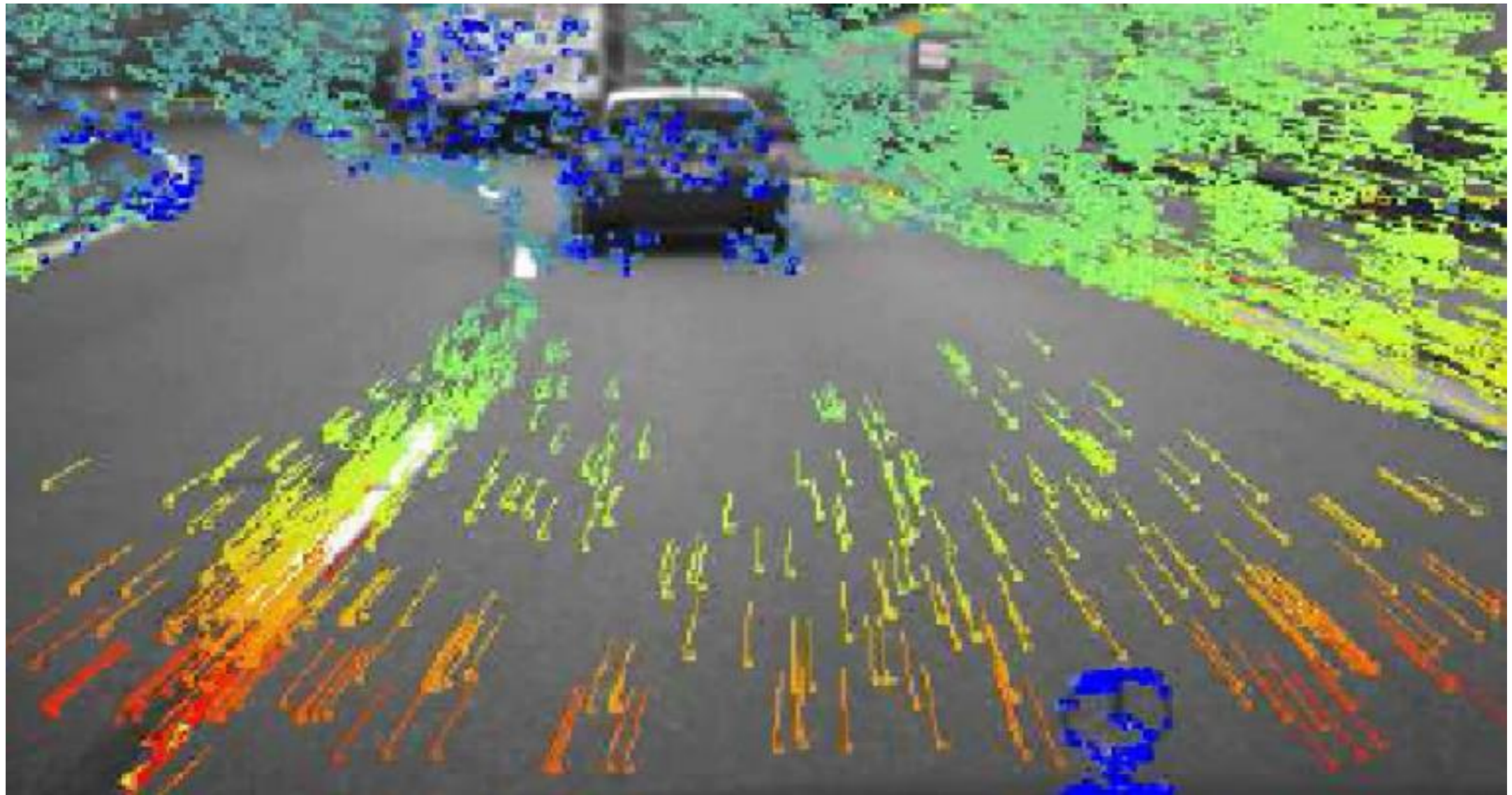


Optical flow



Course announcements

- Programming assignment 6 has been posted and is due on April 22th.
 - Any questions about the homework?
 - How many of you have looked at/started/finished homework 6?
 - Shorter than last year, but **still requires significant time for training.**
- Take-home quiz 10 has been posted and is due on April 19th.
- Prakhar has extra office hours today between 5-6 pm.
- Please see Piazza poll about an extra take-home quiz 12:
<https://piazza.com/class/k53x5h48my264d?cid=515>
- Make-up lecture this Friday, noon-1:30 pm.

Overview of today's lecture

- Continue convolutional neural networks.
- Quick intro to vision for video.
- Optical flow.
- Constant flow.
- Horn-Schunck flow.

Slide credits

Most of these slides were adapted from:

- Kris Kitani (16-385, Spring 2017).

Course overview

1. Image processing. ← Lectures 1 – 6
See also 18-793: Image and Video Processing
2. Geometry-based vision. ← Lectures 7 – 13
See also 16-822: Geometry-based Methods in Vision
3. Physics-based vision. ← Lectures 14 – 17
See also 16-823: Physics-based Methods in Vision
See also 15-463: Computational Photography
4. Semantic vision. ← Lectures 18 – 22
See also 16-824: Vision Learning and Recognition
5. Dealing with motion. ← We are starting this part now

Computer vision for video

$$\begin{bmatrix} I_x(\mathbf{p}_1) & I_y(\mathbf{p}_1) \\ I_x(\mathbf{p}_2) & I_y(\mathbf{p}_2) \\ \vdots & \vdots \\ I_x(\mathbf{p}_{25}) & I_y(\mathbf{p}_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p}_1) \\ I_t(\mathbf{p}_2) \\ \vdots \\ I_t(\mathbf{p}_{25}) \end{bmatrix}$$

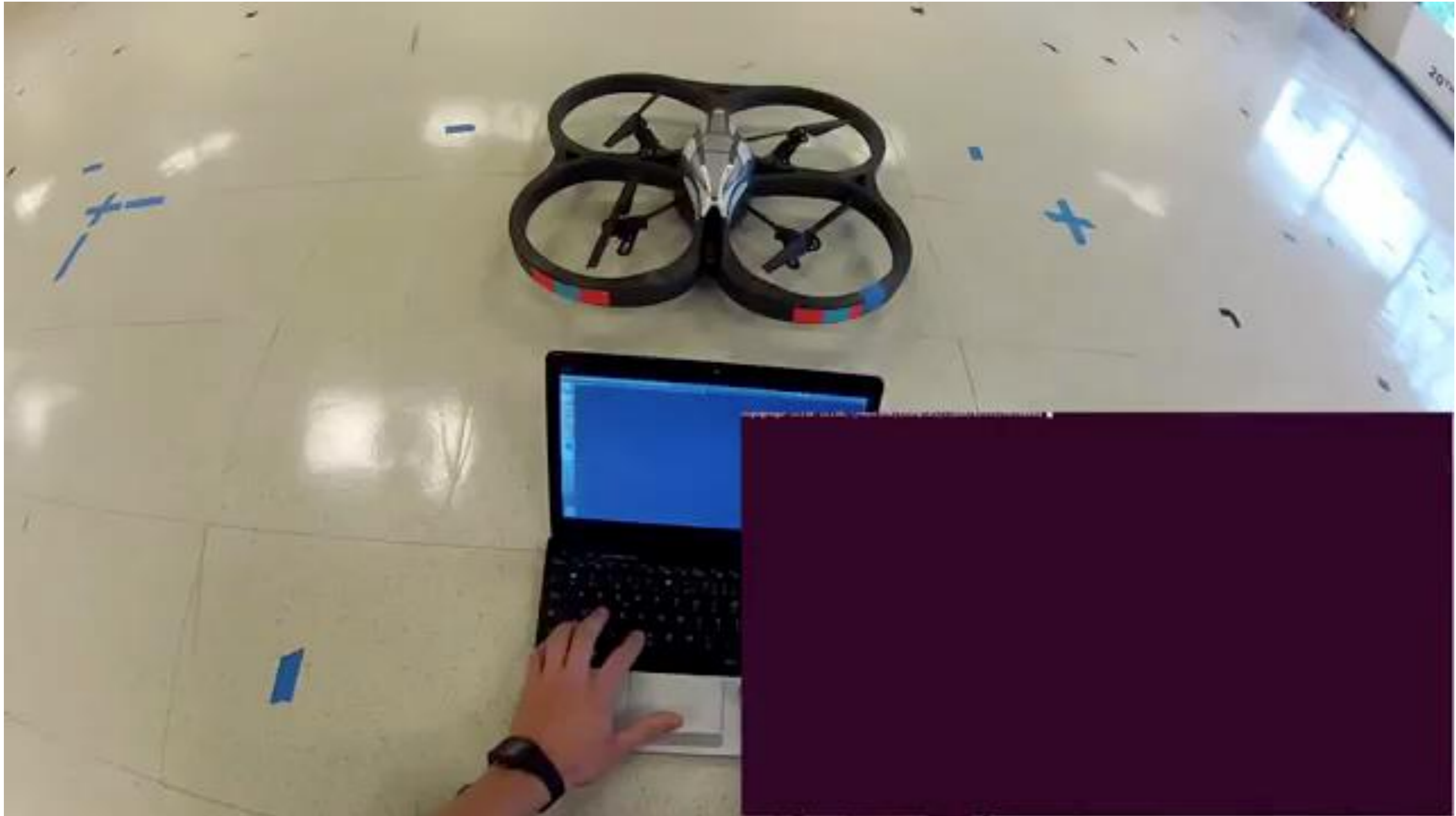
Constant Flow

$$\min_{\mathbf{u}, \mathbf{v}} \sum_{ij} \left\{ E_d(i, j) + \lambda E_s(i, j) \right\}$$

Horn-Schunck

Optical Flow

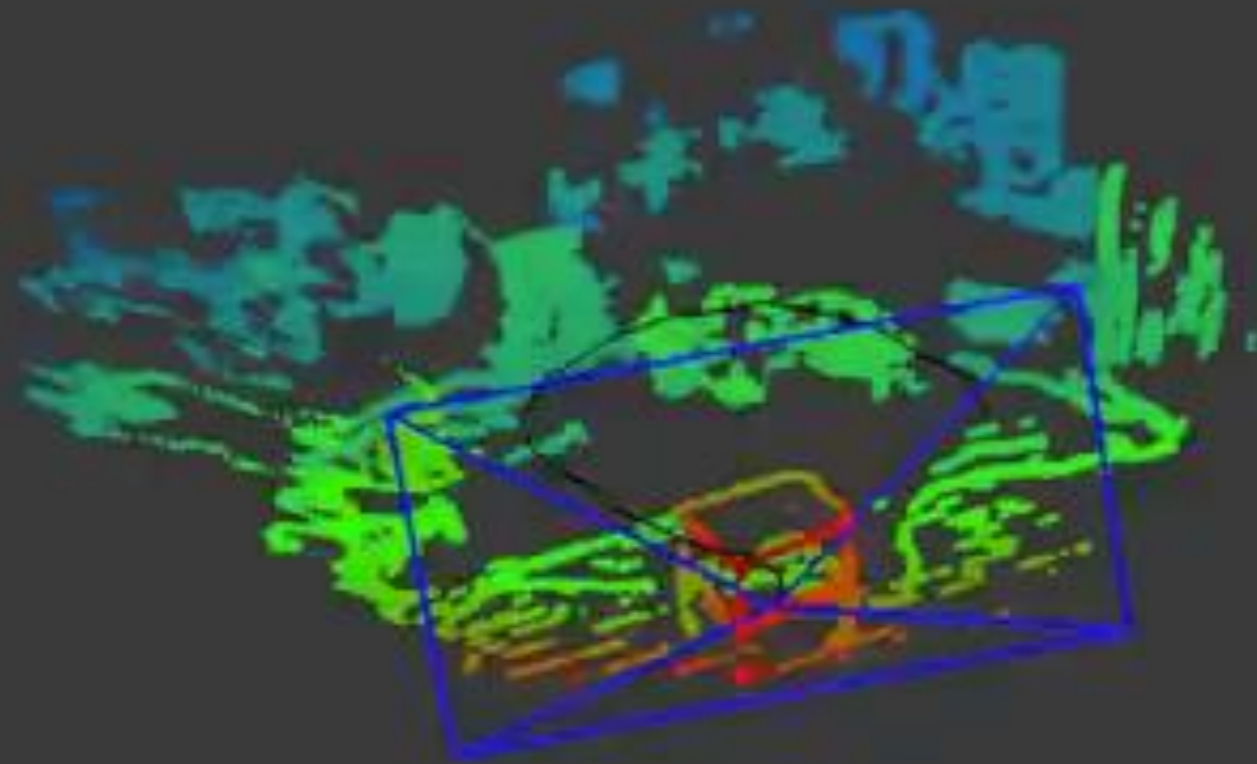
Optical flow used for feature tracking on a drone



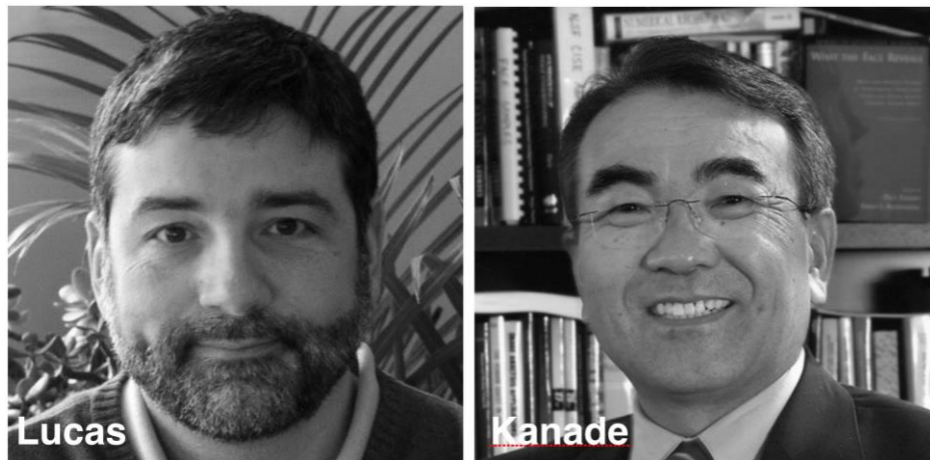
optical flow used for motion estimation in visual odometry



00:00:08.000



It was captured in a motion capture system,
which is reason for the flickering lights.

