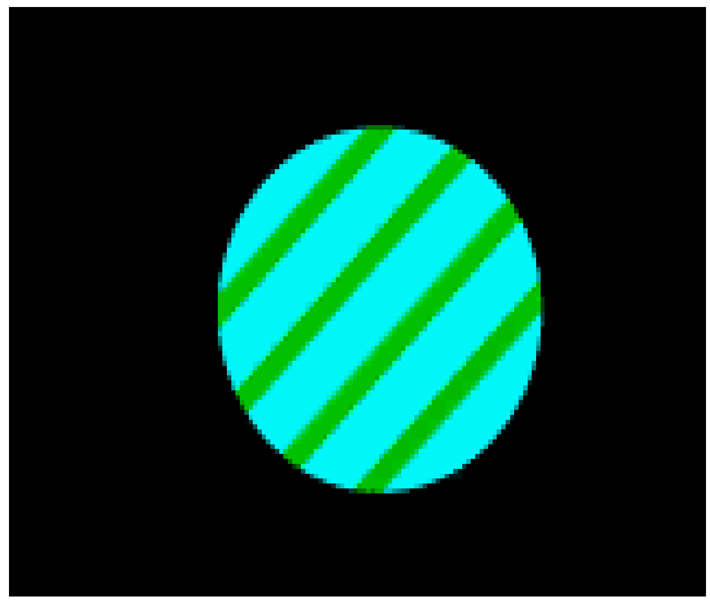
etc.

Find Least squares solution for multiple patches.

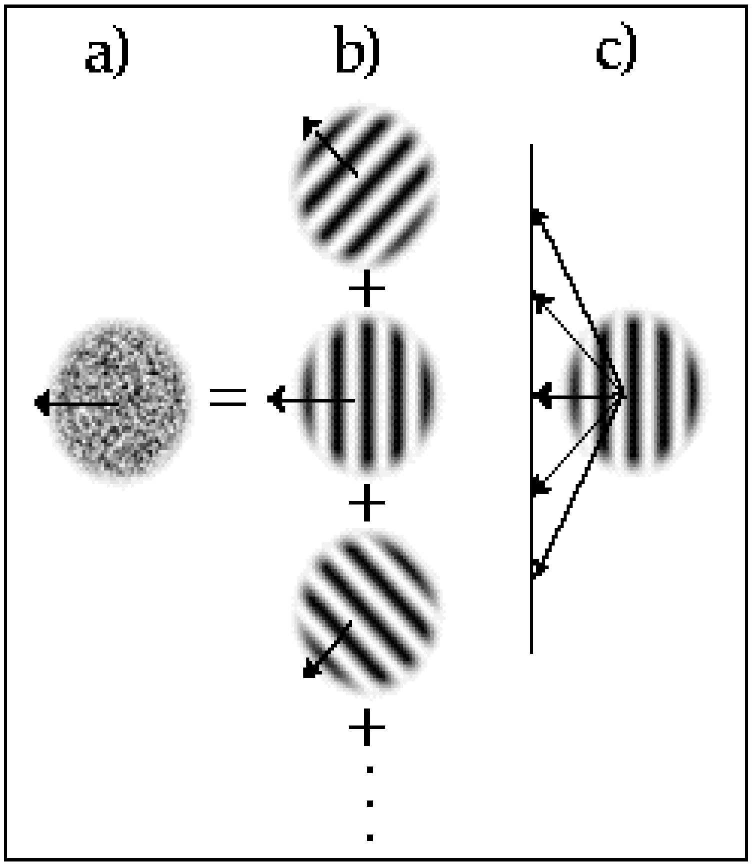
Aperture Problem (Motion/Form Ambiguity)



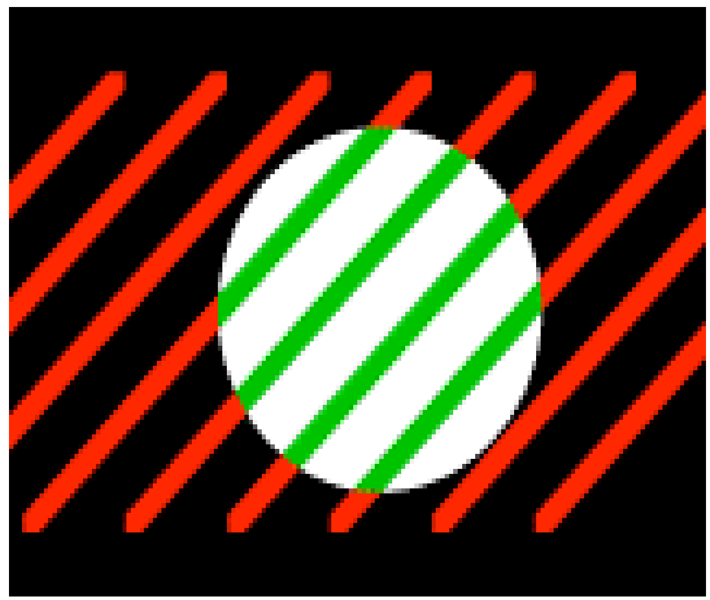
Result: Early visual measurements are ambiguous w.r.t. motion.



Aperture Problem (Motion/Form Ambiguity)



However, both the motion and the form of the pattern are implicitly encoded across the *population* of V1 neurons.



Actual motion

Lucas-Kanade: Integrate over a Patch

Assume a single velocity for all pixels within an image patch

$$E(u,v) = \sum_{x,y\in\Omega}\left(I_x(x,y)u + I_y(x,y)v + I_t\right)^2$$

Solve with:

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}\begin{pmatrix} u \\ v \end{pmatrix} = -\begin{pmatrix} \sum I_x I_t \\ \sum I_y I_t \end{pmatrix}$$

On the LHS: sum of the 2x2 outer product tensor of the gradient vector

$$\left(\sum \nabla I \nabla I^T\right)\vec{U} = -\sum \nabla I I_t$$

Lucas-Kanade: Singularities and the Aperture Problem

Let $M = \sum (\nabla I)(\nabla I)^T$ and $b = \begin{bmatrix} -\sum I_x I_t \\ -\sum I_y I_t \end{bmatrix}$

- Algorithm: At each pixel compute U by solving $MU = b$

- M is singular if all gradient vectors point in the same directior
  - -- e.g., along an edge
  - -- of course, trivially singular if the summation is over a single pixel
  - -- i.e., only *normal flow* is available (aperture problem)

- Corners and textured areas are OK

Solutions to “the aperture problem”

- Regularize the solution: add a velocity smoothness constraint (eg, Horn 12.6).
- Integrate over a larger region in image (e.g., Lukas and Kanade).
- More sophisticated scene models: segment object boundaries, specularities, textureless regions, etc.

