Computer vision for video (Motion analysis)

Translational alignment

Optical flow

Motion analysis

Algorithms for aligning images and estimating motion in video sequences are among the most widely used in computer vision. For example, frame-rate image alignment is widely used in digital cameras to implement their image stabilization (IS) feature.

An early example of a widely used image registration algorithm is the patch-based translational alignment (optical flow) technique developed by Lucas and Kanade (1981). Variants of this algorithm are used in almost all motion-compensated video compression schemes such as MPEG/H.263 (Le Gall 1991) and HEVC/H.265 (Sullivan, Ohm *et al.* 2012). Similar parametric motion estimation algorithms have found a wide variety of applications, including video summarization (Teodosio and Bender 1993; Irani and Anandan 1998), video stabilization (Hansen, Anandan *et al.* 1994; Srinivasan, Chellappa *et al.* 2005; Matsushita, Ofek *et al.* 2006), and video compression (Irani, Hsu, and Anandan 1995; Lee, Chen *et al.* 1997). More sophisticated image registration algorithms have also been developed for medical imaging and remote sensing. Image registration techniques are surveyed by Brown (1992), Zitov'aa and Flusser (2003), Goshtasby (2005), and Szeliski (2006a).

9.1 Translational alignment

The simplest way to establish an alignment between two images or image patches is to shift one image relative to the other. Given a *template* image $I_0(\mathbf{x})$ sampled at discrete pixel locations $\{\mathbf{x}_i = (x_i, y_i)\}$, we wish to find where it is located in image $I_1(\mathbf{x})$. A least squares solution to this problem is to find the minimum of the *sum of squared differences* (SSD) function

$$E_{\mathrm{SSD}}(\mathbf{u}) = \sum_i [I_1(\mathbf{x}_i + \mathbf{u}) - I_0(\mathbf{x}_i)]^2 = \sum_i e_i^2, \tag{9.1}$$

where $\mathbf{u} = (u, v)$ is the *displacement* and $e_i = I_1(\mathbf{x}_i + \mathbf{u}) - I_0(\mathbf{x}_i)$ is called the *residual error* (or the *displaced frame difference* in the video coding literature).[1] (We ignore for the moment the possibility that parts of I_0 may lie outside the boundaries of I_1 or be otherwise not visible.) The assumption that corresponding pixel values remain the same in the two images is often called the *brightness constancy constraint*.[2]

Robust error metrics. We can make the above error metric more robust to outliers by replacing the squared error terms with a robust function $\rho(e_i)$ (Huber 1981; Hampel, Ronchetti *et al.* 1986; Black and Anandan 1996; Stewart 1999) to obtain

$$E_{\mathrm{SRD}}(\mathbf{u}) = \sum_i \rho(I_1(\mathbf{x}_i + \mathbf{u}) - I_0(\mathbf{x}_i)) = \sum_i \rho(e_i). \tag{9.2}$$

The robust norm $\rho(e)$ is a function that grows less quickly than the quadratic penalty associated with least squares. One such function, sometimes used in motion estimation for video coding because of its speed, is the *sum of absolute differences* (SAD) metric[3] or L_1 norm, i.e.,

$$E_{\mathrm{SAD}}(\mathbf{u}) = \sum_i |I_1(\mathbf{x}_i + \mathbf{u}) - I_0(\mathbf{x}_i)| = \sum_i |e_i|. \tag{9.3}$$

However, because this function is not differentiable at the origin, it is not well suited to gradient-descent approaches such as the ones presented in Section 9.1.3.

Spatially varying weights. The error metrics above ignore that fact that for a given alignment, some of the pixels being compared may lie outside the original image boundaries. Furthermore, we may want to partially or completely downweight the contributions of certain pixels. For example, we may want to selectively "erase" some parts of an image from consideration when stitching a mosaic where unwanted foreground objects have been cut out. For applications such as background stabilization, we may want to downweight the middle part of the image, which often contains independently moving objects being tracked by the camera.

All of these tasks can be accomplished by associating a spatially varying per-pixel weight with each of the two images being matched. The error metric then becomes the weighted (or *windowed*) SSD function,

$$E_{\mathrm{WSSD}}(\mathbf{u}) = \sum_i w_0(\mathbf{x}_i) w_1(\mathbf{x}_i + \mathbf{u})[I_1(\mathbf{x}_i + \mathbf{u}) - I_0(\mathbf{x}_i)]^2, \tag{9.5}$$

where the weighting functions w_0 and w_1 are zero outside the image boundaries.

If a large range of potential motions is allowed, the above metric can have a bias towards smaller overlap solutions. To counteract this bias, the windowed SSD score can be divided by the overlap area

$$A = \sum_i w_0(\mathbf{x}_i) w_1(\mathbf{x}_i + \mathbf{u}) \tag{9.6}$$

to compute a *per-pixel* (or mean) squared pixel error E_{WSSD}/A. The square root of this quantity is the *root mean square* intensity error

$$RMS = \sqrt{E_{\mathrm{WSSD}}/A} \tag{9.7}$$

often reported in comparative studies.