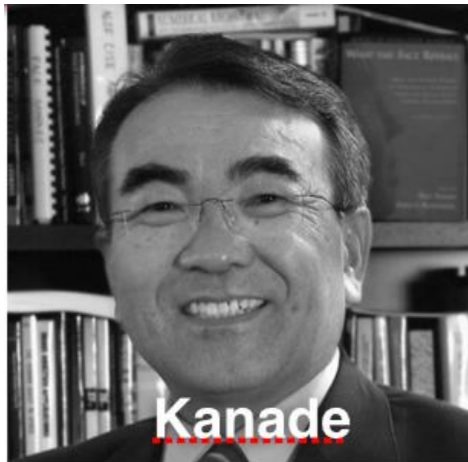


Lucas-Kanade
(Forward additive)

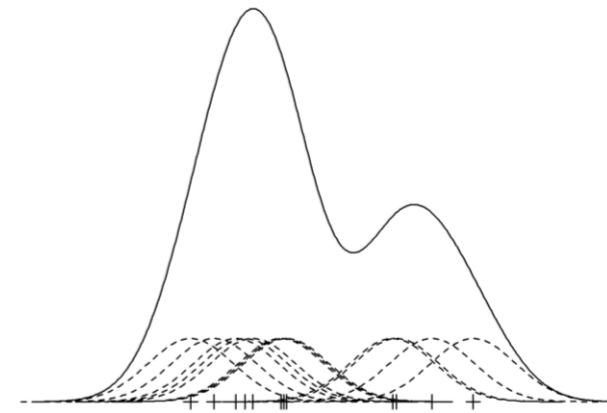


Baker-Matthews
(Inverse Compositional)

Image Alignment



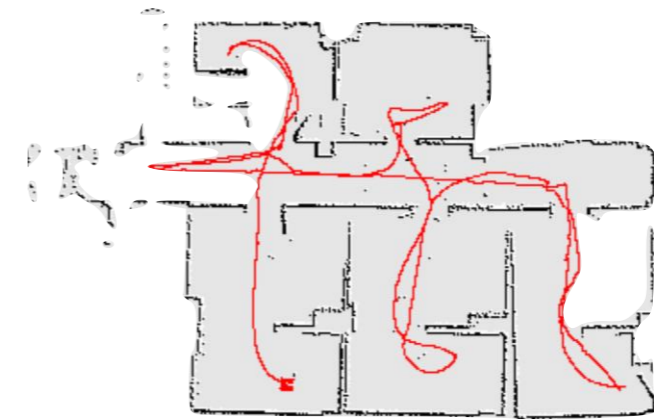
KLT



Mean shift



Kalman Filtering



SLAM

Tracking in Video

Optical flow

Optical Flow

Problem Definition

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

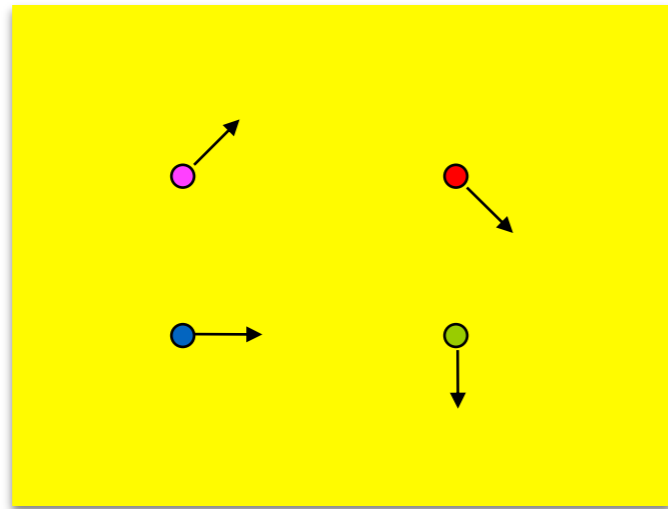
Assumptions

Brightness constancy

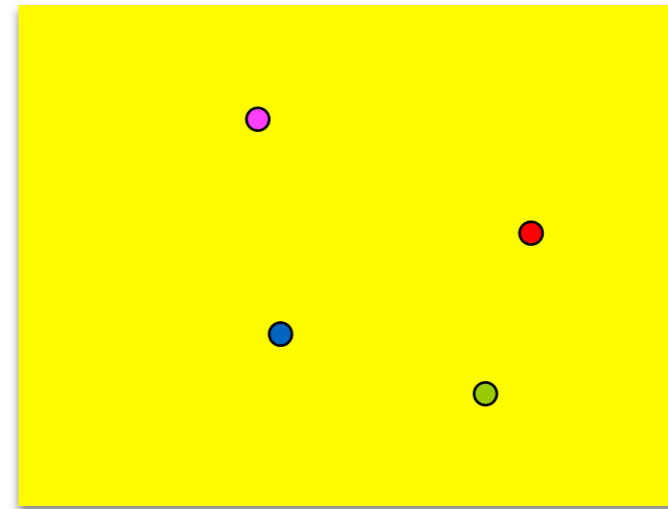
Small motion

Optical Flow

(Problem definition)



$I(x, y, t)$



$I(x, y, t')$

Estimate the motion
(flow) between these two
consecutive images

How is this different from estimating a 2D transform?

Key Assumptions

(unique to optical flow)

Color Constancy

(Brightness constancy for intensity images)

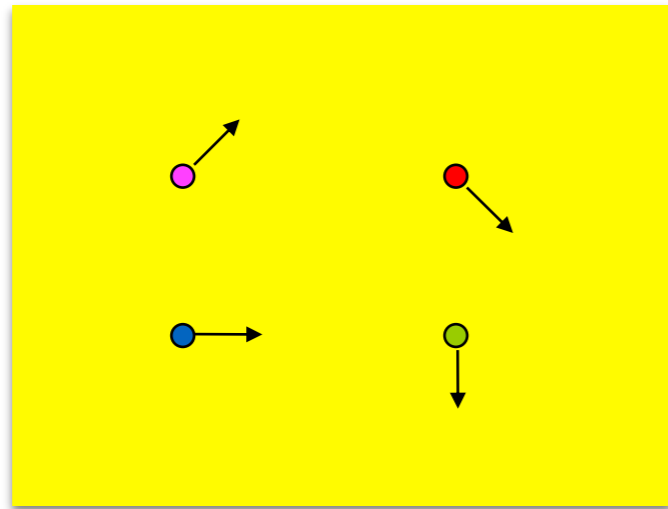
Implication: allows for pixel to pixel comparison
(not image features)

Small Motion

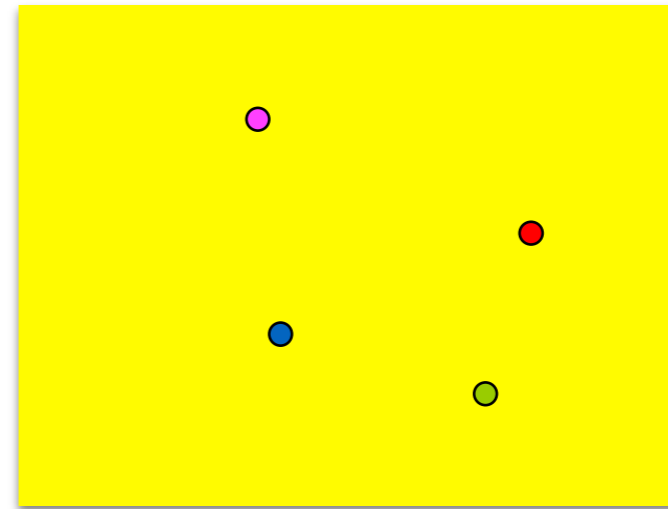
(pixels only move a little bit)

Implication: linearization of the brightness
constancy constraint

Approach



$I(x, y, t)$



$I(x, y, t')$

Look for **nearby pixels** with the **same color**

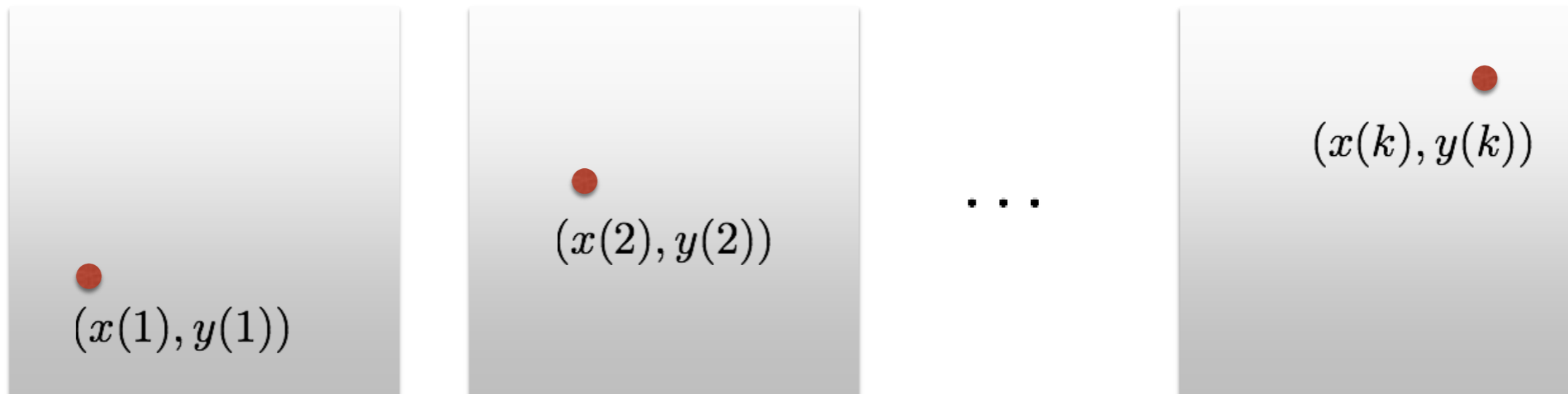
(small motion)

(color constancy)

Assumption 1

Brightness constancy

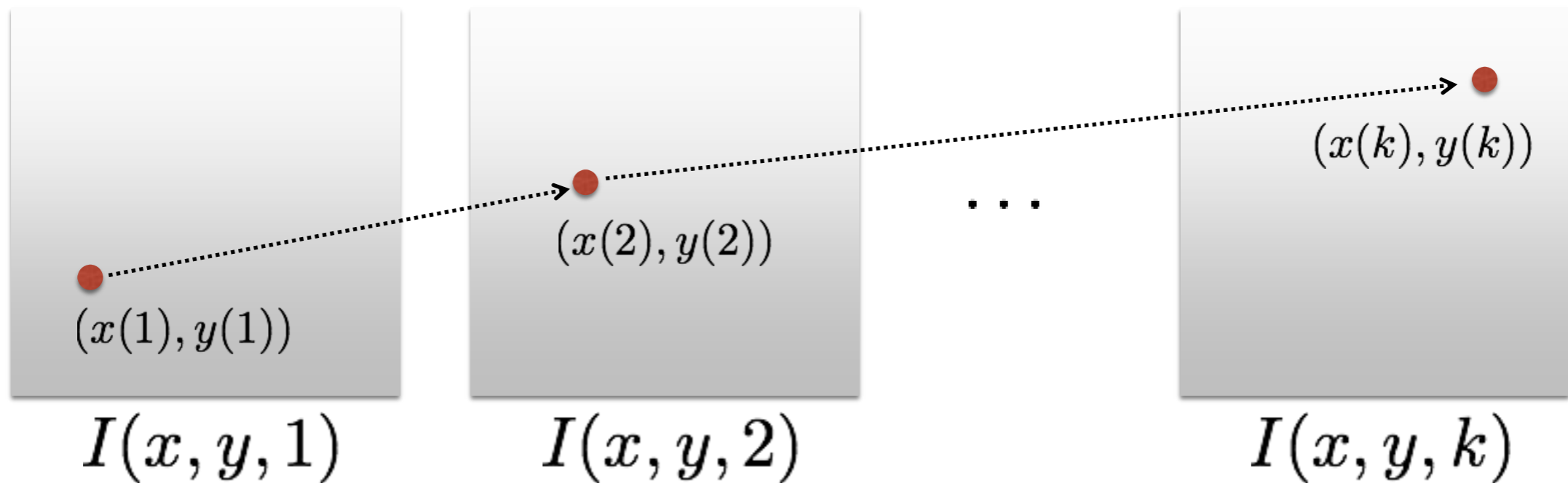
Scene point moving through image sequence