Horn and Schunck

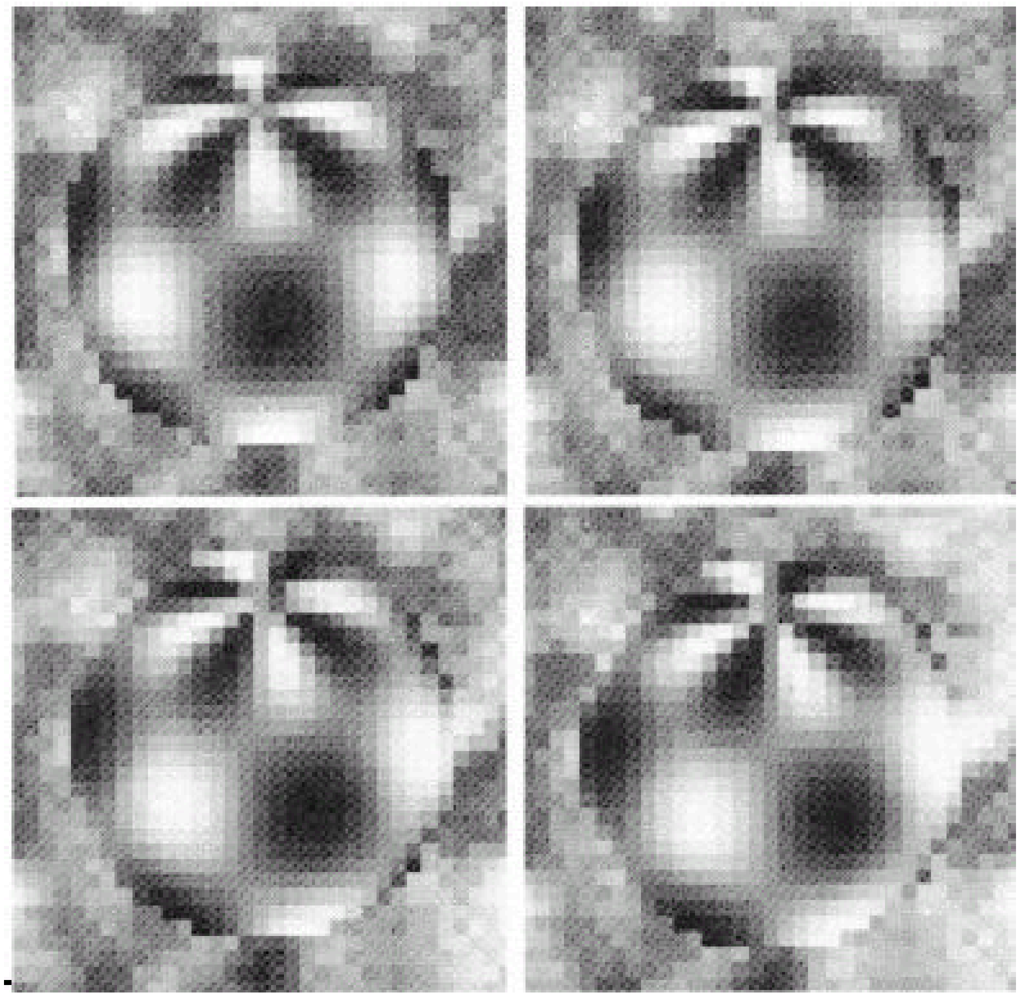
- Smoothness is most natural:
  - find **v** that minimizes

$$\int\int \left(uI_x + vI_y + I_t\right)^2 + \alpha^2\left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right)dxdy$$

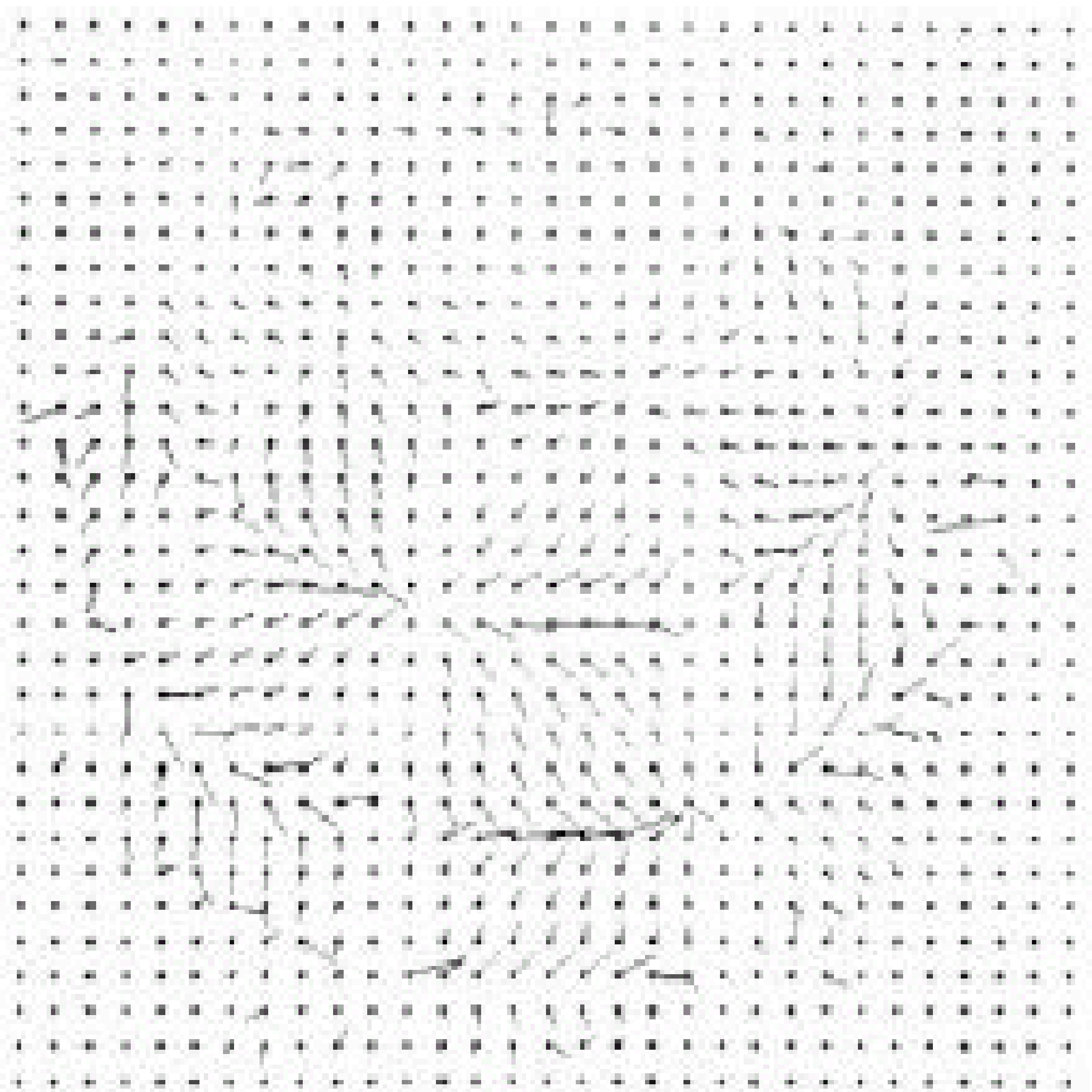
$$I_x = \frac{\partial I}{\partial x}, I_y = \frac{\partial I}{\partial y}, I_t = \frac{\partial I}{\partial t}$$

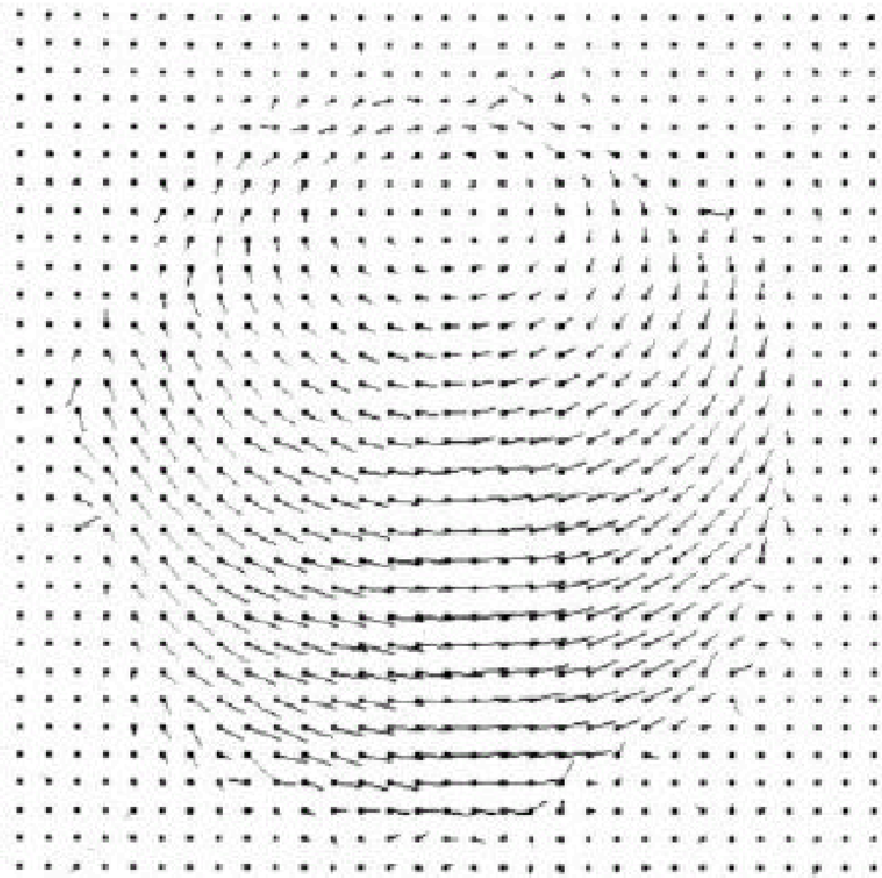
$$u = \frac{dx}{dt}, v = \frac{dy}{dt}, u_x = \frac{\partial u}{\partial x}, v_x = \frac{\partial v}{\partial x}, u_y = \frac{\partial u}{\partial y}, v_y = \frac{\partial v}{\partial y}$$