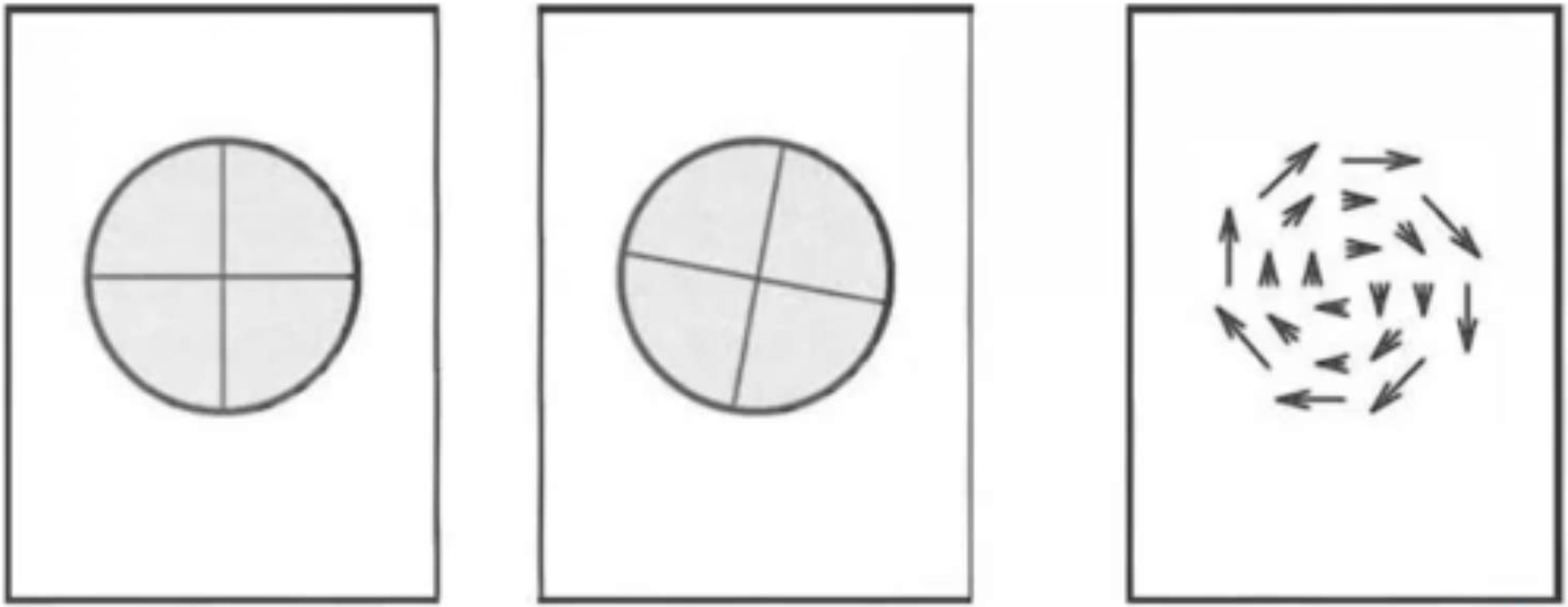
Optical Flow

- The most general (and challenging) version of motion estimation is to compute an independent estimate of motion at each pixel, which is generally known as optical (or optic) flow.

OPTICAL FLOW = apparent motion of brightness patterns

Ideally, the optical flow is the projection of the three-dimensional velocity vectors on the image

Basic idea

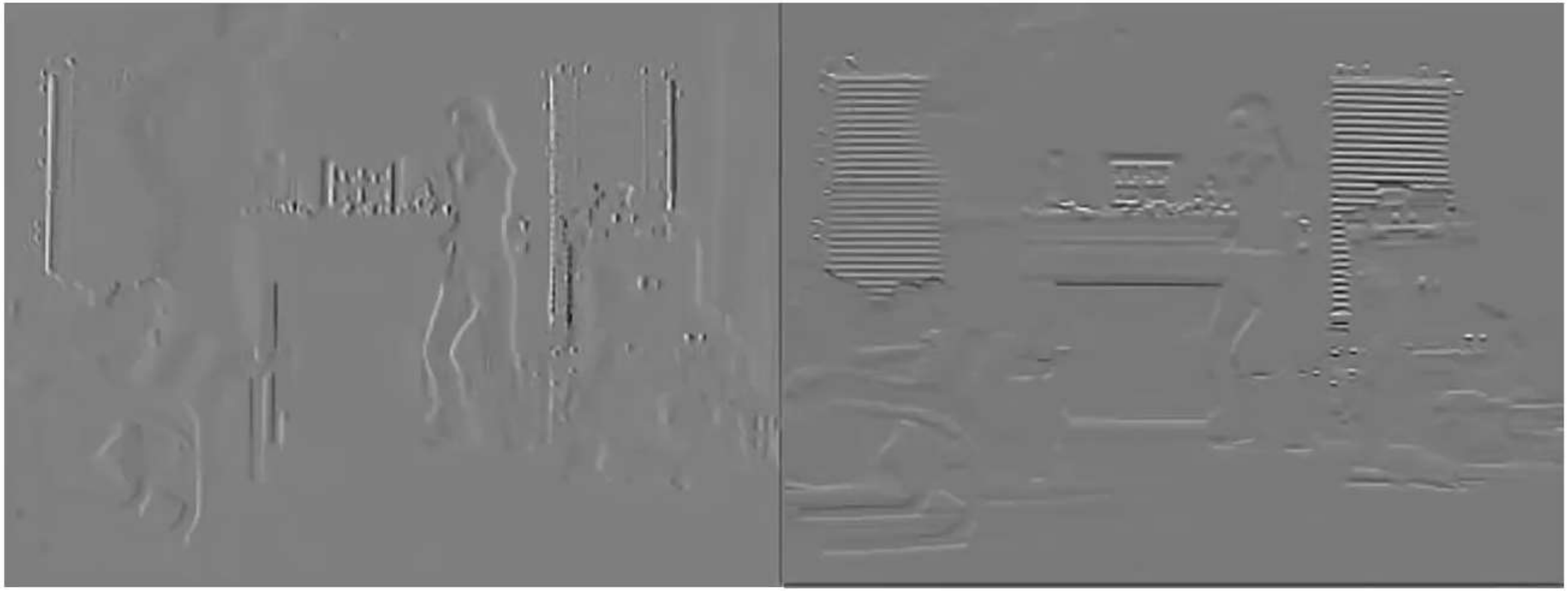


Basic idea

- Optical flow computation is based on two assumptions:
  1. The observed brightness of any object point is constant over time
  2. Nearby points in the image plane move in a similar manner (the velocity smoothness constraint).