Assumption 1

Brightness constancy

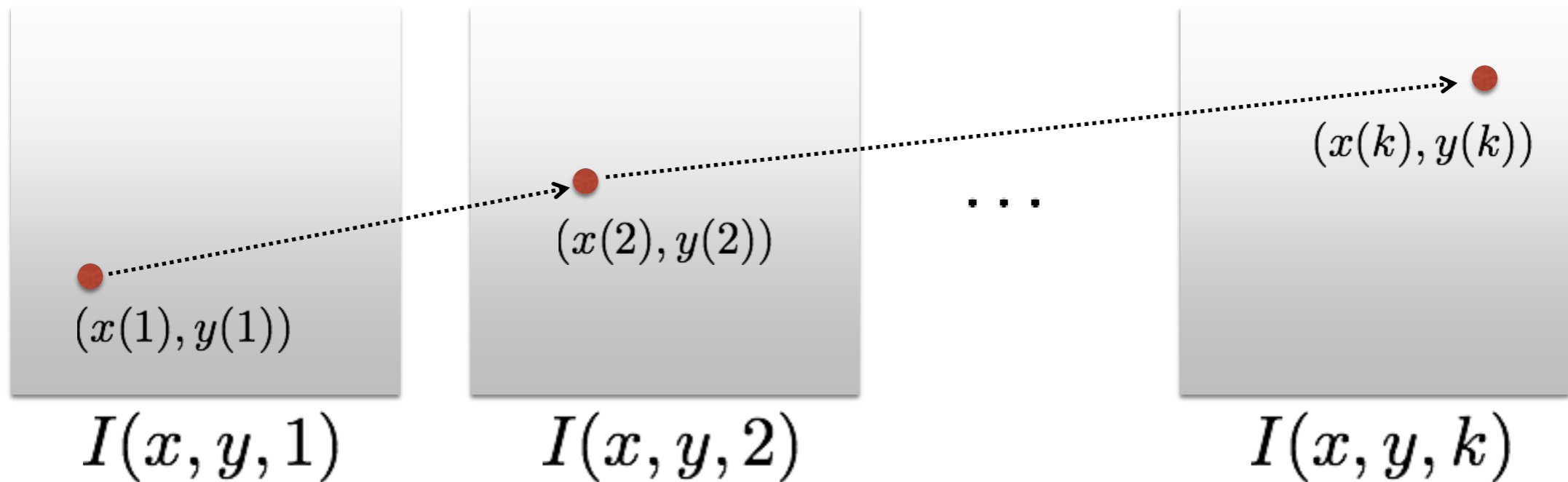
Scene point moving through image sequence



Assumption 1

Brightness constancy

Scene point moving through image sequence

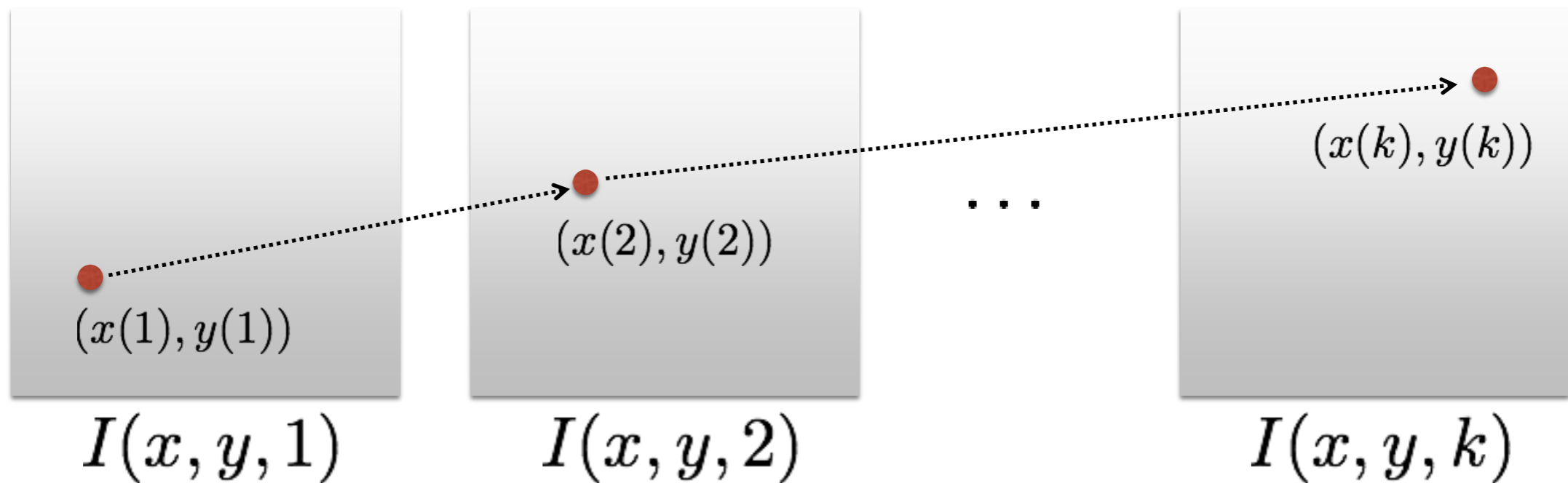


Assumption: Brightness of the point will remain the same

Assumption 1

Brightness constancy

Scene point moving through image sequence



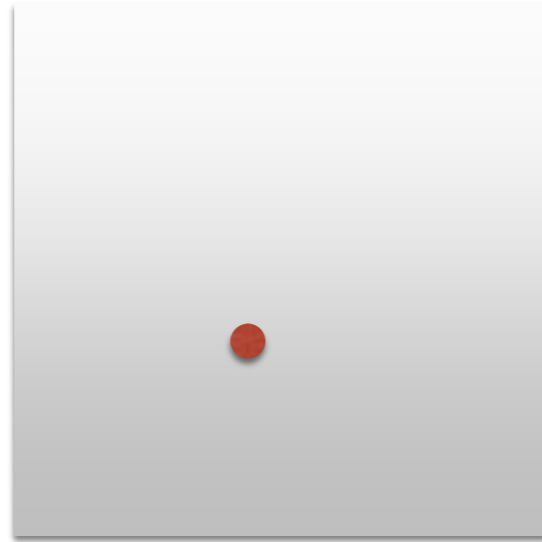
Assumption: Brightness of the point will remain the same

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

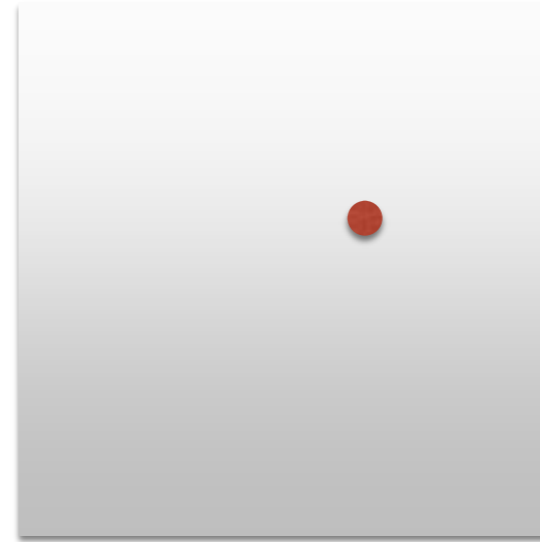
constant

Assumption 2

Small motion



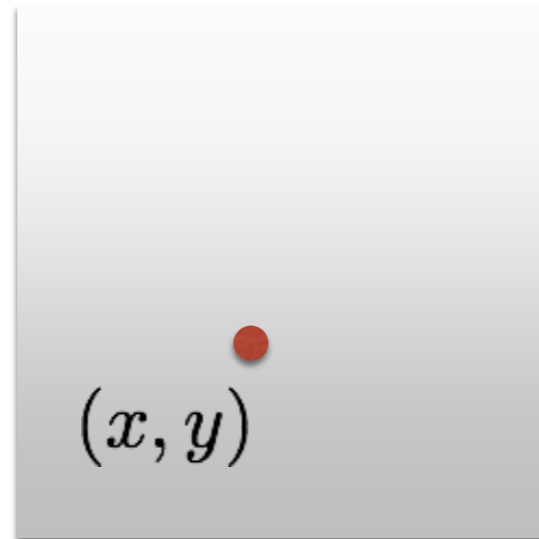
$I(x, y, t)$



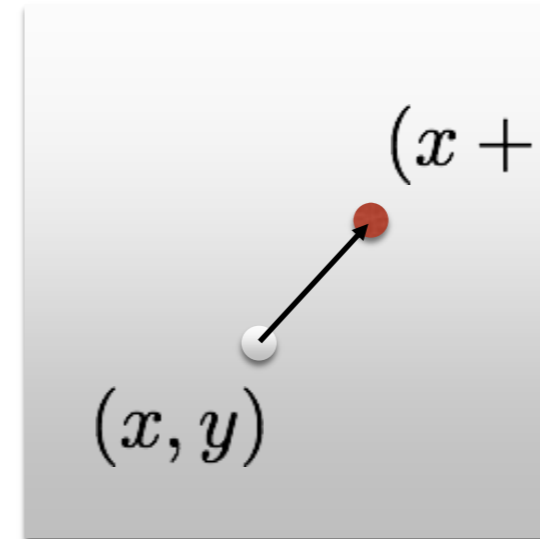
$I(x, y, t + \delta t)$

Assumption 2

Small motion



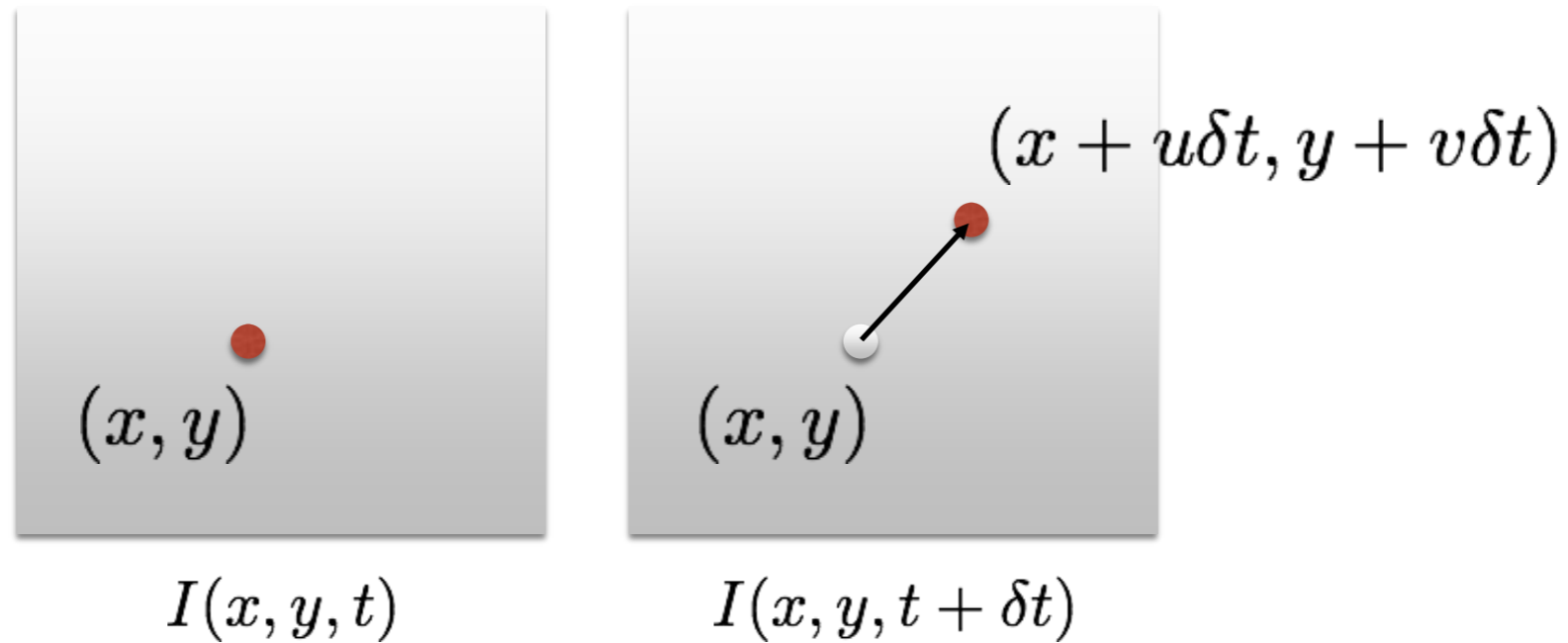
$I(x, y, t)$



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

Assumption 2

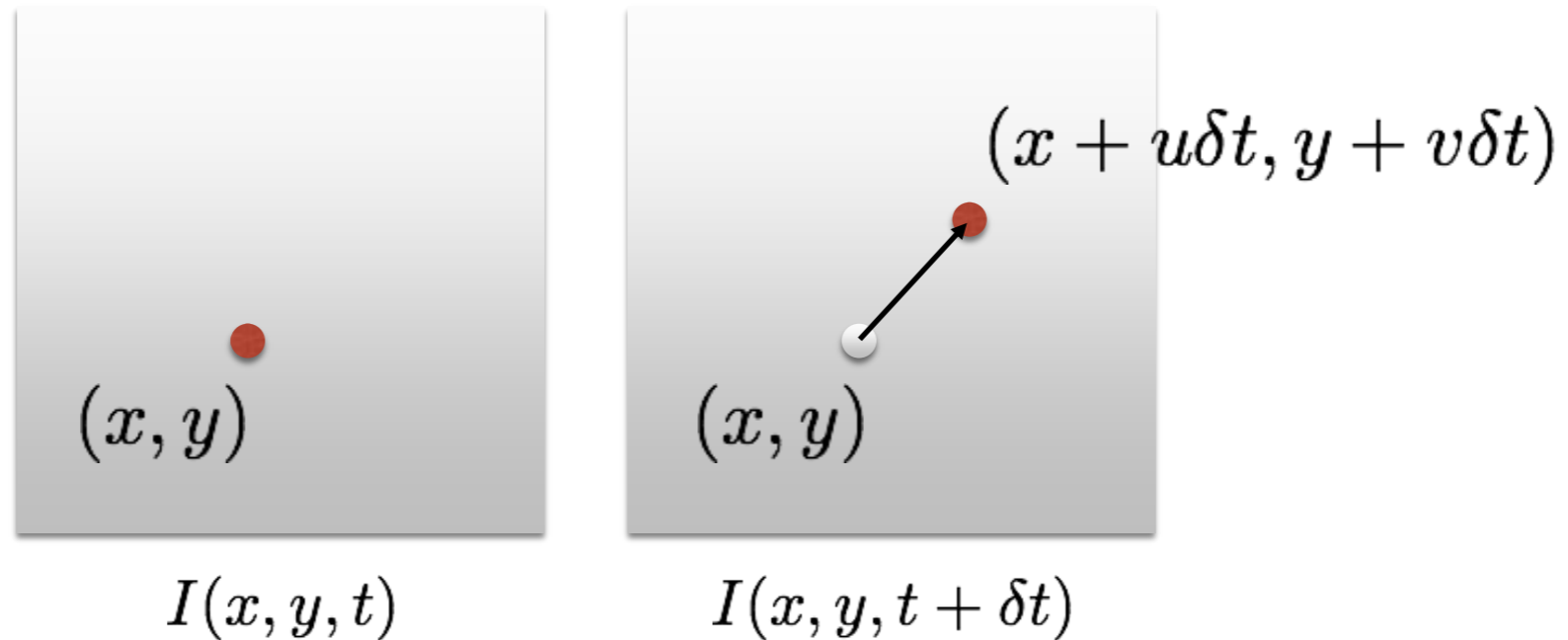
Small motion



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

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

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

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

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t) \quad \text{assuming small motion}$$