Horn and Schunck

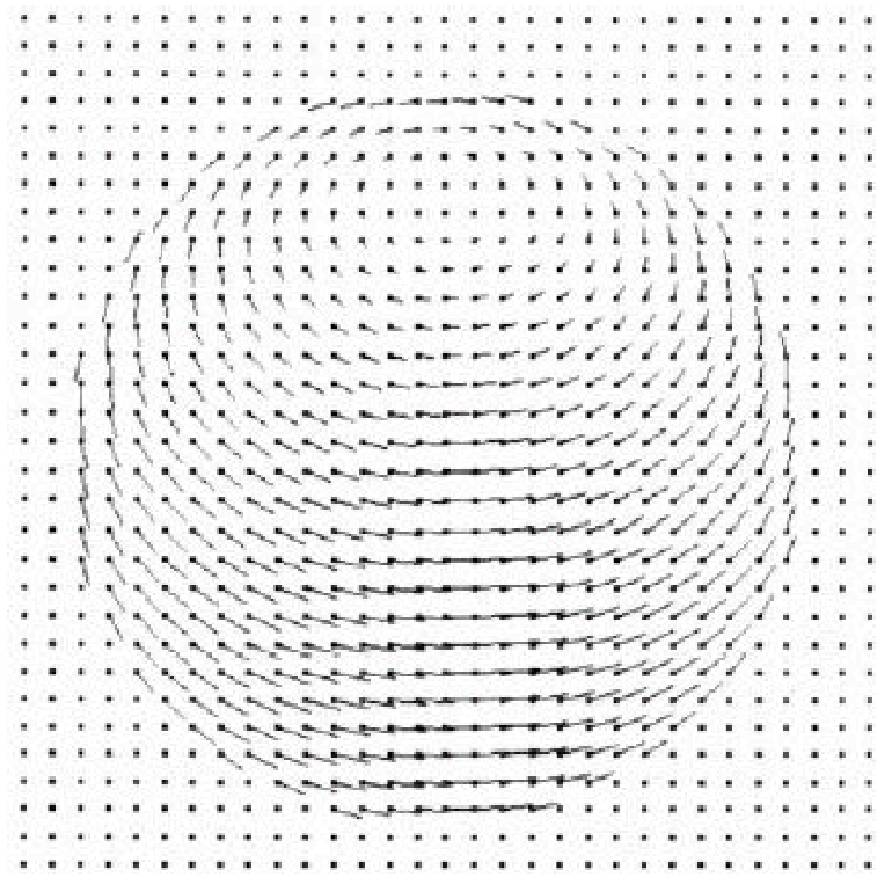


Horn and Schunck





Final estimate



Actual motion

Problems:

–edges

–large uniform regions

–can require many iterations

Fourier Methods

More fundamental than the Taylor approximation is the brightness constraint

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

For a globally translating image, $I(x,y,t)$ can be rewritten as $I(x-v_x t, y-v_y t)$. This signal can be written as the sum of sinusoids which have a constant spatial spectrum, with the translation causing a phase shift in each sinusoid:

$$I(x(t),y(t)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\omega_x,\omega_y)\exp(2\pi i(\omega_x(x-v_x t)+\omega_y(y-v_y t))d\omega_x d\omega_y \tag{2}$$

$$I = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\omega_x,\omega_y)\exp(2\pi i(\omega_x x+\omega_y y))\exp(-2\pi i(\omega_x v_x t+\omega_y v_y t))d\omega_x d\omega_y$$

The 3D Fourier Transform of this expression is:

$$\begin{aligned}\mathcal{F}\{I\} &= \int_{x,y,t}\exp(-2\pi i(\omega_x x+\omega_y y+\omega_t t)) \\ &\quad \left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\omega_x,\omega_y)\exp(-2\pi i(\omega_x(x-v_x t)+\omega_y(y-v_y t)))d\omega_x d\omega_y\right) dx\,dy\,dt\end{aligned} \tag{3}$$

$$\mathcal{F}\{I\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\omega_x, \omega_y)\delta(\omega_t + \omega_x v_x + \omega_y v_y)d\omega_x d\omega_y$$

$$\begin{aligned} \omega_t + \omega_x v_x + \omega_y v_y &= 0 \\ \vec{u} \cdot \vec{\omega} &= 0 \end{aligned}$$

- Lo

Which is the equation of a plane, weighted by the spatial texture

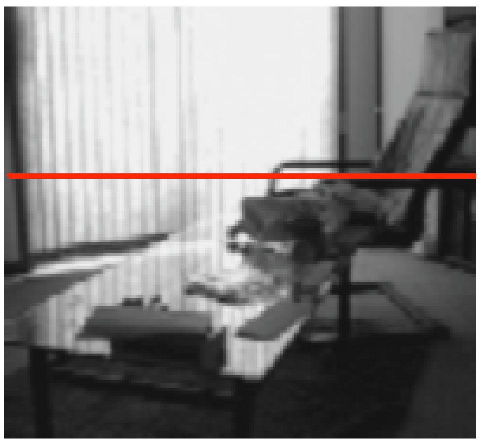
Localizing the velocity to a patch using a windowing function:

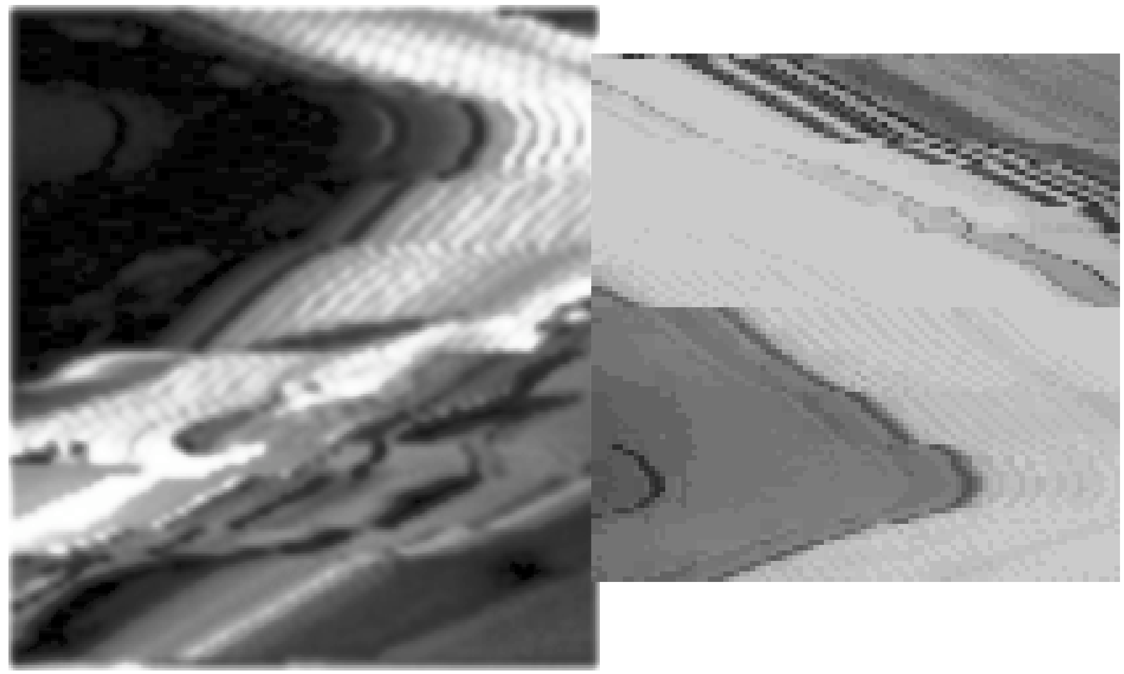
$$\mathcal{F}\{w_{ij}(\vec{x} - \vec{x}_i, t - t_j)I(\vec{x} - \vec{v}_{ij}t, t)\} =$$

$$S(\vec{\omega_x}, \omega_t) = W(\vec{\omega_x}, \omega_t) \otimes (S_I(\vec{\omega_x})\delta([\vec{\omega_x} \ \ \omega_t]^T[\vec{v}_{ij} \ \ 1]))$$

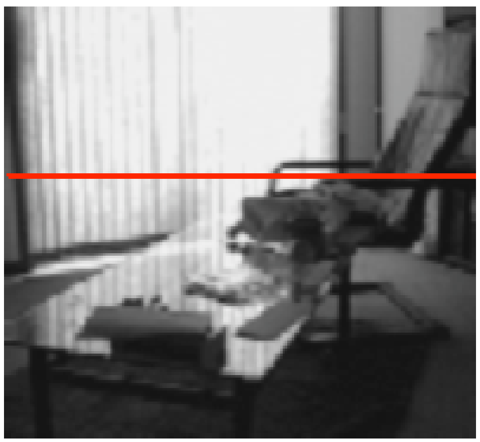
Has the effect of blurring the plane with the transform of the window: A Fourier Pancake.