Basic idea

Basic idea



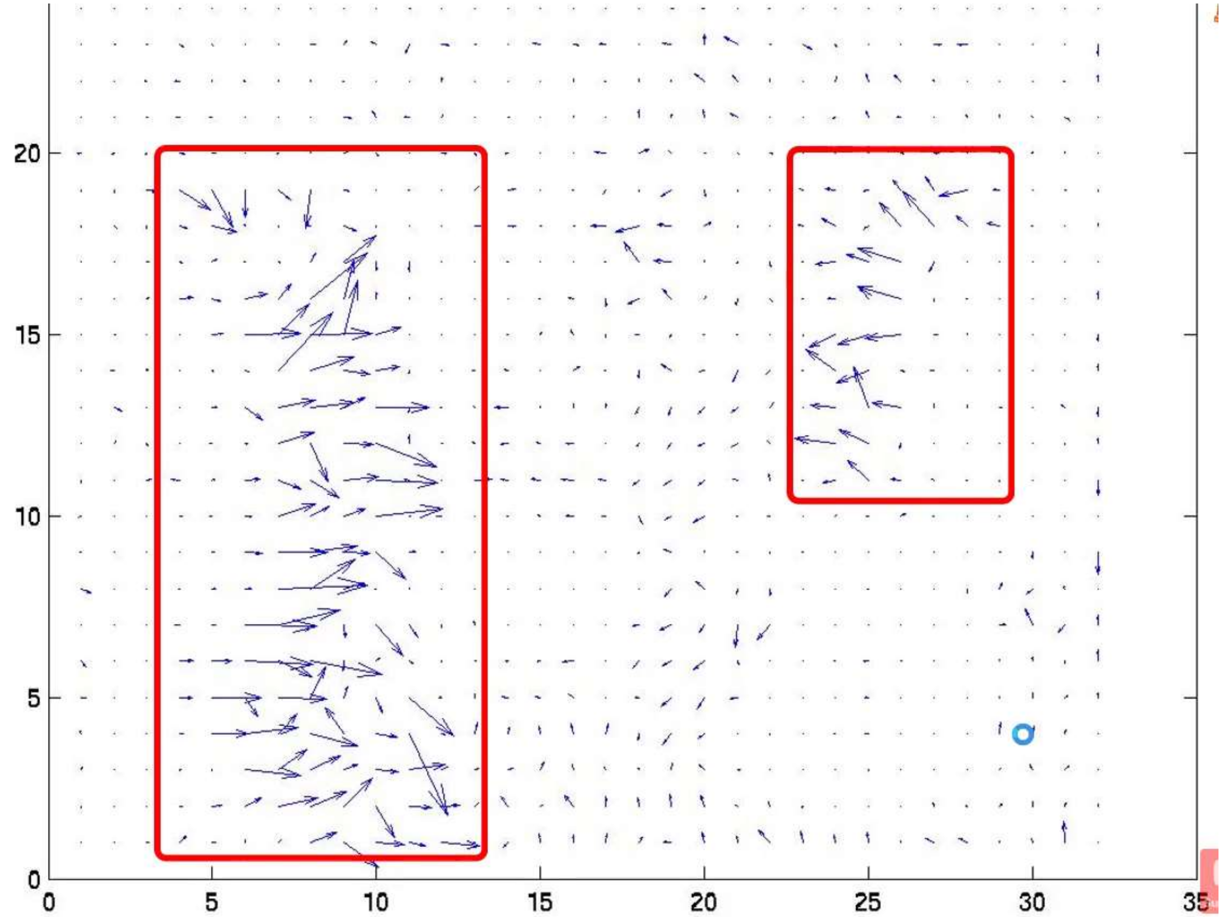
The partial spatial derivative in the ***x*** direction and the partial spatial derivative in the ***y*** direction are shown here

Basic idea

the partial derivative in time direction is shown here

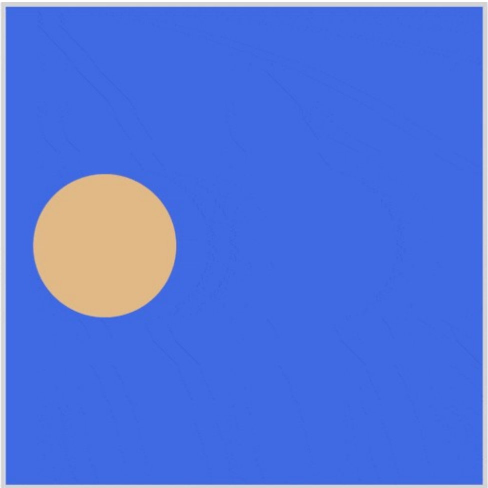
Basic idea

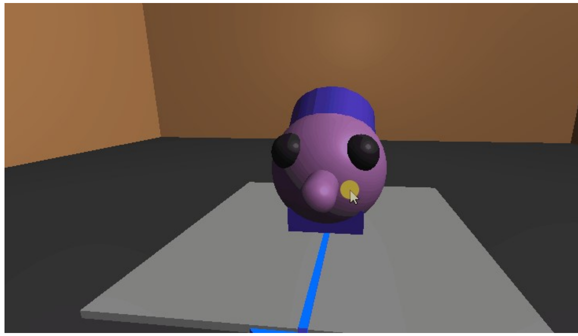
Lucas and Kanade



Basic idea

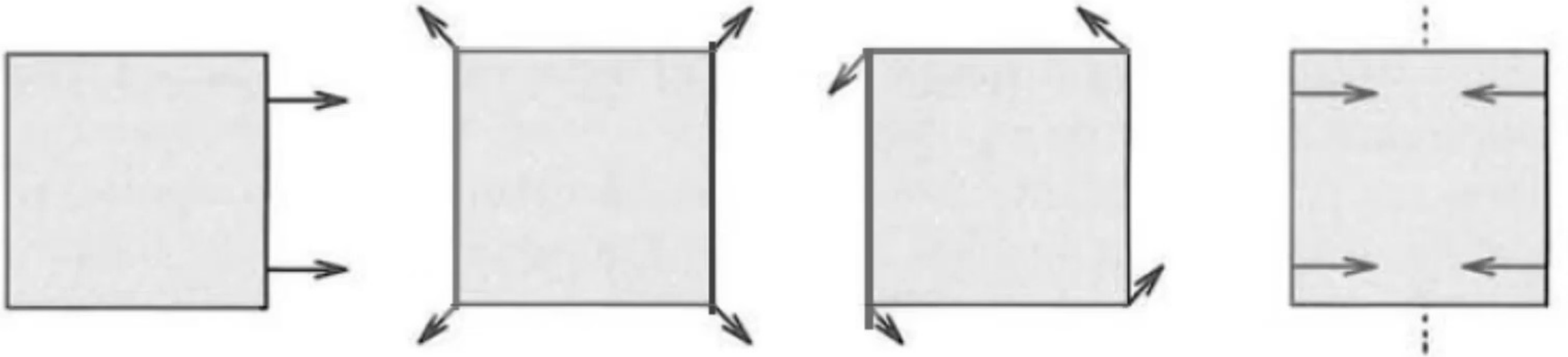
- Motion, as it appears in dynamic images, is usually some combination of four basic elements:
  - Translation at constant distance from the observer.
  - Translation in depth relative to the observer.
  - Rotation at constant distance about the view axis.
  - Rotation of a planar object perpendicular to the view axis.