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

partial derivative

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

fixed point

assuming small motion

cancel terms

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t) \quad \text{assuming small motion}$$

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \quad \text{cancel terms}$$

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t) \quad \text{assuming small motion}$$

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \quad \begin{array}{l} \text{divide by } \delta t \\ \text{take limit } \delta t \rightarrow 0 \end{array}$$

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

$$I(x, y, t) + \frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t = I(x, y, t) \quad \text{assuming small motion}$$

$$\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t = 0 \quad \begin{array}{l} \text{divide by } \delta t \\ \text{take limit } \delta t \rightarrow 0 \end{array}$$

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

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

Multivariable Taylor Series Expansion

(First order approximation, two variables)

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

$$I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x, y, t) \quad \text{assuming small motion}$$

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \quad \begin{array}{l} \text{divide by } \delta t \\ \text{take limit } \delta t \rightarrow 0 \end{array}$$

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

Brightness Constancy Equation

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

**Brightness
Constancy Equation**

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

(x-flow) (y-flow)

shorthand notation

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

(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?

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

(putting the math aside for a second...)

What do the term of the
brightness constancy equation represent?

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

Image gradients
(at a point p)



(putting the math aside for a second...)

What do the term of the
brightness constancy equation represent?

flow velocities

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

Image gradients
(at a point p)

(putting the math aside for a second...)

What do the term 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

How do you compute these terms?

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

How do you compute ...

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

spatial derivative

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

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

...

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

How do you compute ...

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

spatial derivative

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

temporal derivative

Forward difference

Sobel filter

Derivative-of-Gaussian filter

...

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

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

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

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

-

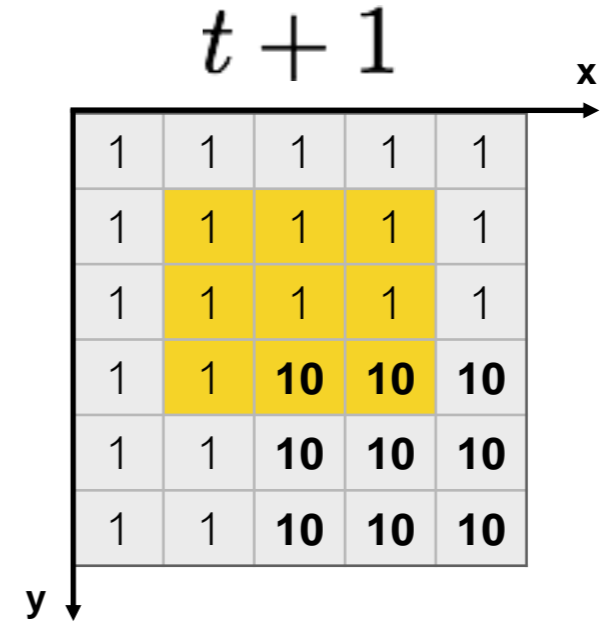
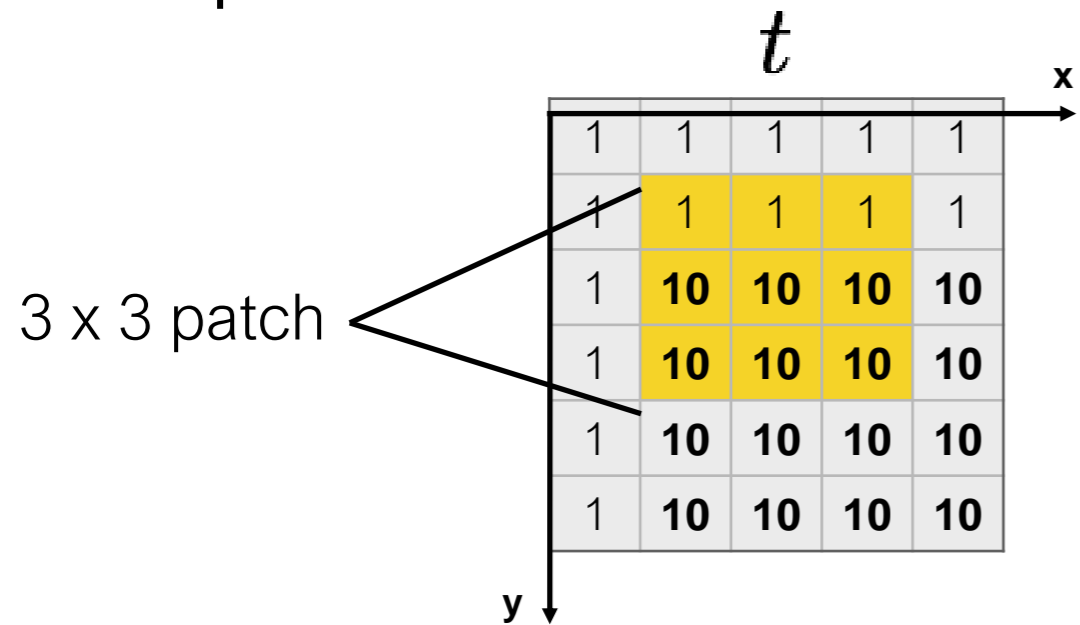
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	10	10	10
1	1	10	10	10
1	1	10	10	10

=

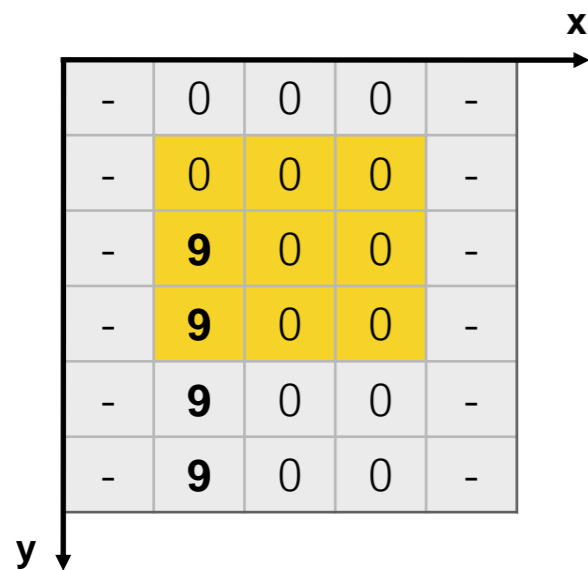
0	0	0	0	0
0	0	0	0	0
0	9	9	9	9
0	9	0	0	0
0	9	0	0	0
0	9	0	0	0

(example of a forward difference)

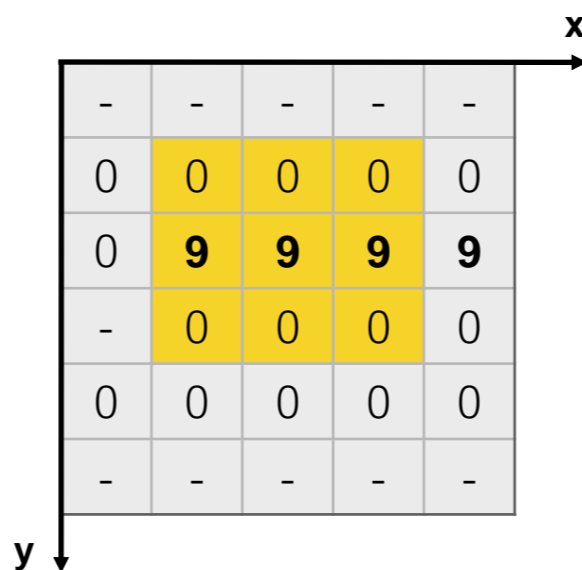
Example:



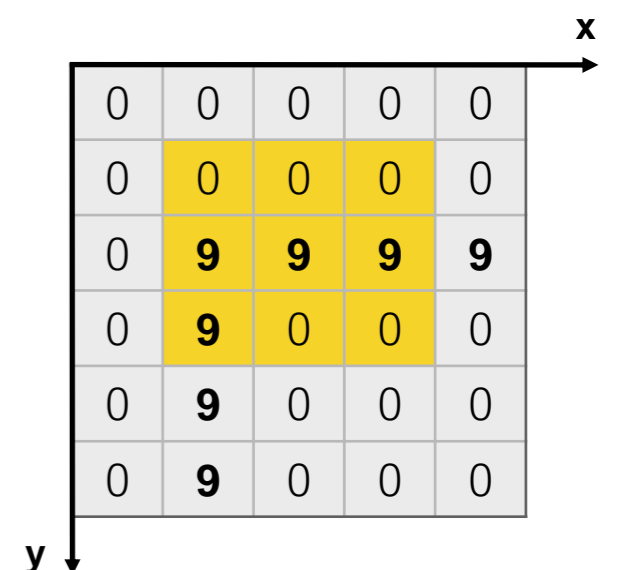
$$I_x = \frac{\partial I}{\partial x}$$



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



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



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

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
...

$$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$$

optical flow

How do you compute this?

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

temporal derivative

frame differencing

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

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
...

$$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$$

optical flow

We need to solve for this!
(this is the unknown in the optical flow problem)

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

temporal derivative

frame differencing

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

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

...

$$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$$

optical flow

(u, v)

Solution lies on a line

Cannot be found uniquely
with a single constraint

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

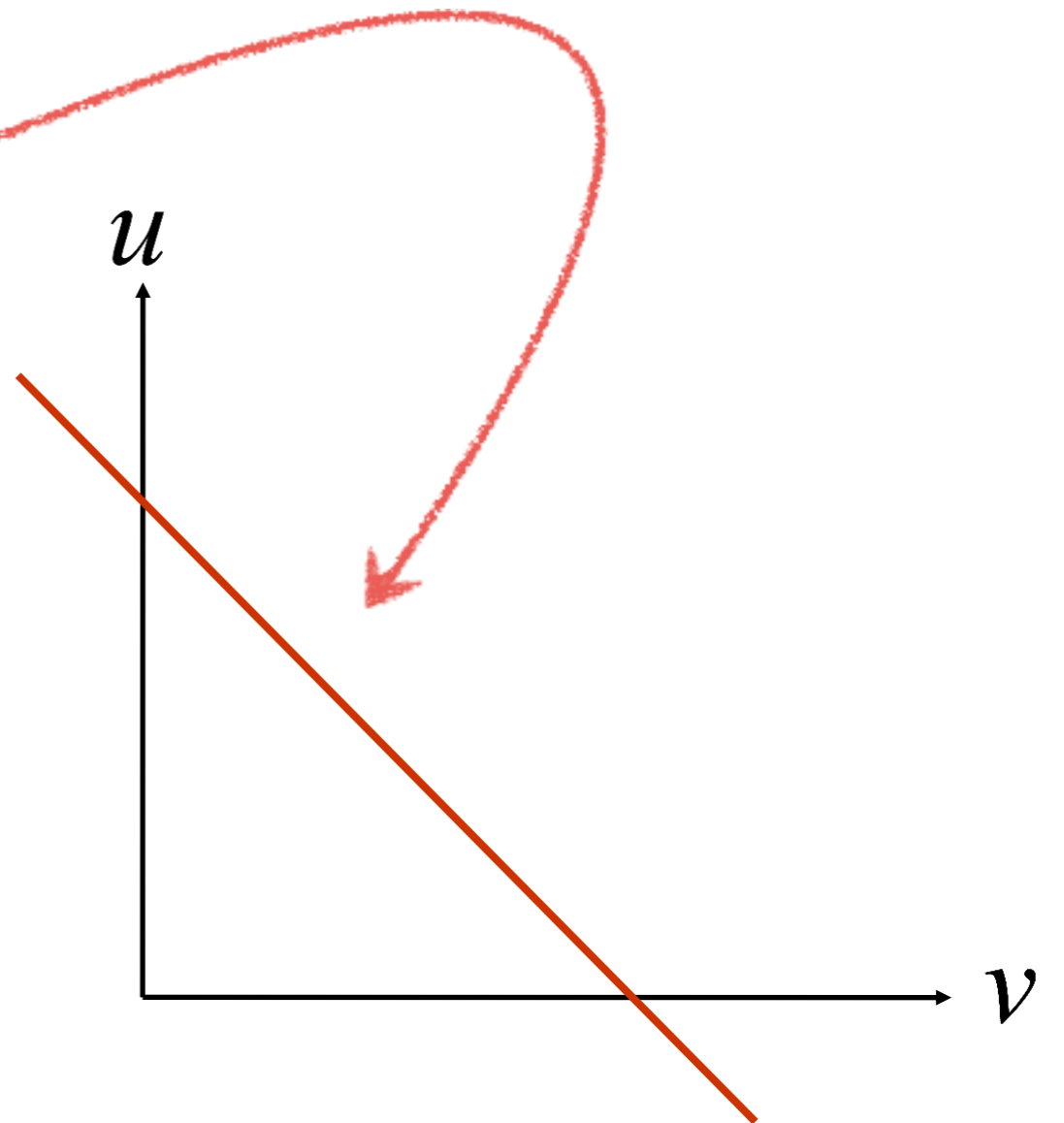
temporal derivative

frame differencing

Solution lies on a straight line

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

many combinations of u and v will satisfy the equality



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

unknown

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

known

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

unknown

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

known

Where do we get more equations (constraints)?