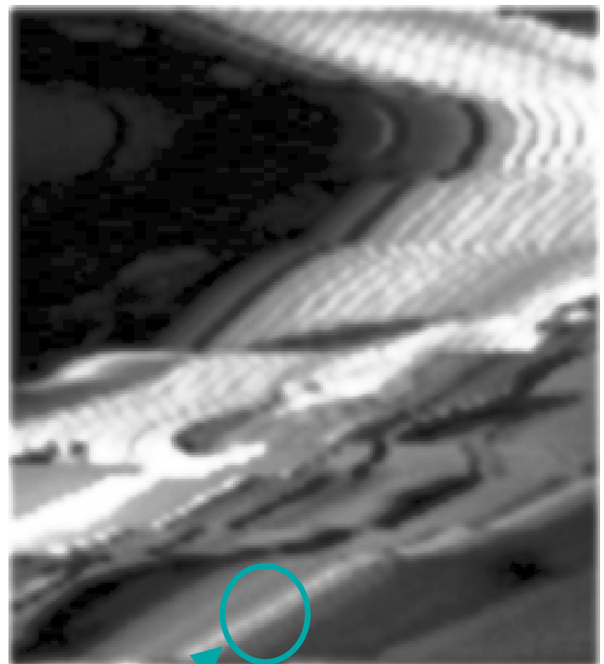
X-T Slice of Translating Camera





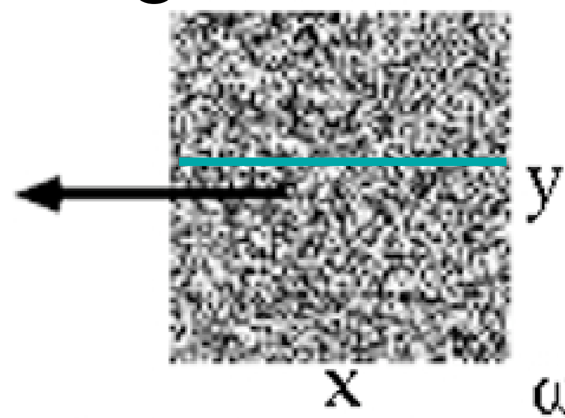
X-T Slice of Translating Camera



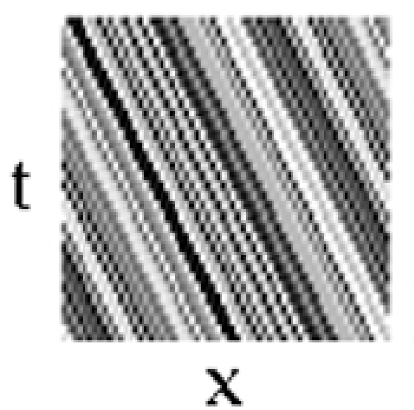


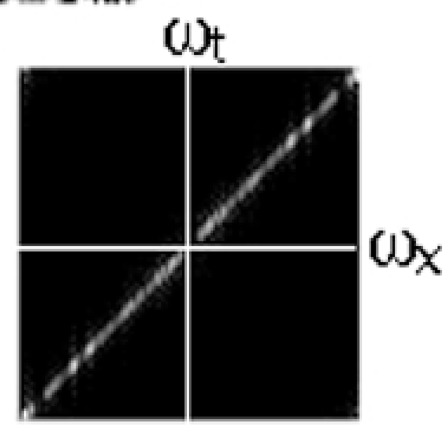
2-D Fourier Analysis

Image translations:



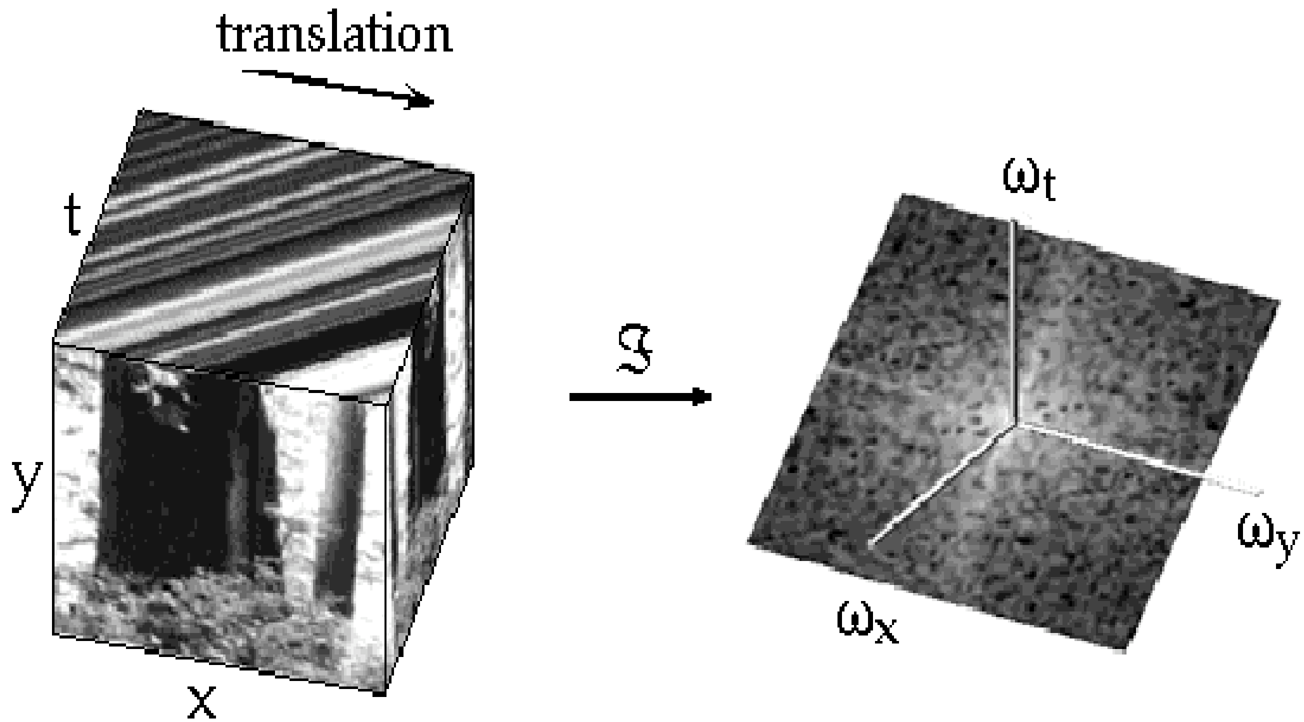
are Oriented in Space-Time



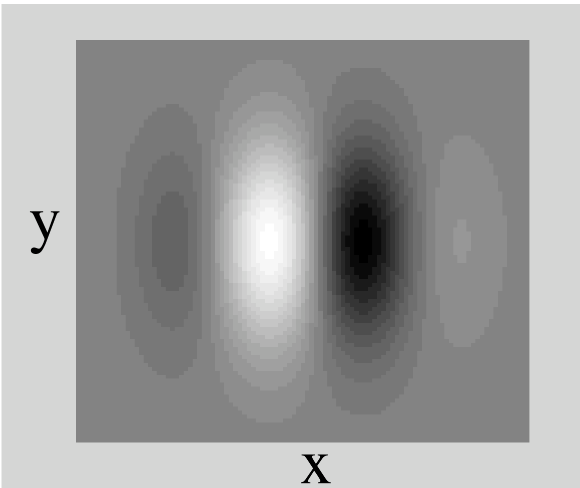


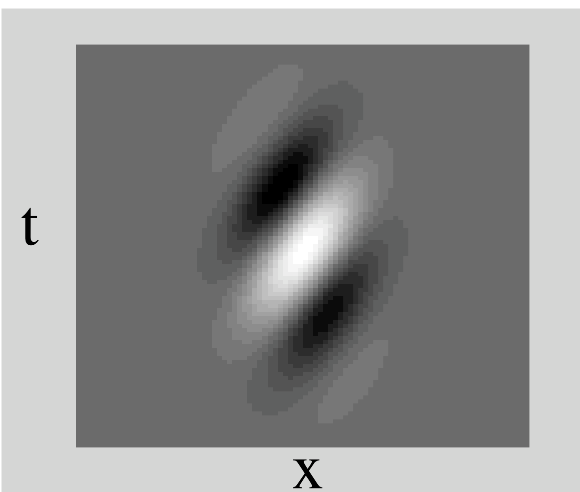
have Power Spectra along a line in Fourier space

3-D Motion Information (x-y-t)

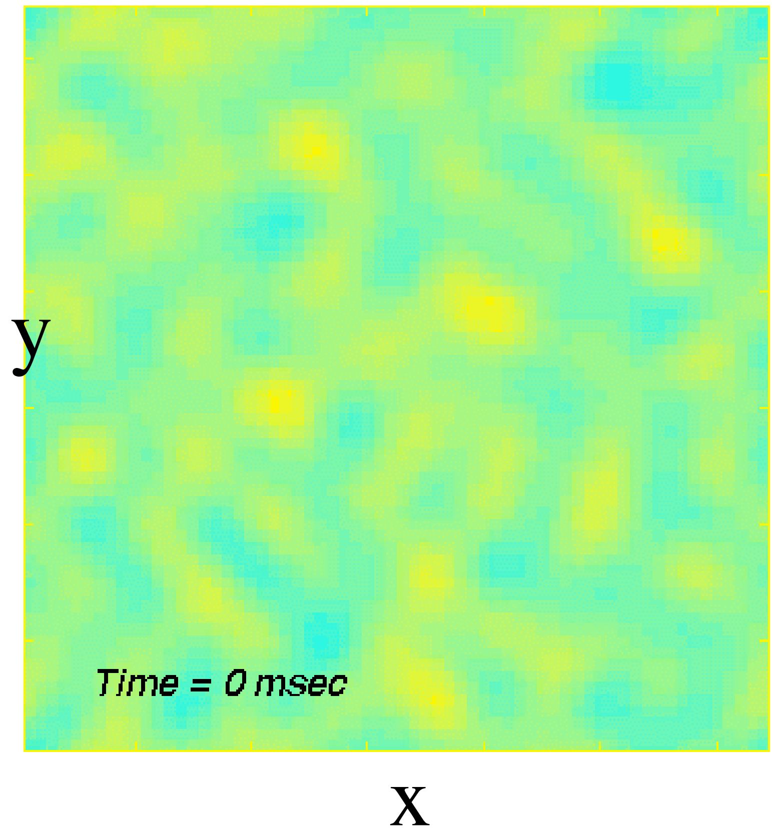


Early Visual Neurons (V1)





Ringach et al (1997)



Information in Translating Images

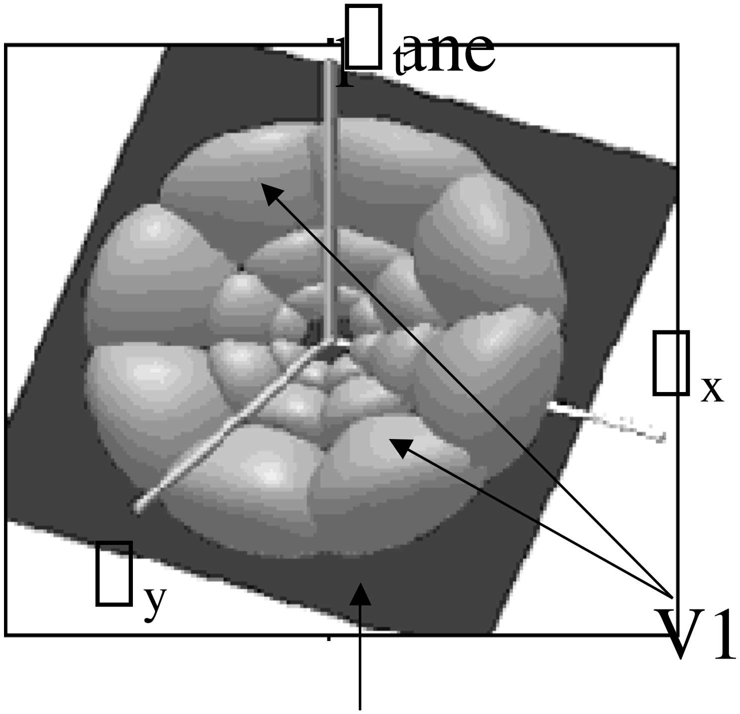
- The power spectral density of a translating image lies on a plane in $(\square_x, \square_y, \square_t)$ space.

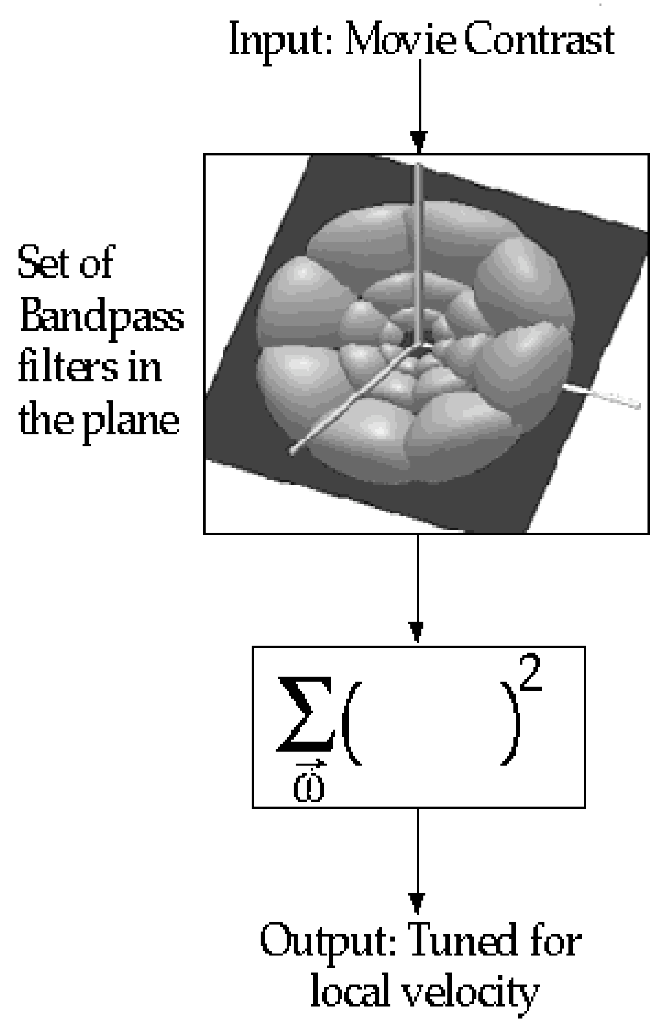
- The orientation of this plane is uniquely determined by the velocity of the translation.

- The amplitudes on the plane are determined by the (spatial) image spectrum.