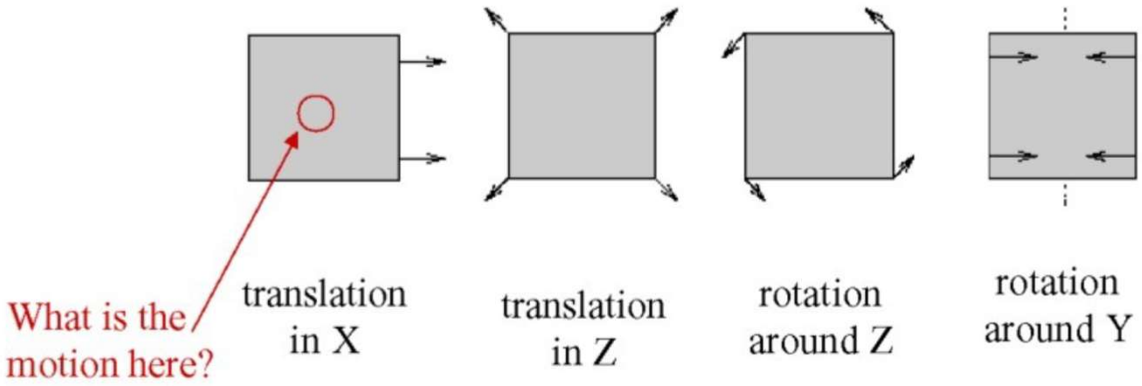
Basic idea

- Motion, as it appears in dynamic images, is usually some combination of four basic elements:
  - Translation at constant distance from the observer.
  - Translation in depth relative to the observer.
  - Rotation at constant distance about the view axis.
  - Rotation of a planar object perpendicular to the view axis.



Planer motion example

- Ideal motion of a plane



Scene Flow: $\longrightarrow$
Normal Flow: undef
Optic Flow: ?, probably 0

Optical Flow

Problem Definition

Given two consecutive image frames,
estimate the motion of each pixel

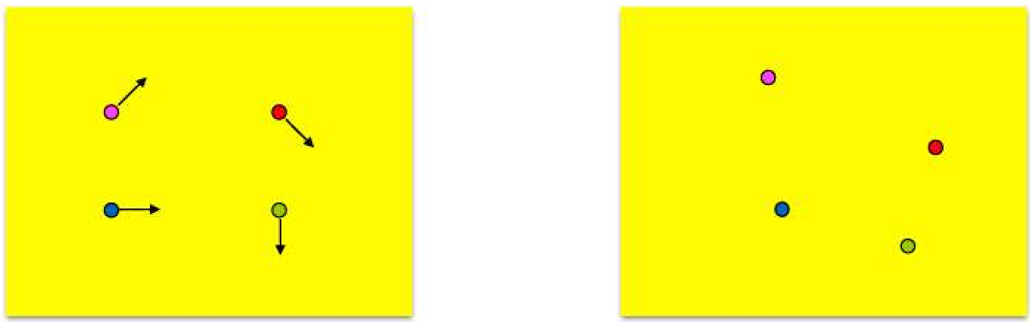
Assumptions

Brightness constancy

Small motion

Optical Flow

(Problem definition)



$I(x, y, t)$ $\qquad$ $I(x, y, t')$

Estimate the motion
(flow) between these
two consecutive images

How is this different from estimating a 2D transform?

Optical Flow

Key Assumptions

(unique to optical flow)

Color Constancy

(Brightness constancy for intensity images)

Implication: allows for pixel to pixel comparison
(not image features)

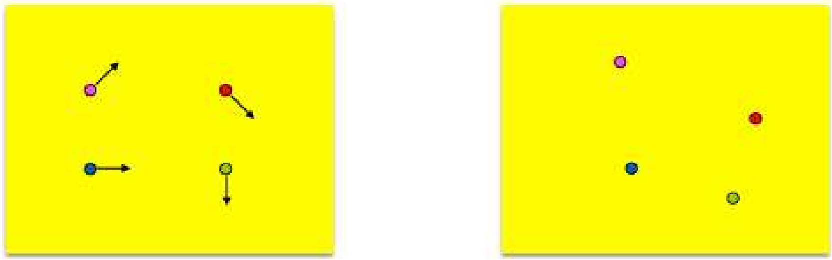
Small Motion

(pixels only move a little bit)

Implication: linearization of the brightness
constancy constraint

Optical Flow

Approach



$I(x, y, t)$ $\quad$ $I(x, y, t')$

Look for nearby pixels with the same color

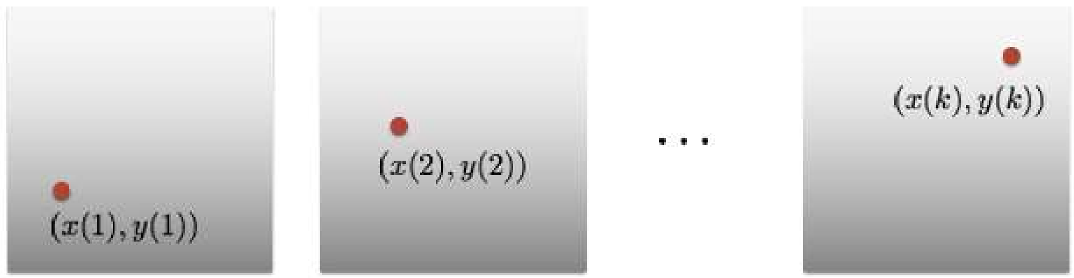
(small motion) (color constancy)