Horn-Schunck Optical Flow (1981)

brightness constancy

small motion

'smooth' flow

(flow can vary from pixel to pixel)

global method
(dense)

Lucas-Kanade Optical Flow (1981)

method of differences

'constant' flow

(flow is constant for all pixels)

local method
(sparse)

Constant flow

Where do we get more equations (constraints)?

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

Assume that the surrounding patch (say 5x5) has
'constant flow'

Assumptions:

Flow is locally smooth

Neighboring pixels have same displacement

Using a 5×5 image patch, gives us  equations

Assumptions:

Flow is locally smooth

Neighboring pixels have same displacement

Using a 5 x 5 image patch, gives us 25 equations

$$I_x(\mathbf{p}_1)u + I_y(\mathbf{p}_1)v = -I_t(\mathbf{p}_1)$$

$$I_x(\mathbf{p}_2)u + I_y(\mathbf{p}_2)v = -I_t(\mathbf{p}_2)$$

⋮

$$I_x(\mathbf{p}_{25})u + I_y(\mathbf{p}_{25})v = -I_t(\mathbf{p}_{25})$$

Equivalent (see TQ10) to solving:

$$A^T A \hat{x} = A^T b$$
$$\begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum_{p \in P} I_x I_t \\ \sum_{p \in P} I_y I_t \end{bmatrix}$$

where the summation is over each pixel p in patch P

$$x = (A^T A)^{-1} A^T b$$

Equivalent (see TQ10) to solving:

$$A^T A \hat{x} = A^T b$$
$$\begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum_{p \in P} I_x I_t \\ \sum_{p \in P} I_y I_t \end{bmatrix}$$

where the summation is over each pixel \mathbf{p} in patch \mathbf{P}

Sometimes called 'Lucas-Kanade Optical Flow'

(can be interpreted to be a special case of the LK method with a translational warp model)

When is this solvable?

$$A^{\top} A \hat{x} = A^{\top} b$$

When is this solvable?

$$A^T A \hat{x} = A^T b$$

$A^T A$ should be invertible

$A^T A$ should not be too small

λ_1 and λ_2 should not be too small

$A^T A$ should be well conditioned

λ_1/λ_2 should not be too large (λ_1 =larger eigenvalue)

Where have you seen this before?

$$A^T A = \begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix}$$

Where have you seen this before?

$$A^T A = \begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix}$$

Harris Corner Detector!

Where have you seen this before?

$$A^T A = \begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix}$$

Harris Corner Detector!

What are the implications?

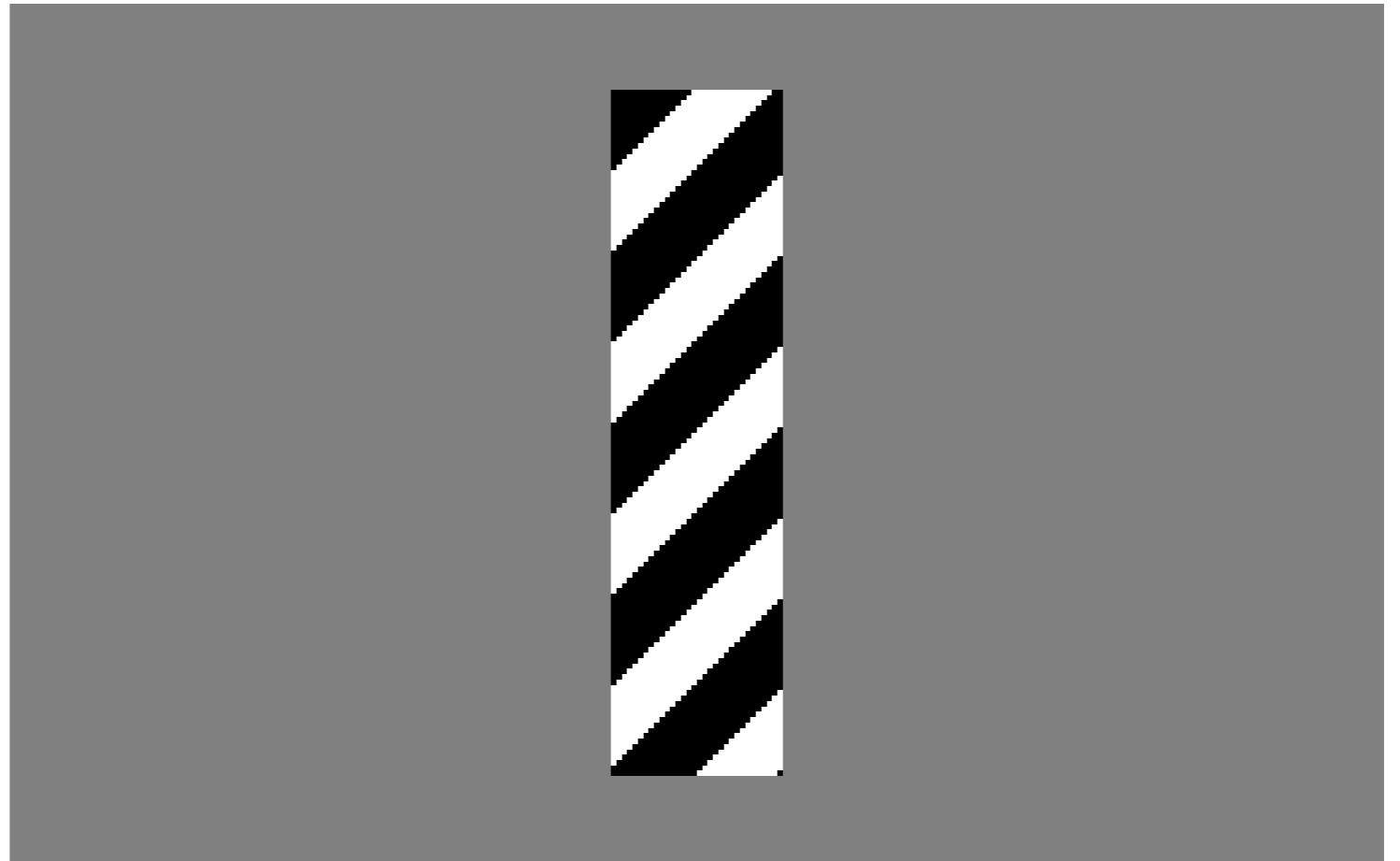
Implications

- Corners are when λ_1, λ_2 are big; this is also when Lucas-Kanade optical flow works best
- Corners are regions with two different directions of gradient (at least)
- Corners are good places to compute flow!

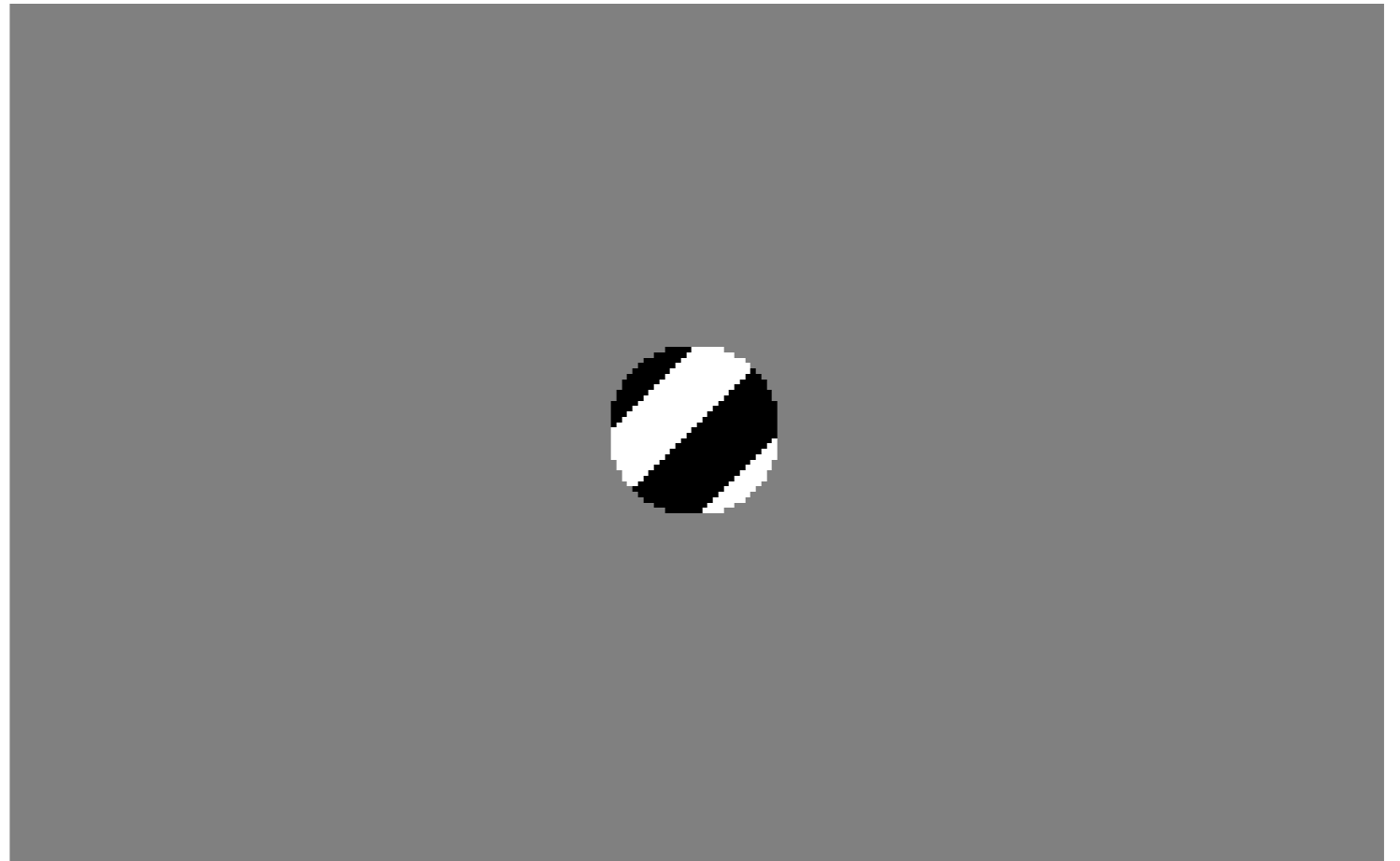
What happens when you have no 'corners'?

*You want to compute optical flow.
What happens if the image patch contains only a line?*