Solving the ambiguities

Pool neurons tuned to frequencies in a common plane





Limits of the (local) gradient method

1. Fails when intensity structure within window is poor
2. Fails when the displacement is large (typical operating range is motion of 1 pixel per iteration!)
   - *Linearization of brightness is suitable only for small displacements*

Also, brightness is not strictly constant in images

- *actually less problematic than it appears, since we can pre-filter images to make them look similar*

Parametric (Global) Motion Models

Global motion models offer

- more constrained solutions than smoothness (Horn-Schunck)
- integration over a larger area than a translation-only model can accommodate (Lucas-Kanade)

Parametric (Global) Motion Models

2D Models:

(Translation)

Affine

Quadratic

Planar projective transform (Homography)

3D Models:

Instantaneous camera motion models

Homography+epipole

Plane+Parallax

- Transformations/warping of image

$$E(\mathbf{h}) = \sum_{\mathbf{x} \in R} \left[I(\mathbf{x} + \mathbf{h}) - I_0(\mathbf{x}) \right]^2$$

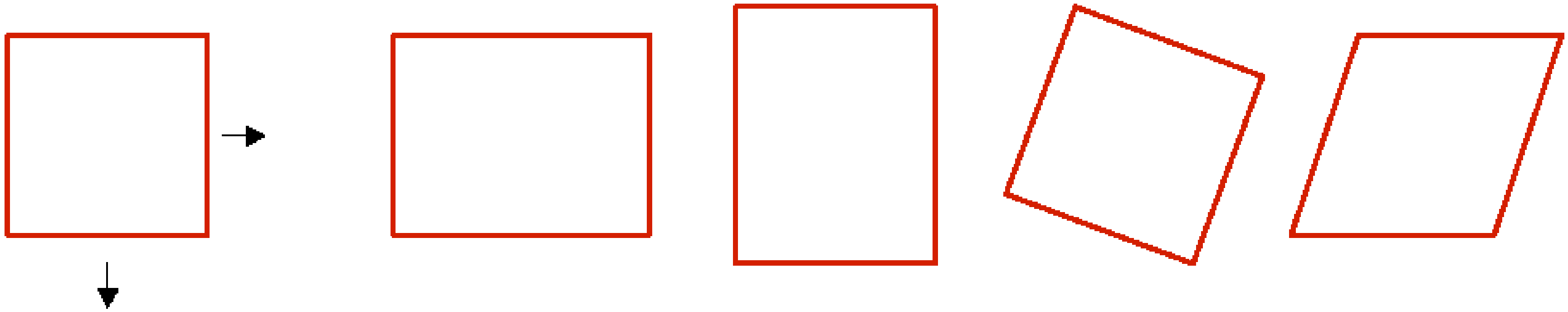
Translations: $\mathbf{h} = \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$

Generalizations

$$E(\mathbf{A},\ \mathbf{h}) = \sum_{\mathbf{x} \in R} \left[I(\mathbf{A}\mathbf{x} + \mathbf{h}) - I_0(\mathbf{x}) \right]^2$$

Affine: $\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \mathbf{h} = \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$

Generalization



Affine: $\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ $\mathbf{h} = \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$

Example: Affine Motion

$$\boxed{\begin{aligned} u(x,y) &= a_1 + a_2 x + a_3 y \\ v(x,y) &= a_4 + a_5 x + a_6 y \end{aligned}}$$

Substituting into the B.C. Equation:

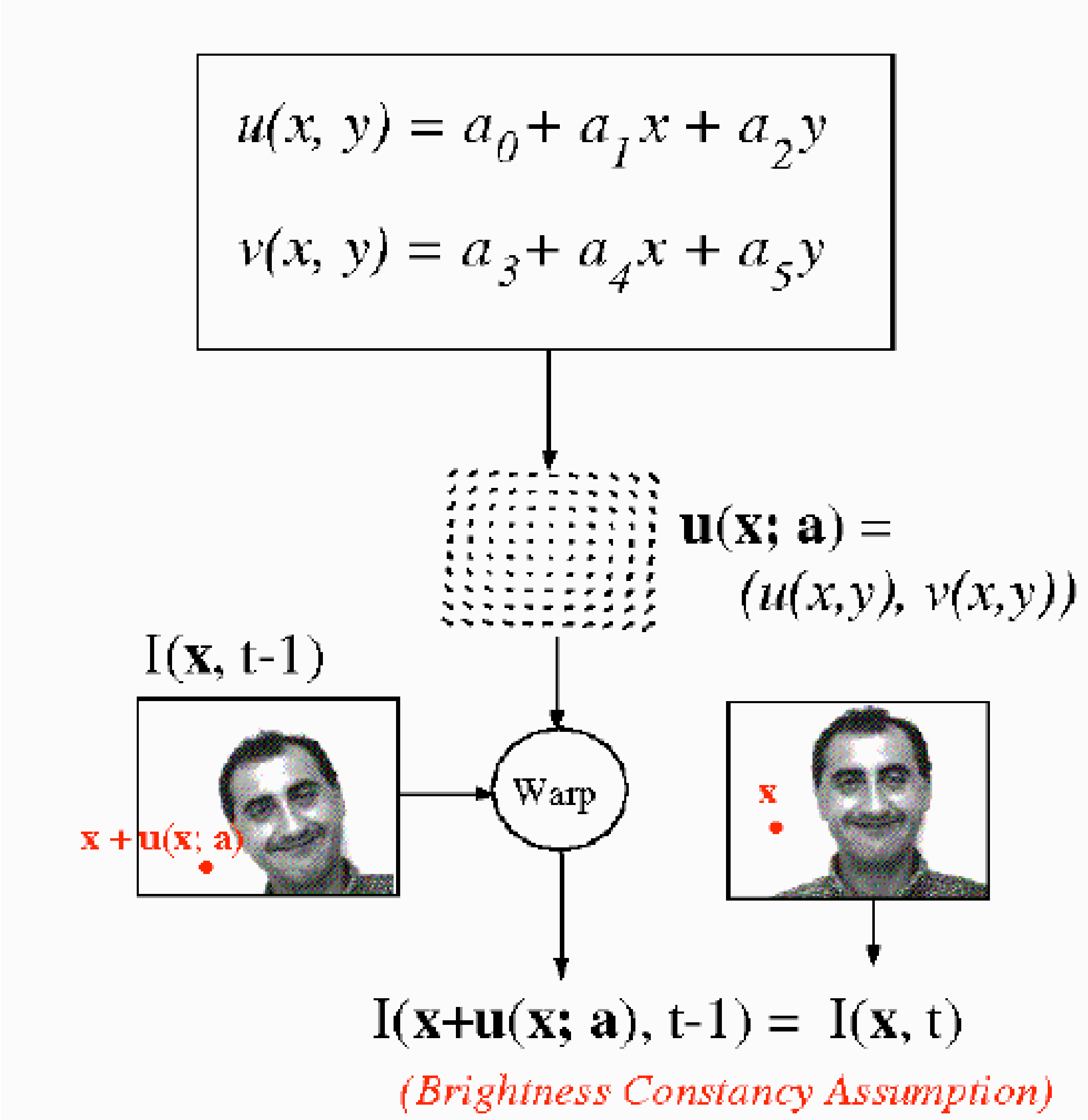
$$I_x \cdot u + I_y \cdot v + I_t \approx 0$$

$$\boxed{I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \approx 0}$$

Each pixel provides 1 linear constraint in 6 *global* unknowns
(minimum 6 pixels necessary)

Least Square Minimization (over all pixels):

$$Err(\vec{a}) = \sum \left[I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \right]$$

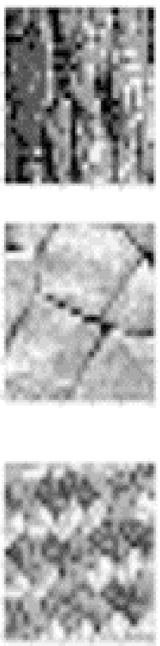


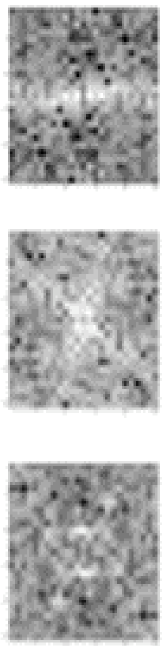
$u(x, y) = a_0 + a_1 x + a_2 y$
$v(x, y) = a_3 + a_4 x + a_5 y$
$\mathbf{u}(\mathbf{x}; \mathbf{a}) = (u(x,y), v(x,y))$
$I(\mathbf{x}, t-1)$
$\mathbf{x} + \mathbf{u}(\mathbf{x}; \mathbf{a})$
Warp
$\mathbf{x}$
$I(\mathbf{x}+\mathbf{u}(\mathbf{x}; \mathbf{a}), t-1) = I(\mathbf{x}, t)$
(Brightness Constancy Assumption)

How to Pool: Adapt or Average?