Optical Flow

Assumption 1

Brightness constancy

Scene point moving through image sequence

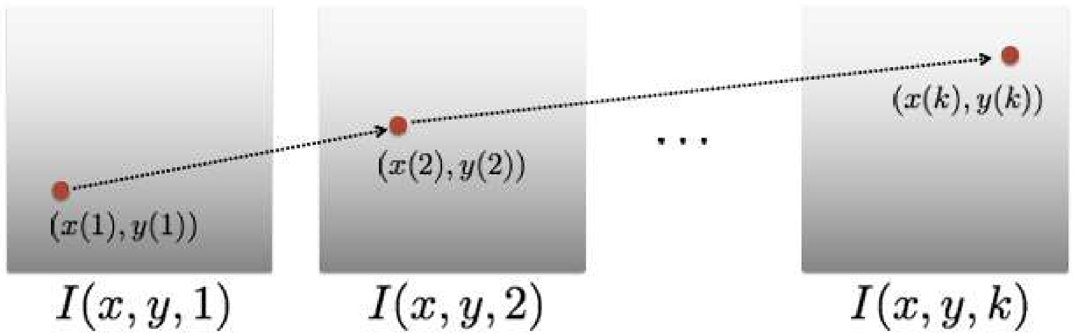


Optical Flow

Assumption 1

Brightness constancy

Scene point moving through image sequence



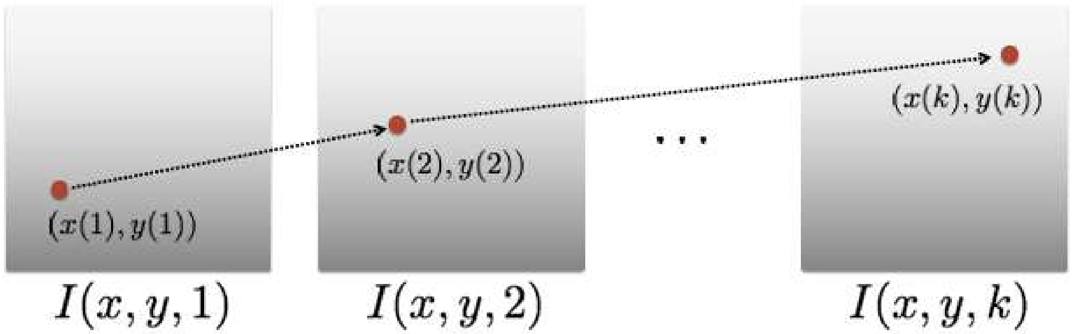
Assumption:Brightness of the point will remain the same

Optical Flow

Assumption 1

Brightness constancy

Scene point moving through image sequence



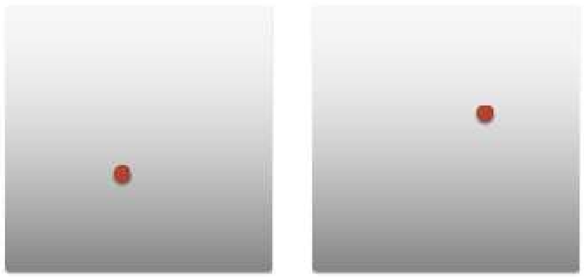
Assumption:Brightness of the point will remain the same

$$I(x(t), y(t), t) = C$$

Optical Flow

Assumption 2

Small motion

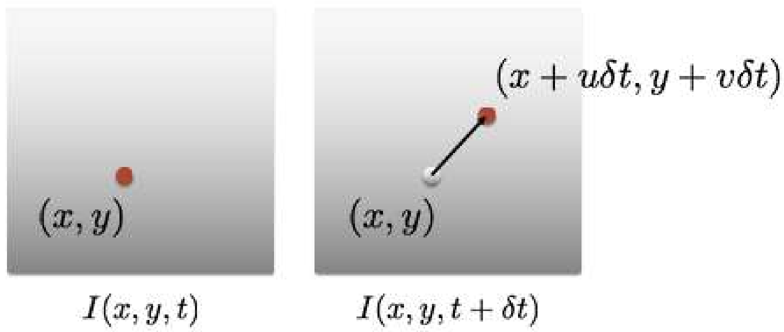


$I(x, y, t)$ $\quad$ $I(x, y, t + \delta t)$

Optical Flow

Assumption 2

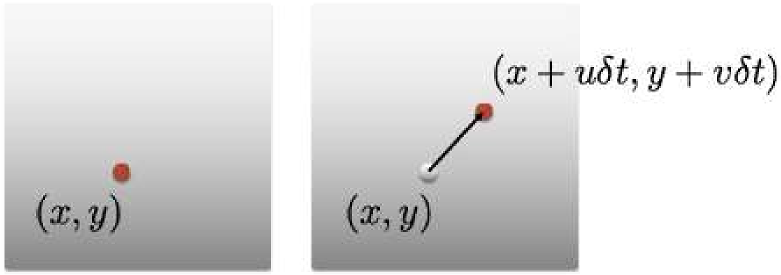
Small motion



Optical Flow

Assumption 2

Small motion



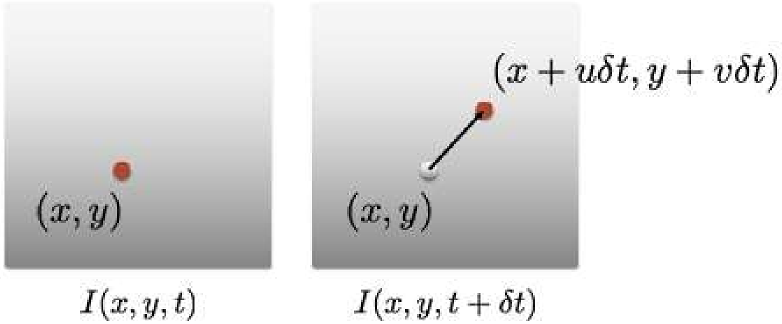
$I(x, y, t)$ $\qquad$ $I(x, y, t + \delta t)$

Optical flow (velocities): (u, v) $\qquad$ Displacement: $(\delta x, \delta y) = (u\delta t, v\delta t)$

Optical Flow

Assumption 2

Small motion



Optical flow (velocities): (u, v) Displacement: $(\delta x, \delta y) = (u\delta t, v\delta t)$