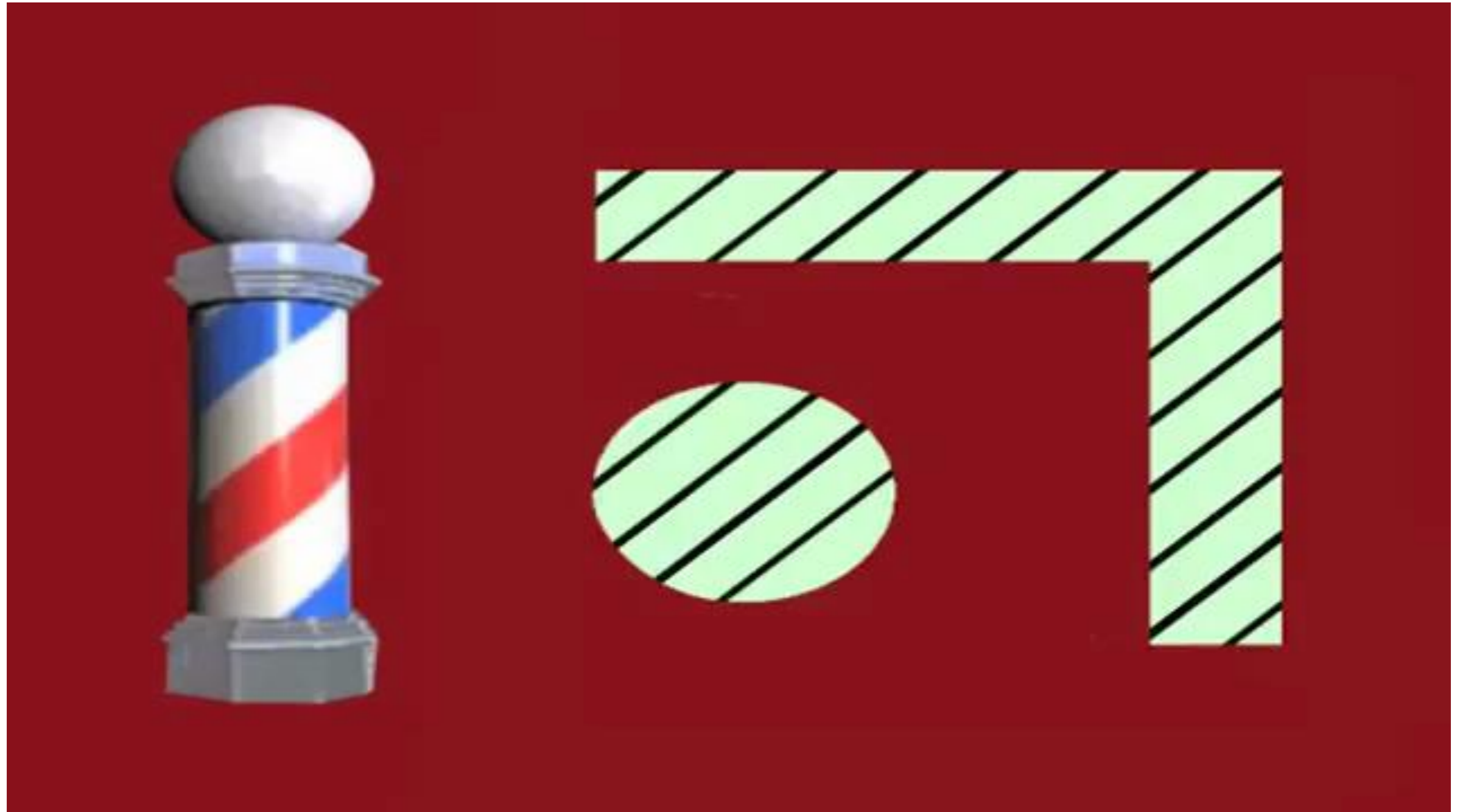
Barber's pole illusion



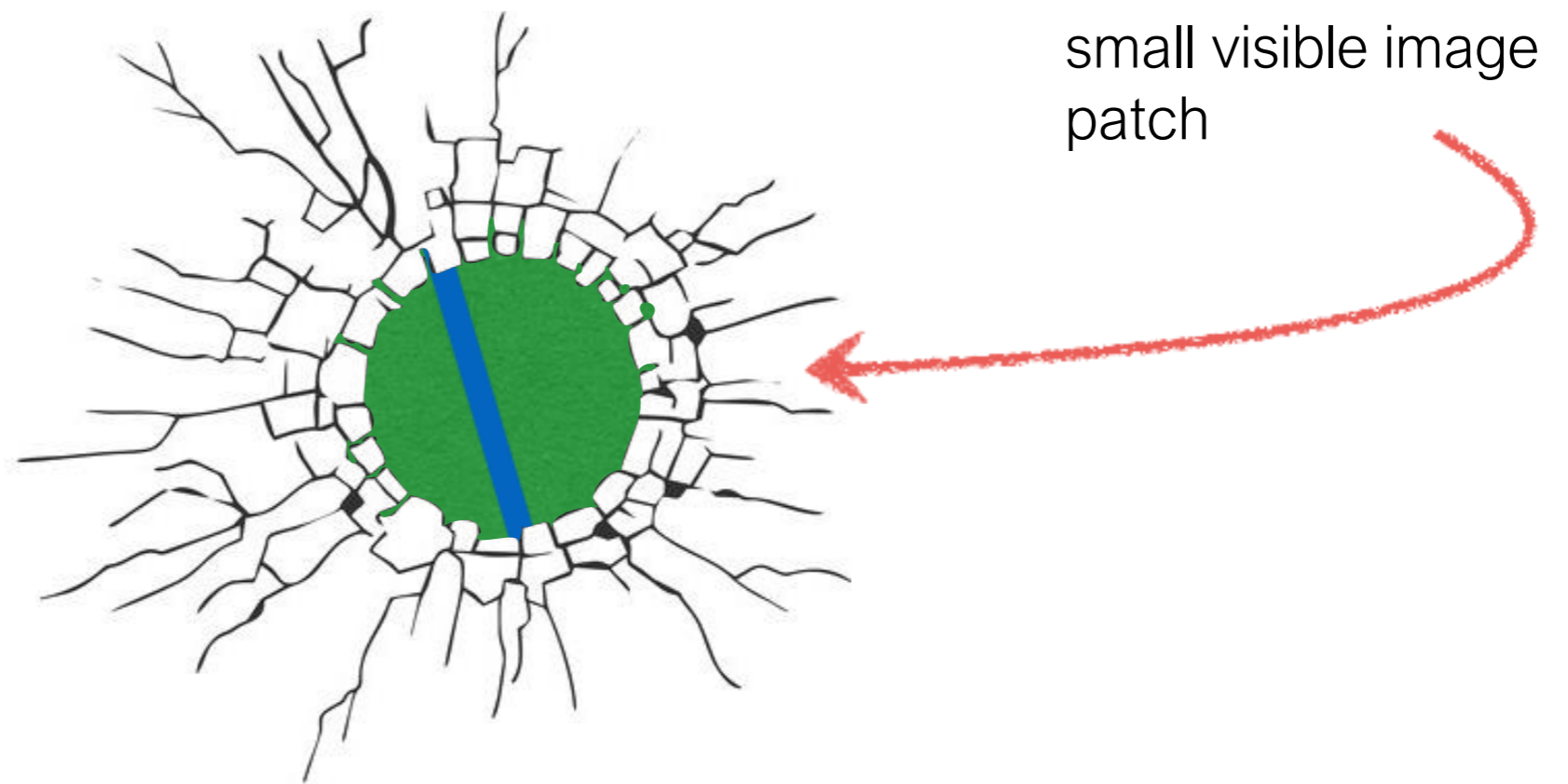
Barber's pole illusion



Barber's pole illusion

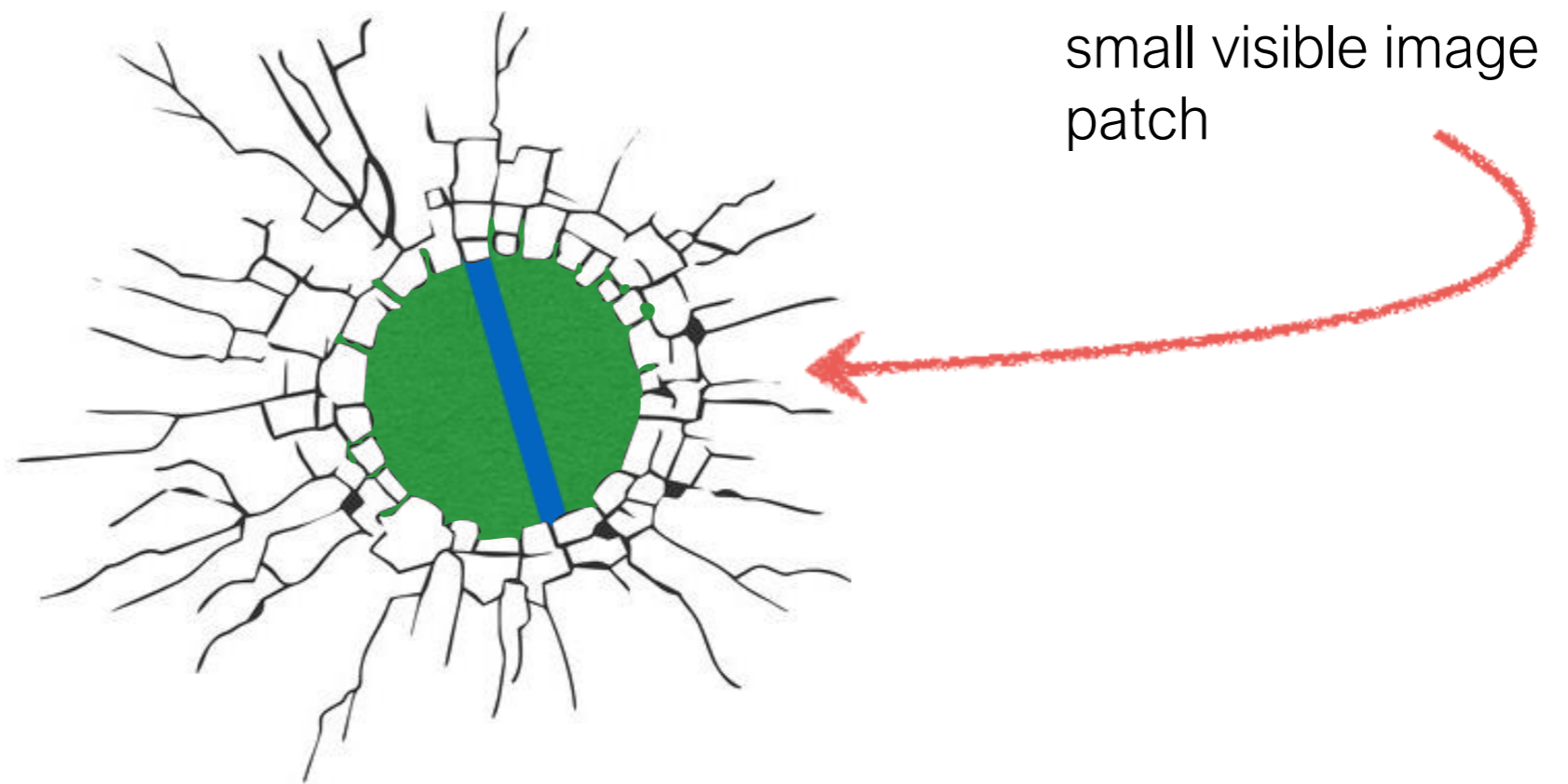


Aperture Problem



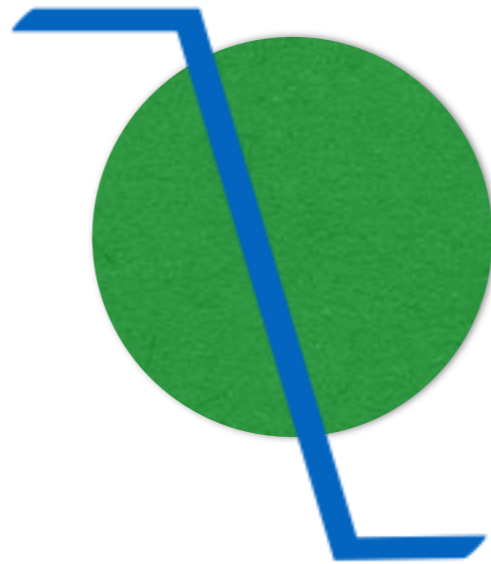
In which direction is the line moving?

Aperture Problem

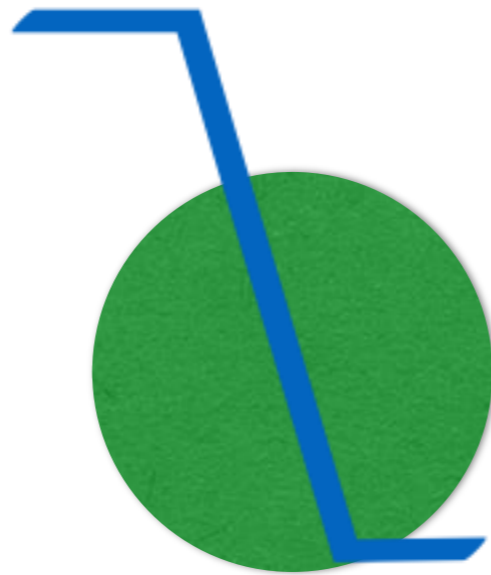


In which direction is the line moving?