Natural
Images

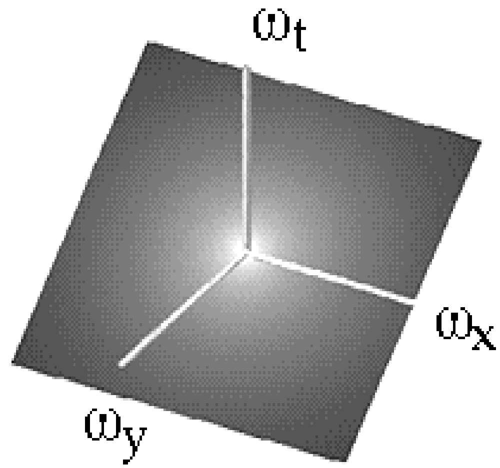
Spatial
Spectra





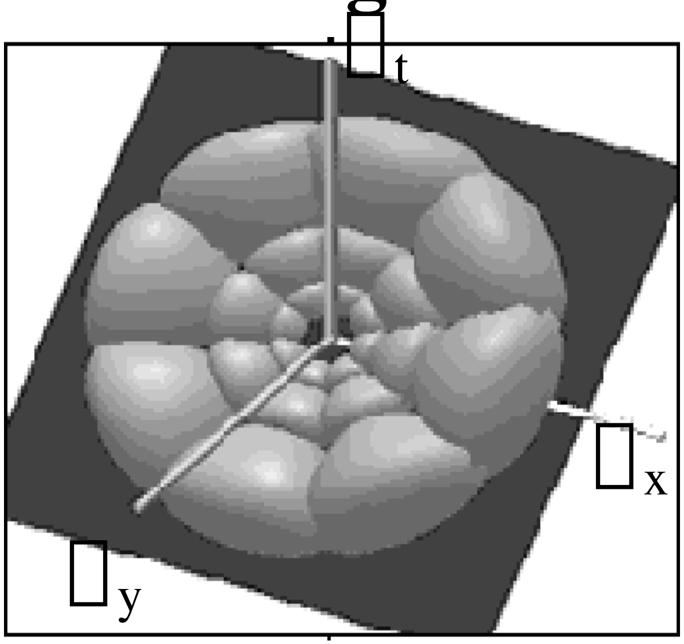
Current local image
spatial form

Average (Expected) spectrum of
an image translation



Model Motion Detectors

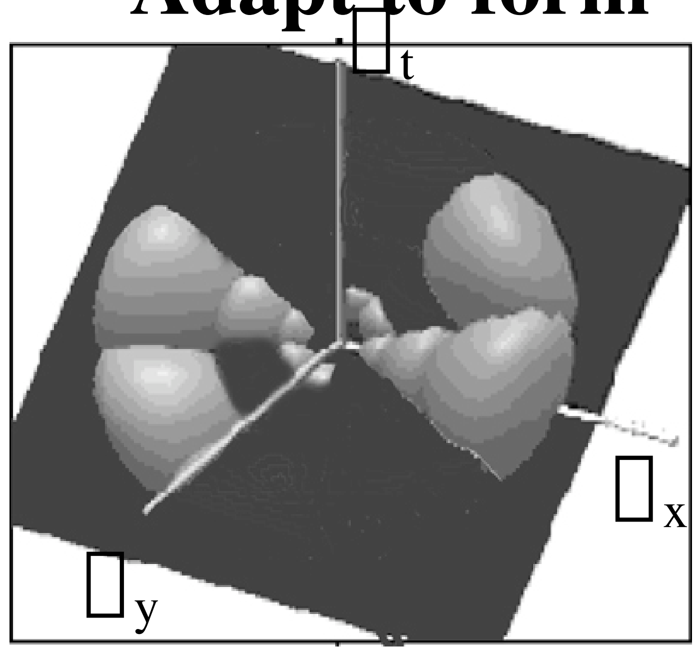
Average form



Average form motion detector
Uniform weights assigned in plan
Optimal when:

- Spatial form uncertainty
- Gaussian noise

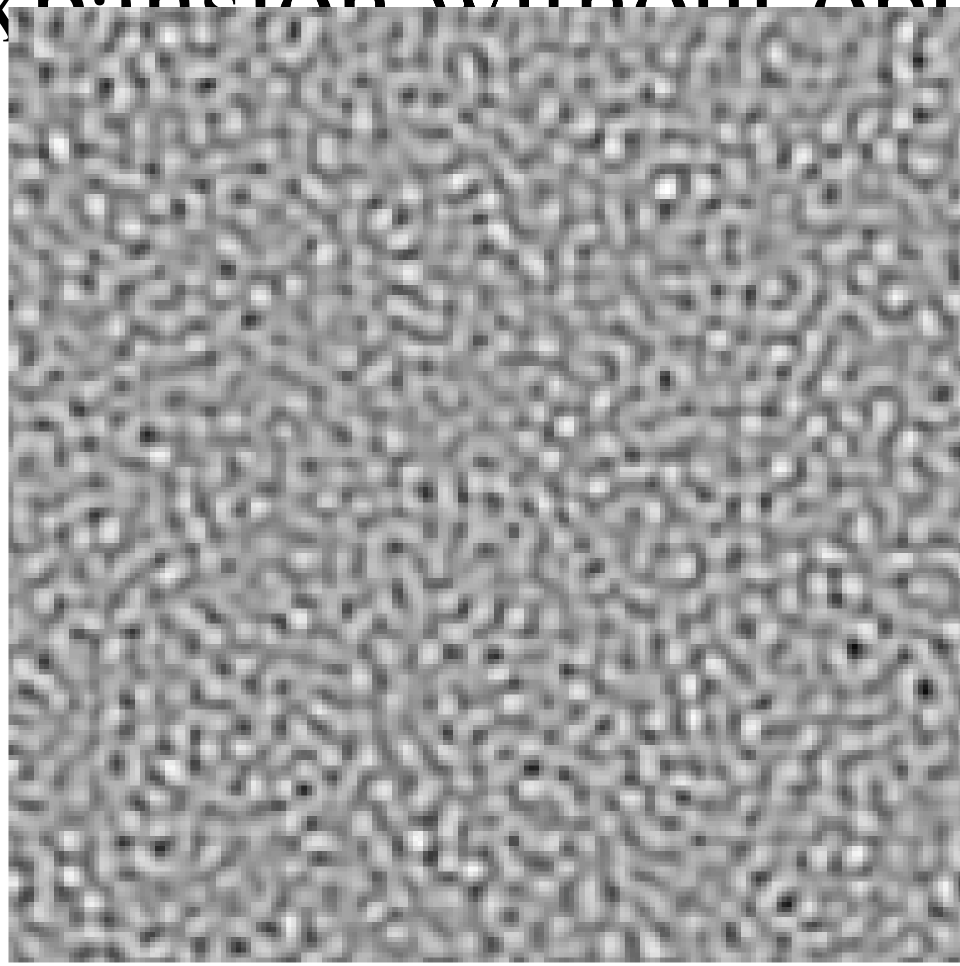
Adapt to form



Adapt to form motion detector
Weighted by spatial spectrum
Optimal when:

- Real-time estimate of spatial form
- Gaussian noise

Expansion without optic flow



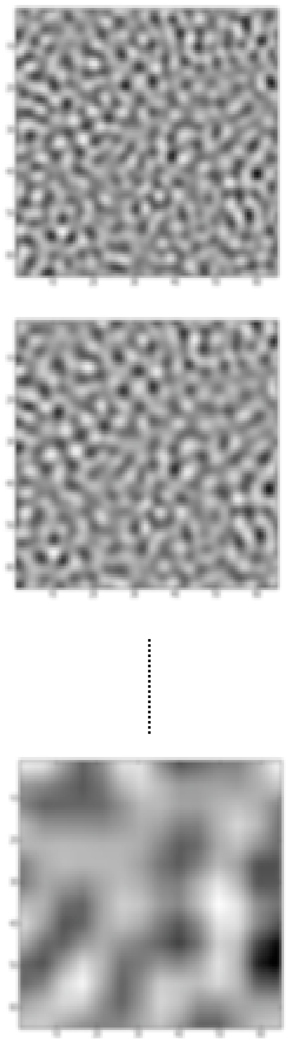
Schrater, et al. *Nature*, 2001

Expanding Texture

t_1

t_2

t_n

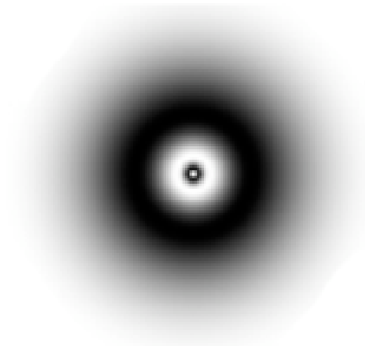


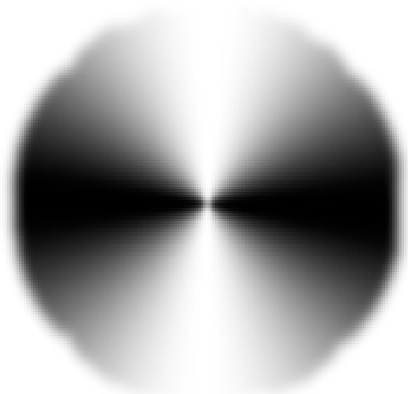
x-t slice

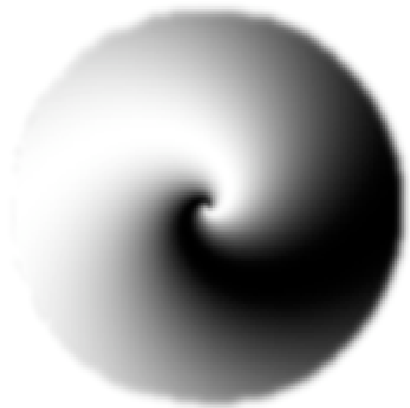
t

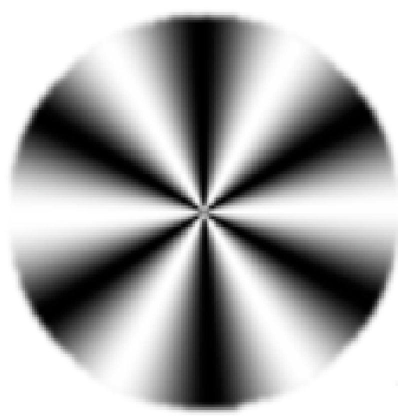
x

Log-Polar Basis Functions



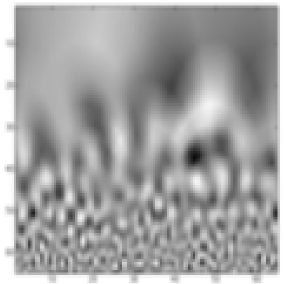




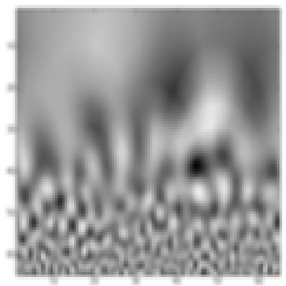


Transform to Log-Polar

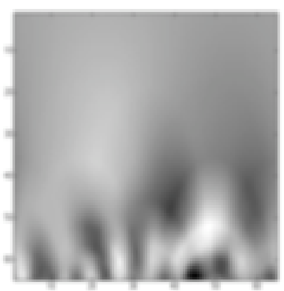
t_1



t_2

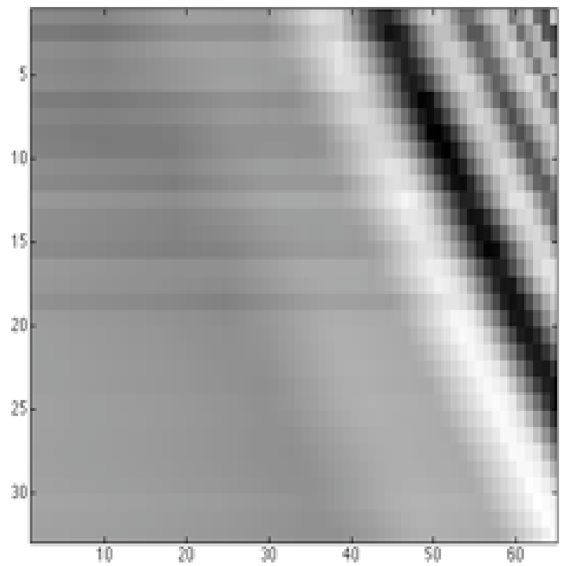


t_n



log(r) -t slice

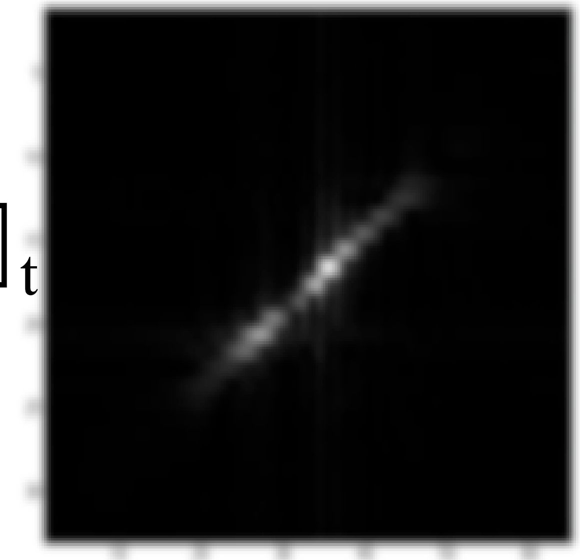
t



log(r)

Fourier domain

□$_t$



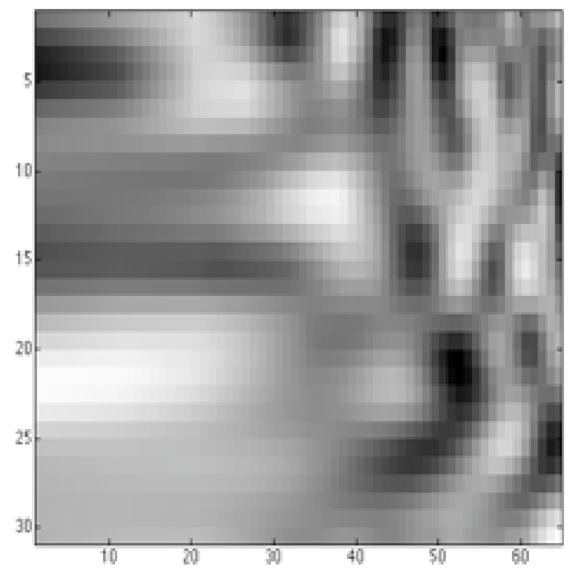
□$_{log(r)}$

Revealing Motion in Noise Movie

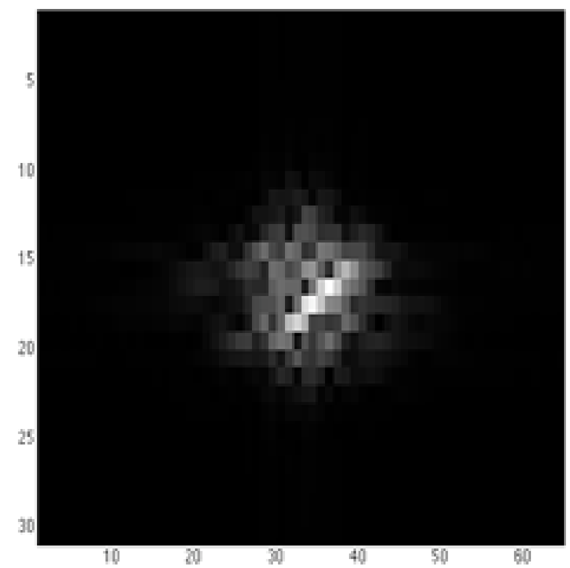
log(r) -t slice

Fourier domain

t



$\square_t$



log(r)

$\square_{log(r)}$