For a ***really small space-time step***…

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

… the brightness between two consecutive image frames is the same

These assumptions yield the …

Brightness Constancy Equation

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

total derivative

partial derivative

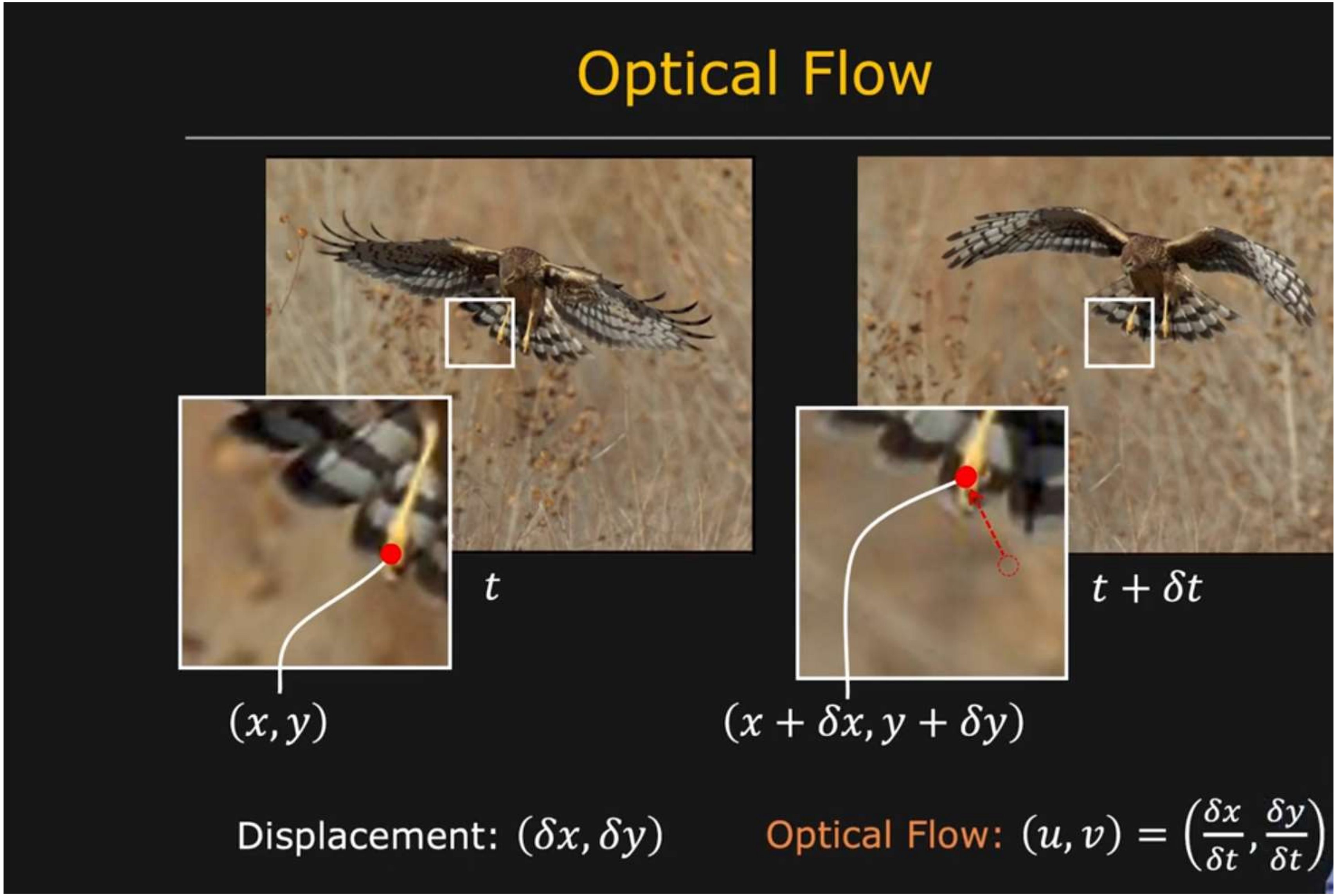
Equation is not obvious. Where does this come from?

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

For small space-time step, brightness of a point is the same

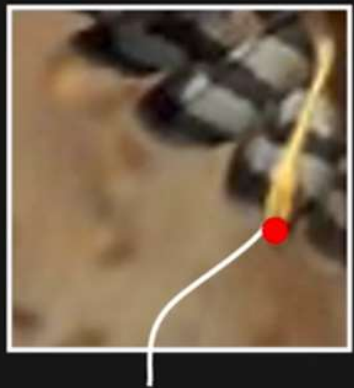
Insight:

If the time step is really small,
we can *linearize* the intensity function

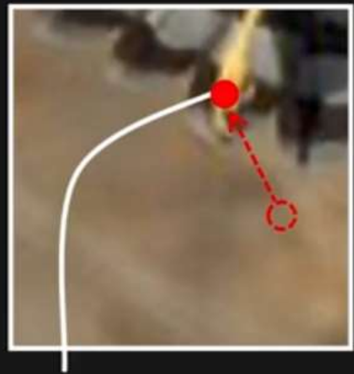


Optical Flow
t
$t + \delta t$
(x, y)
$(x + \delta x, y + \delta y)$
Displacement: $(\delta x, \delta y)$
Optical Flow: $(u, v) = \left(\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}\right)$

Optical Flow Constraint Equation



$I(x, y, t)$



$I(x + \delta x, y + \delta y, t + \delta t)$

Assumption #2:

Displacement $(\delta x, \delta y)$ and time step δt are small

Taylor Series Expansion

Expand a function as an infinite sum of its derivatives

$$f(x+\delta x) = f(x) + \frac{\partial f}{\partial x}\delta x + \frac{\partial^2 f}{\partial x^2}\frac{\delta x^2}{2!} + \cdots + \frac{\partial^n f}{\partial x^n}\frac{\delta x^n}{n!}$$

If δx is small:

$$f(x+\delta x) = f(x) + \frac{\partial f}{\partial x}\delta x + \boxed{O(\delta x^2)} \rightarrow \text{Almost Zero}$$

For a function of three variables with small $\delta x, \delta y, \delta t$:

$$f(x+\delta x, y+\delta y, t+\delta t) \approx f(x,y,t) + \frac{\partial f}{\partial x}\delta x + \frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial t}\delta t$$

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

Multivariable Taylor Series Expansion

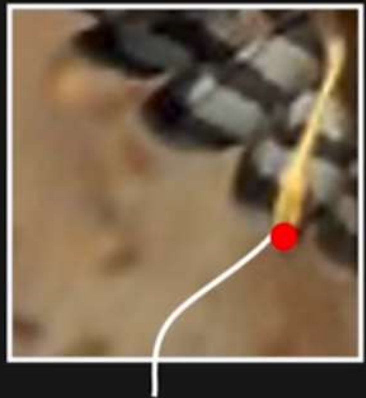
(First order approximation, two variables)

$$f(x, y) \approx f(a, b) + f_x(a, b)(x - a) - f_y(a, b)(y - b)$$

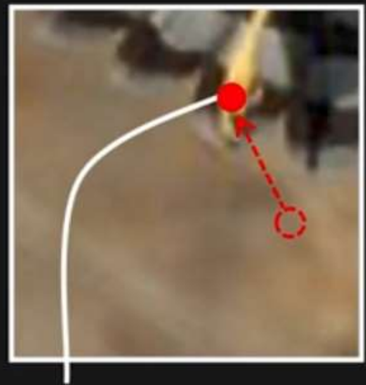
$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t)$$

assuming small motion

Optical Flow Constraint Equation



$I(x, y, t)$



$I(x + \delta x, y + \delta y, t + \delta t)$

Assumption #2:

Displacement $(\delta x, \delta y)$ and time step δt are small

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t$$

$$\boxed{I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + I_x \delta x + I_y \delta y + I_t \delta t}$$

Optical Flow Constraint Equation

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) \qquad \text{------------ (1)}$$