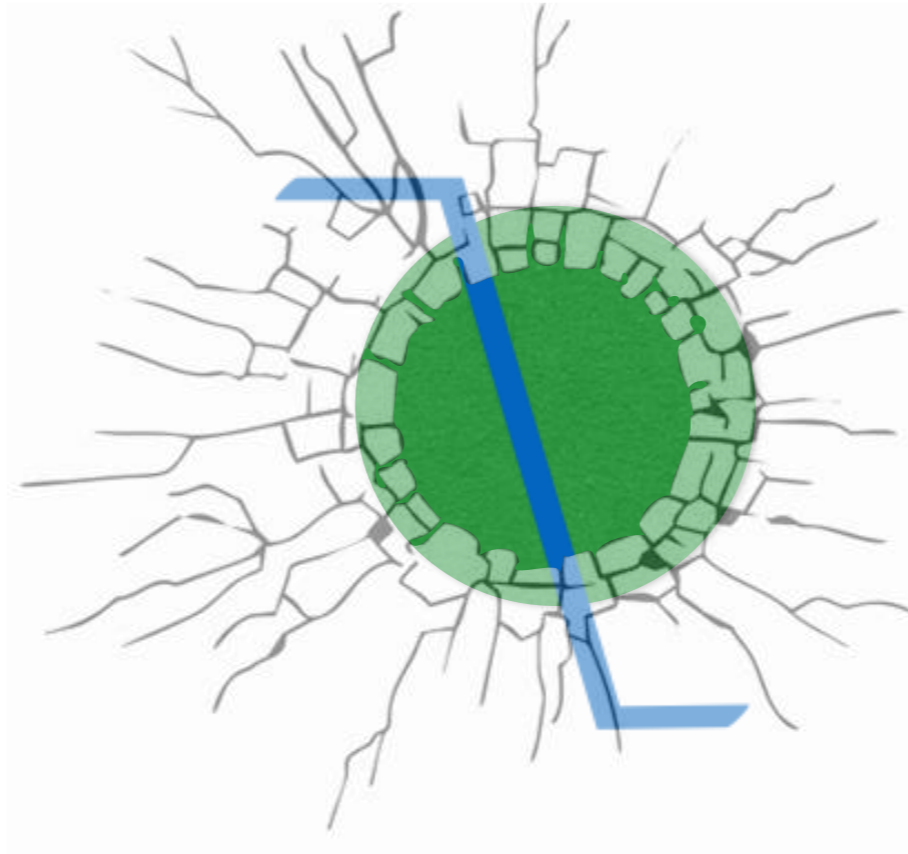
Aperture Problem



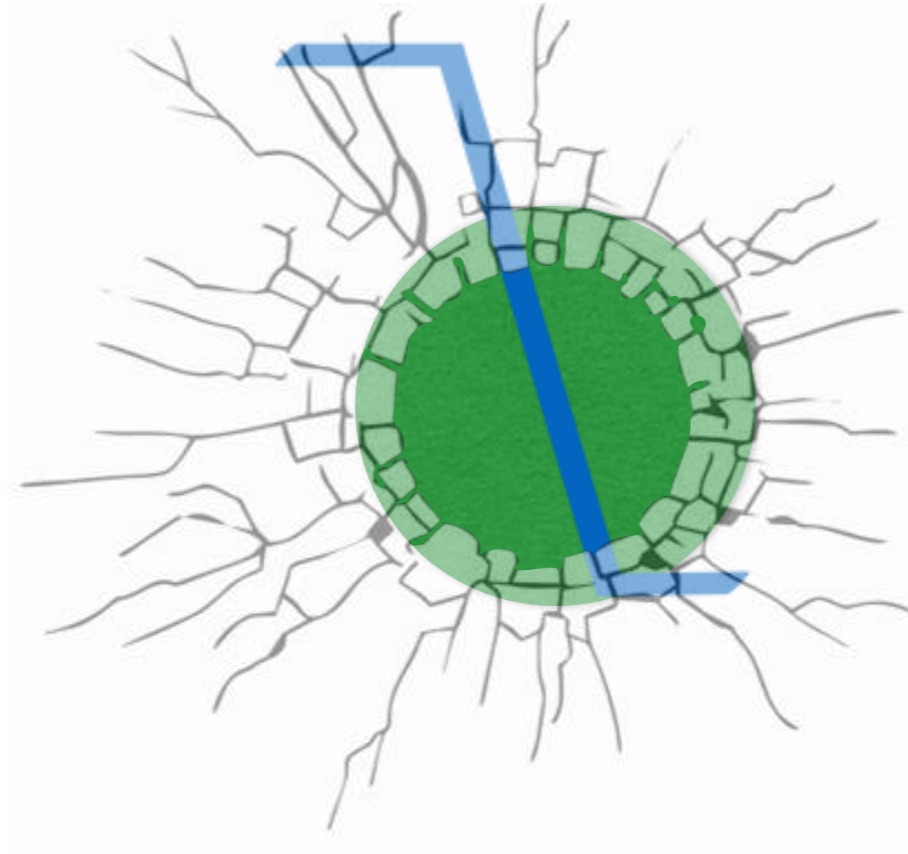
Aperture Problem

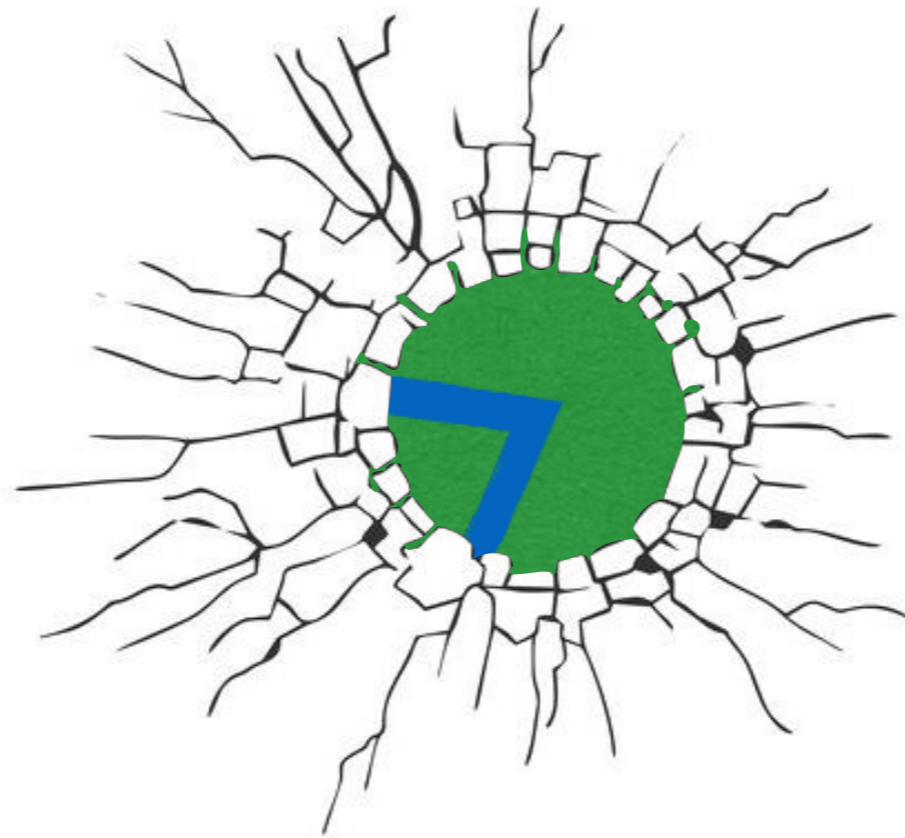


Aperture Problem

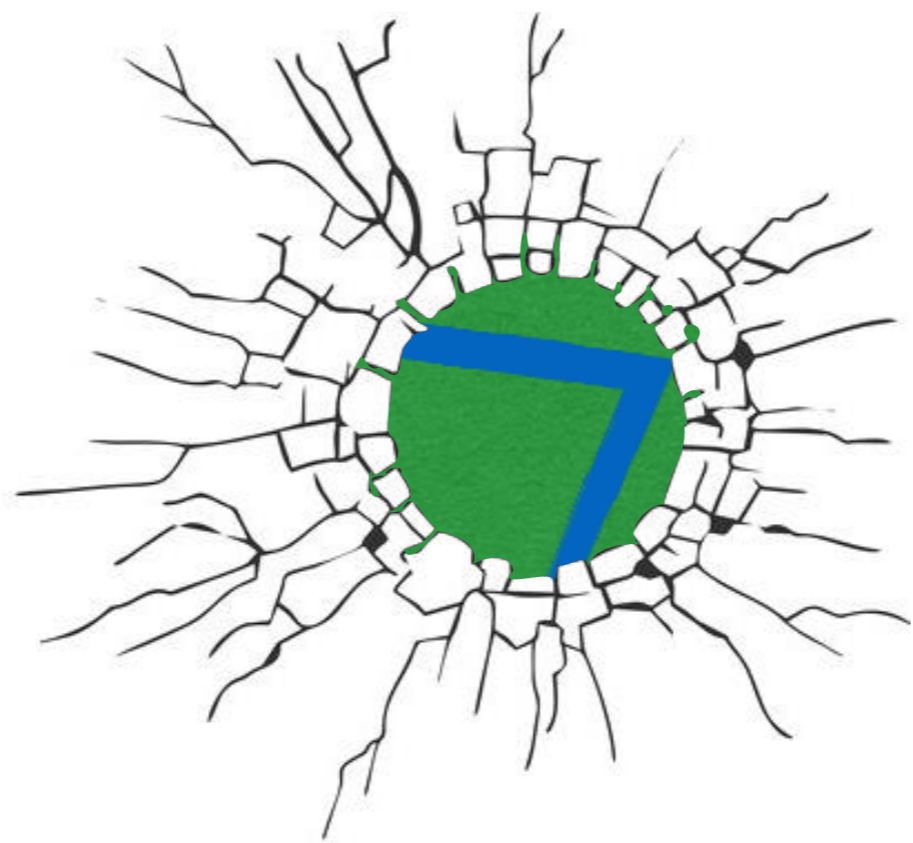


Aperture Problem

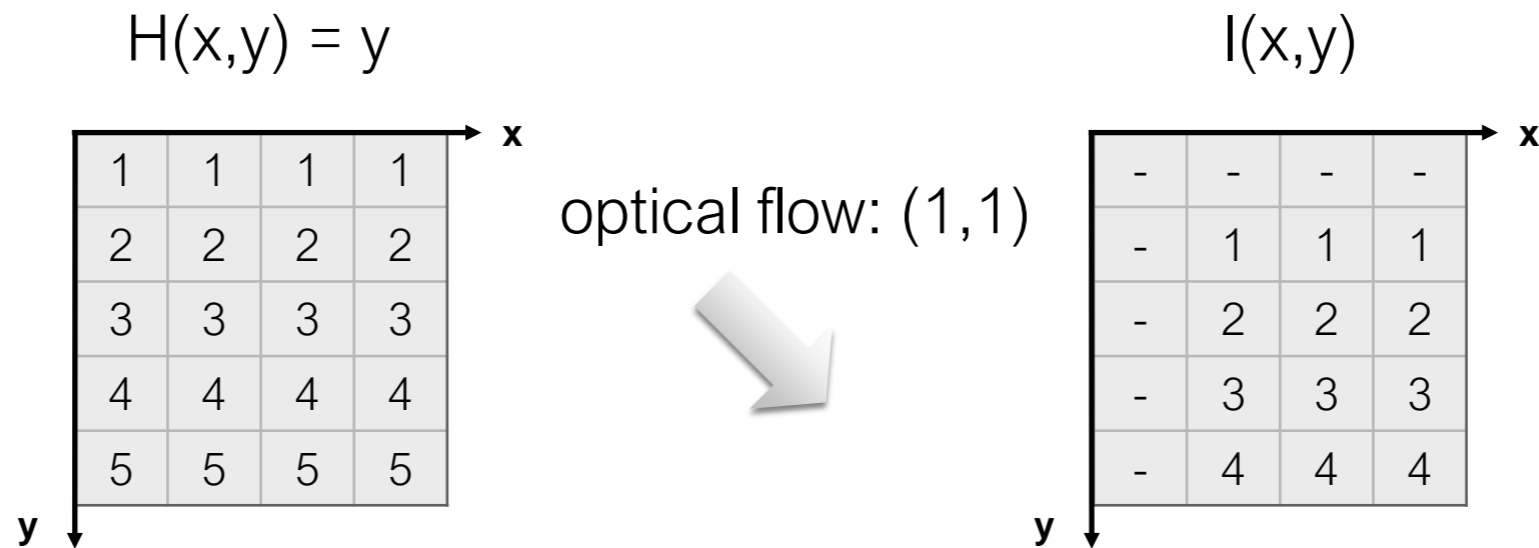




Want patches with different gradients to
the avoid aperture problem



Want patches with different gradients to
the avoid aperture problem



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

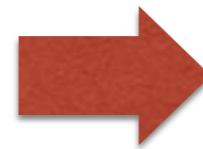
Compute gradients

$$I_x(3,3) = 0$$

$$I_y(3,3) = 1$$

$$I_t(3,3) = I(3,3) - H(3,3) = -1$$

Solution:



$$v = 1$$

We recover the v of the optical flow but not the u .

This is the aperture problem.

Horn-Schunck optical flow

Horn-Schunck Optical Flow (1981)

brightness constancy

small motion

'smooth' flow

(flow can vary from pixel to pixel)

global method
(dense)

Lucas-Kanade Optical Flow (1981)


method of differences

'constant' flow

(flow is constant for all pixels)

local method
(sparse)

Smoothness



**most objects in the world are rigid or
deform elastically
moving together coherently**

we expect optical flow fields to be smooth

Key idea

(of Horn-Schunck optical flow)

Enforce

brightness constancy

Enforce

smooth flow field

to compute optical flow

Key idea

(of Horn-Schunck optical flow)