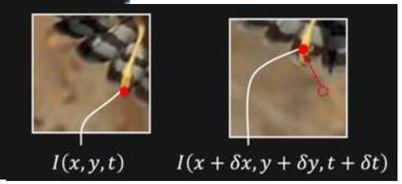
$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + I_x\delta x + I_y \delta y + I_t \delta t \qquad \text{------------ (2)}$$

Subtract (1) from (2): $I_x\delta x + I_y \delta y + I_t \delta t = 0$

Divide by δt and take limit as $\delta t \to 0$: $I_x \dfrac{\partial x}{\partial t} + I_y \dfrac{\partial y}{\partial t} + I_t = 0$

Constraint Equation: $\boxed{I_x u + I_y v + I_t = 0}$ (u,v): Optical Flow

(I_x, I_y, I_t) can be easily computed from two frames



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

Brightness Constancy Equation

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

(x-flow) (y-flow)

shorthand notation

$$\nabla I^\top \boldsymbol{v} + I_t = 0$$

(1 x 2) (2 x 1)

vector form

(putting the math aside for a second…)

What do the terms of the brightness constancy equation represent?

flow velocities

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

Image gradients
(at a point p)

temporal gradient

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

How do you compute …

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$$

spatial derivative

Forward difference
Sobel filter
Derivative-of-Gaussian filter
…

$$I_t = \frac{\partial I}{\partial t}$$

temporal derivative

frame differencing

Frame differencing

t

1	1	1	1	1
1	1	1	1	1
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10

-

$t+1$

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	10	10	10
1	1	10	10	10
1	1	10	10	10

=

$$I_t = \frac{\partial I}{\partial t}$$

0	0	0	0	0
0	0	0	0	0
0	9	9	9	9
0	9	0	0	0
0	9	0	0	0
0	9	0	0	0

(example of a forward difference)

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

Example:

t

3 x 3 patch

1	1	1	1	1
1	1	1	1	1
1	**10**	**10**	**10**	**10**
1	**10**	**10**	**10**	**10**
1	**10**	**10**	**10**	**10**
1	**10**	**10**	**10**	**10**

$t+1$

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	**10**	**10**	**10**
1	1	**10**	**10**	**10**
1	1	**10**	**10**	**10**

$$I_x = \frac{\partial I}{\partial x}$$

-	0	0	0	-
-	0	0	0	-
-	**9**	0	0	-
-	**9**	0	0	-
-	**9**	0	0	-
-	**9**	0	0	-

$$I_y = \frac{\partial I}{\partial y}$$

-	-	-	-	-
0	0	0	0	0
0	**9**	**9**	**9**	**9**
-	0	0	0	0
0	0	0	0	0
-	-	-	-	-

-1

$$I_t = \frac{\partial I}{\partial t}$$

0	0	0	0	0
0	0	0	0	0
0	**9**	**9**	**9**	**9**
0	**9**	0	0	0
0	**9**	0	0	0
0	**9**	0	0	0

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

How do you compute ...

spatial derivative	optical flow	temporal derivative
$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$	$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$	$I_t = \frac{\partial I}{\partial t}$

Forward difference
Sobel filter
Derivative-of-Gaussian filter
...

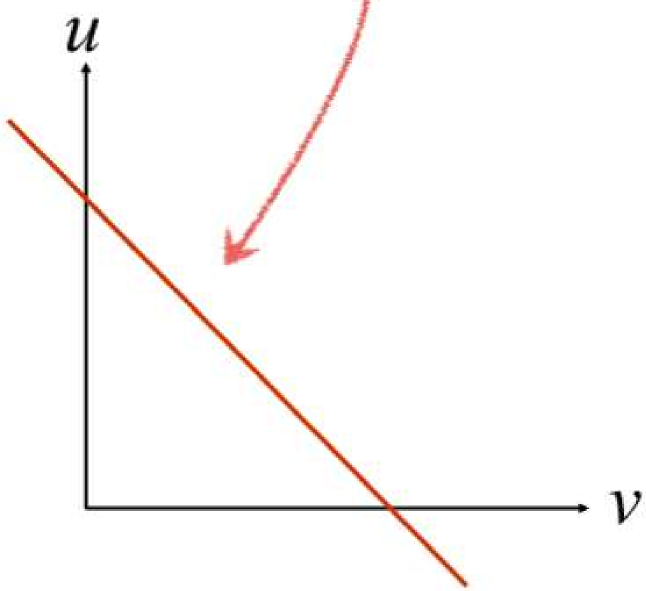
How do you compute this?

frame differencing

Solution lies on a straight line

$$\boxed{I_x u + I_y v + I_t = 0}$$

many combinations of u and v will satisfy the equality



The solution cannot be determined uniquely with a single constraint (a single pixel)

The Aperture Problem

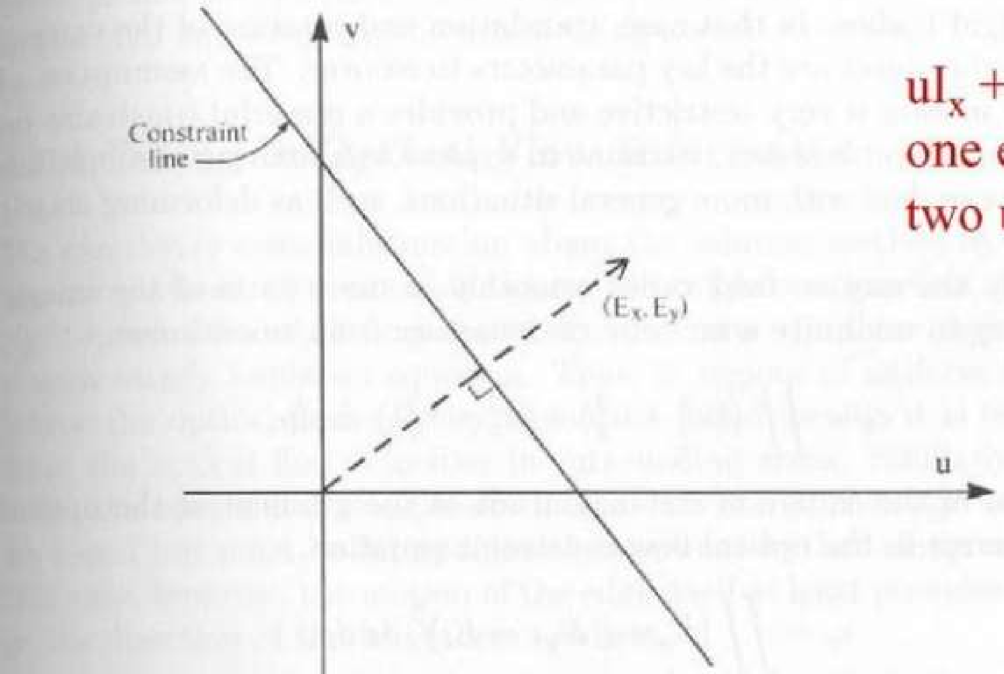
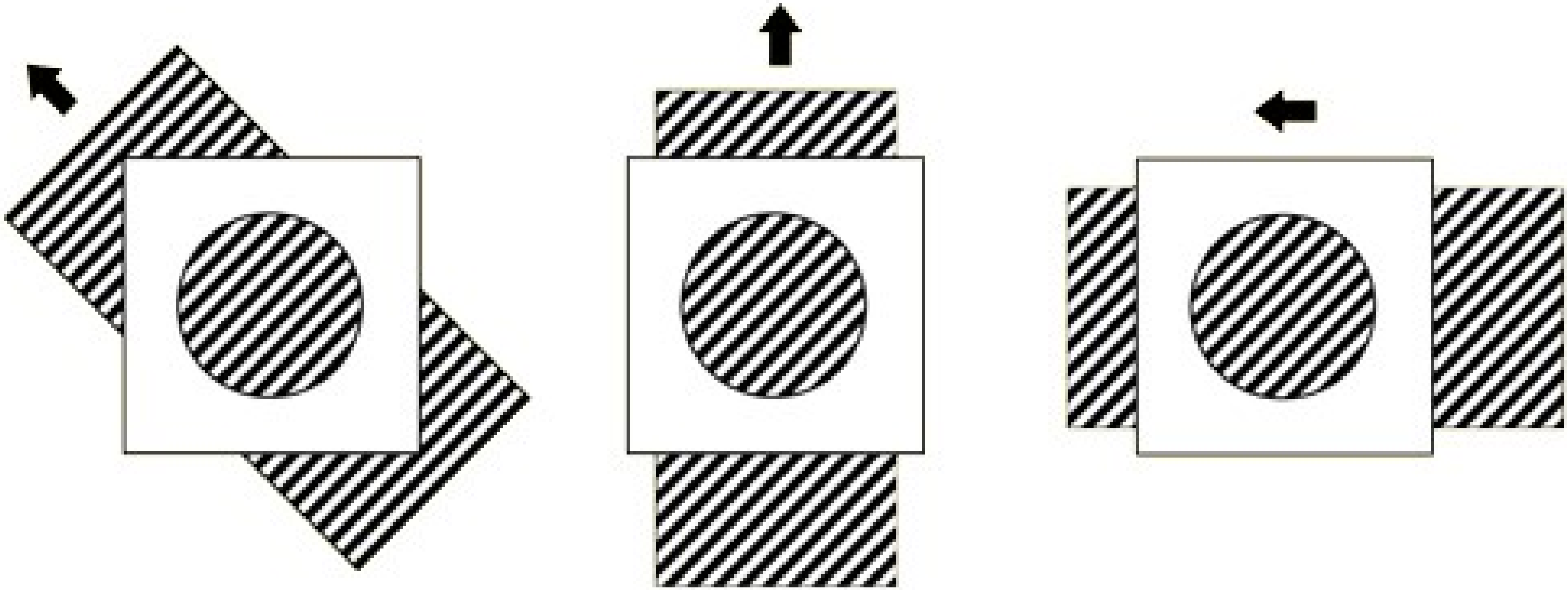


Figure 12-4. Local information on the brightness gradient and the rate of change of brightness with time provides only one constraint on the components of the optical flow vector. The flow velocity has to lie along a straight line perpendicular to the direction of the brightness gradient. We can only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

$$uI_x + vI_y + I_t = 0$$

one equation, two unknowns

Aperture problem and normal flow



unknown

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

known

We need at least ____ equations to solve for 2 unknowns.

Key idea

(of Horn-Schunck optical flow)

Enforce
brightness constancy

Enforce
smooth flow field

to compute optical flow

Enforce **brightness constancy**

$$\boxed{I_x u + I_y v + I_t = 0}$$

For every pixel,

$$\min_{u,v} \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2$$

lazy notation for $I_x(i,j)$

Key idea

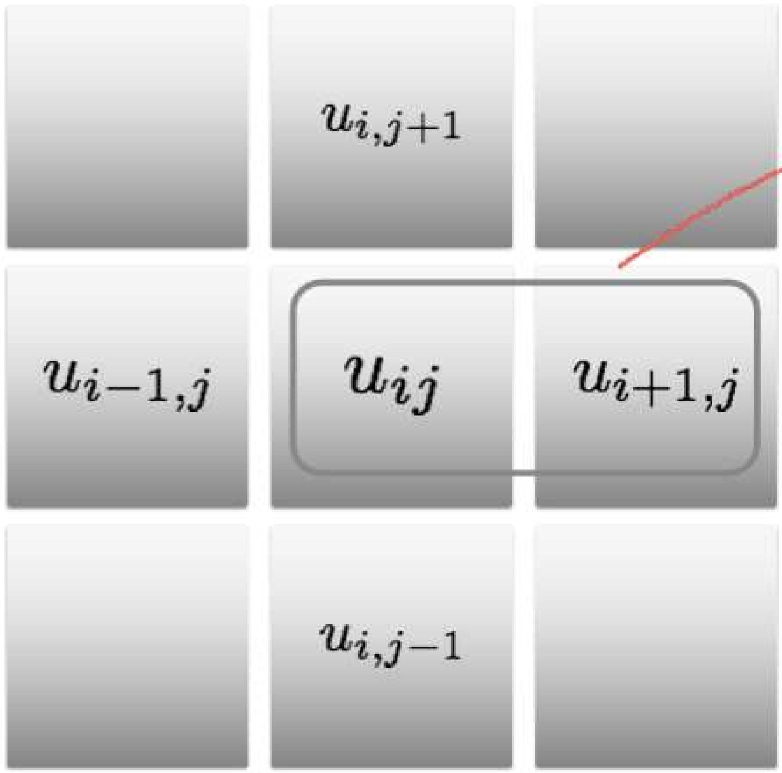
(of Horn-Schunck optical flow)

Enforce
brightness constancy

Enforce
smooth flow field

to compute optical flow

Enforce **smooth flow field**



u-component of flow

$$\min_{\boldsymbol{u}} (u_{i,j} - u_{i+1,j})^2$$

Q&A