Enforce

brightness constancy

Enforce

smooth flow field

to compute optical flow

Enforce brightness constancy

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

For every pixel,

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

Enforce brightness constancy

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

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

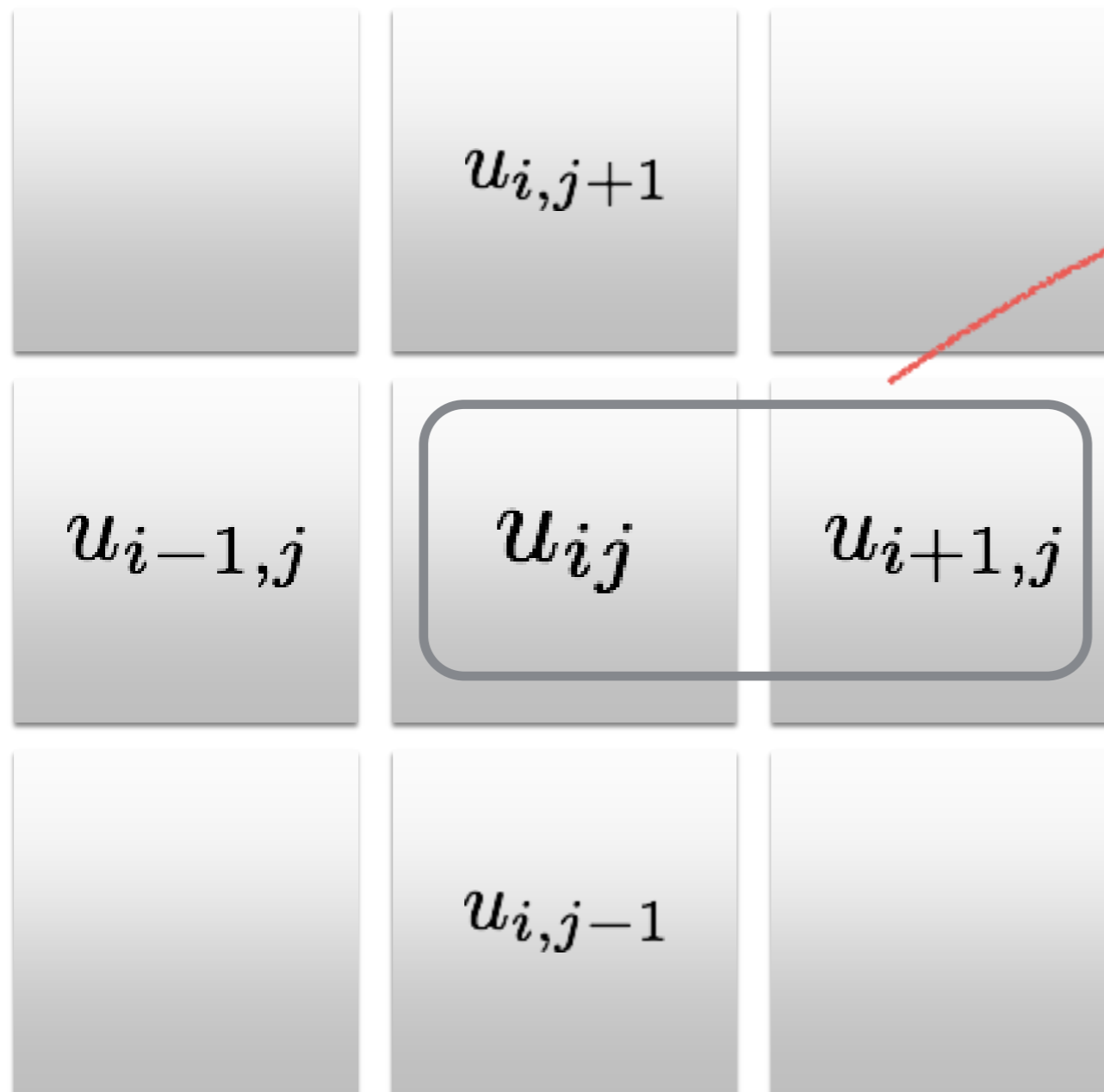
brightness constancy

Enforce

smooth flow field

to compute optical flow

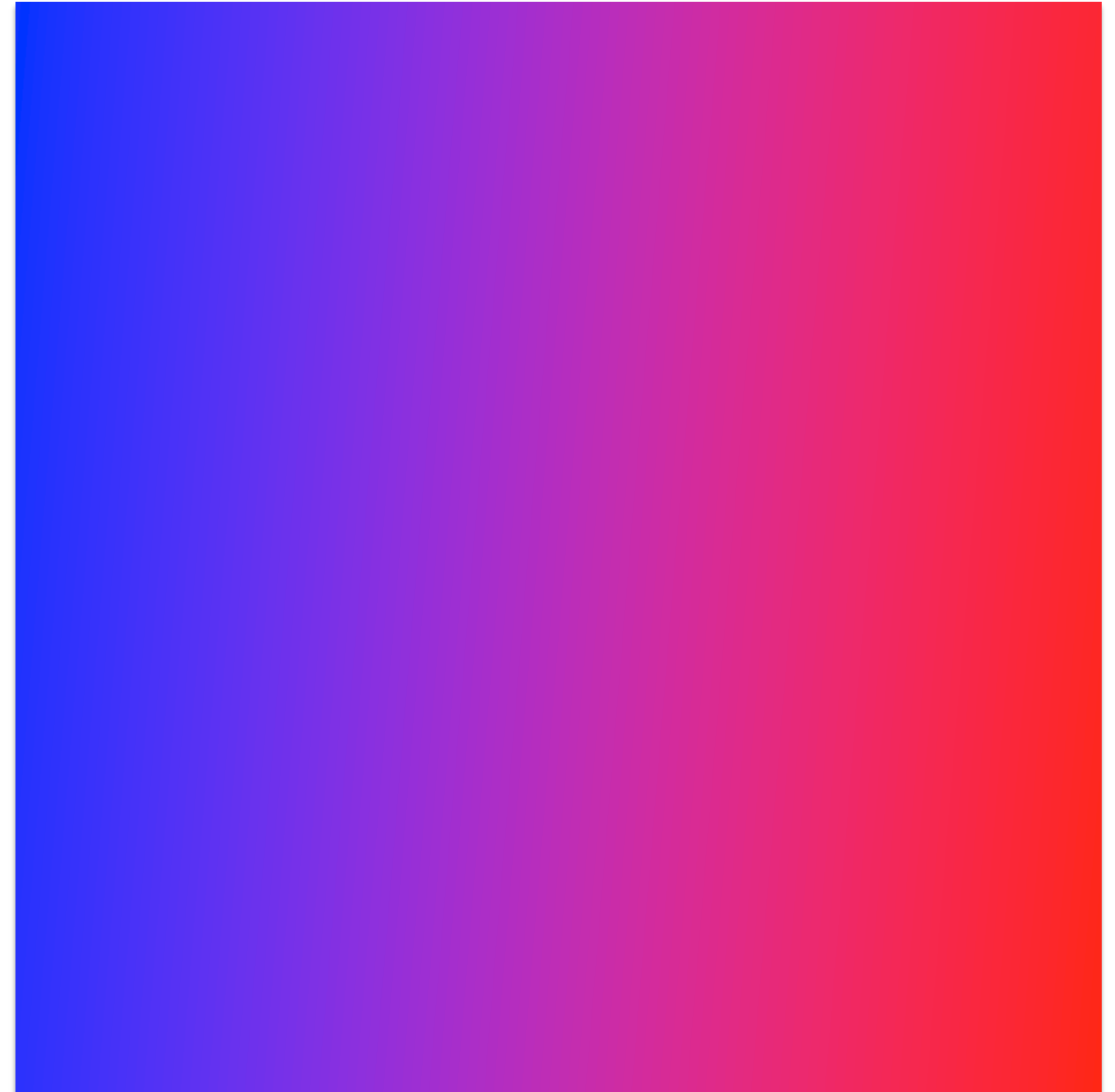
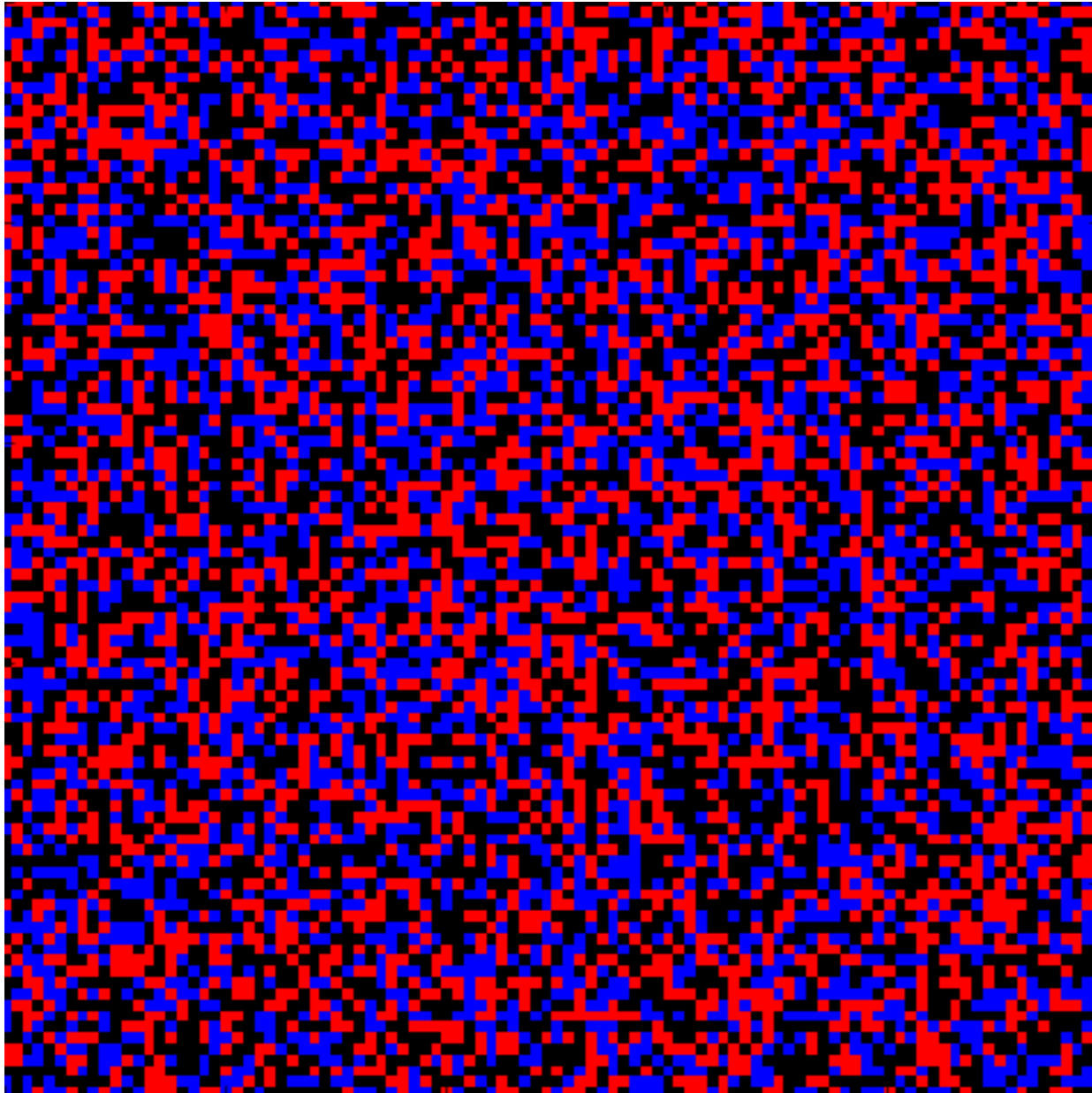
Enforce **smooth flow field**



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

u-component of flow

Which flow field optimizes the objective? $\min_{\mathbf{u}} (u_{i,j} - u_{i+1,j})^2$

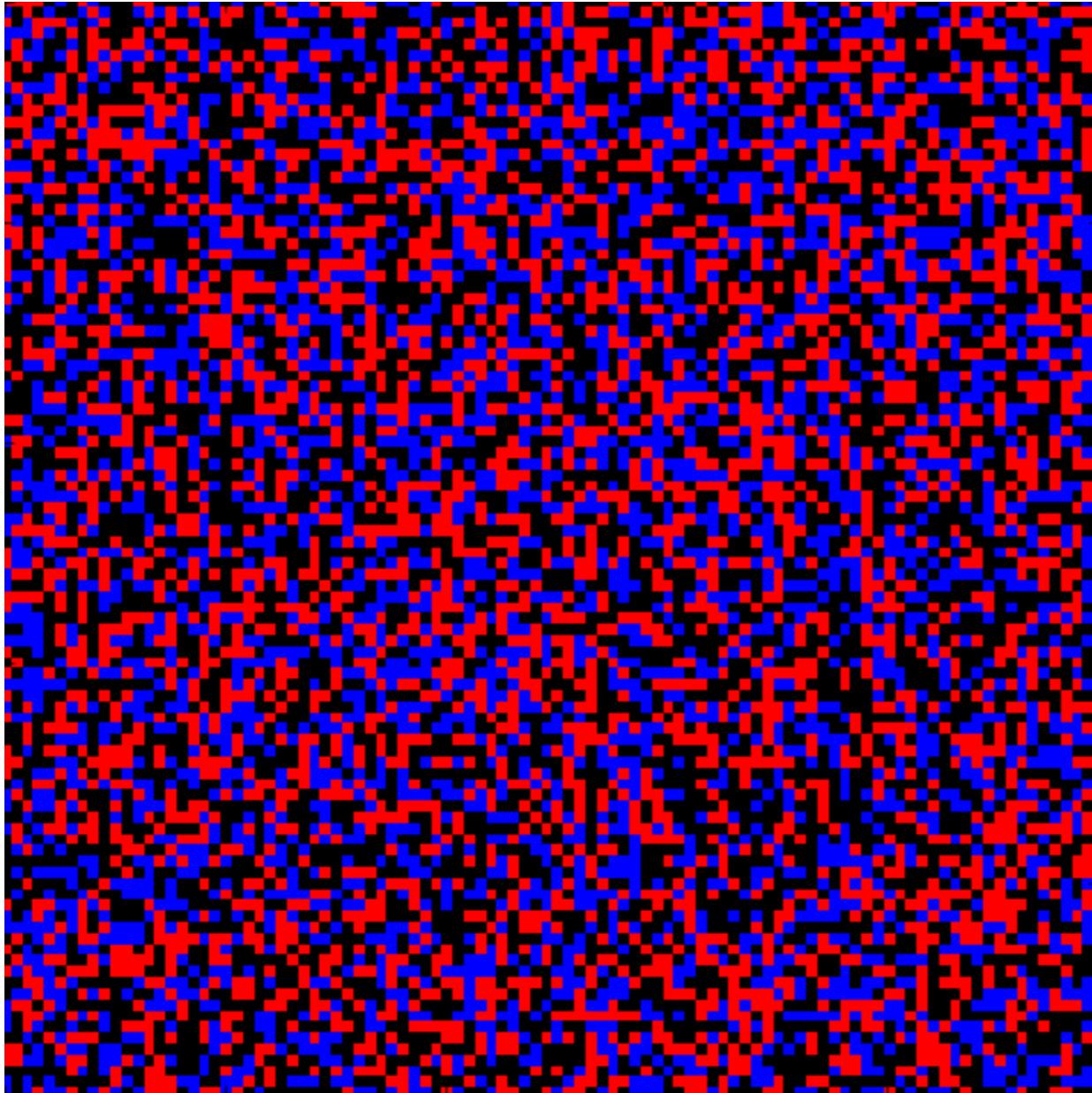


$$\sum_{ij} (u_{ij} - u_{i+1,j})^2$$

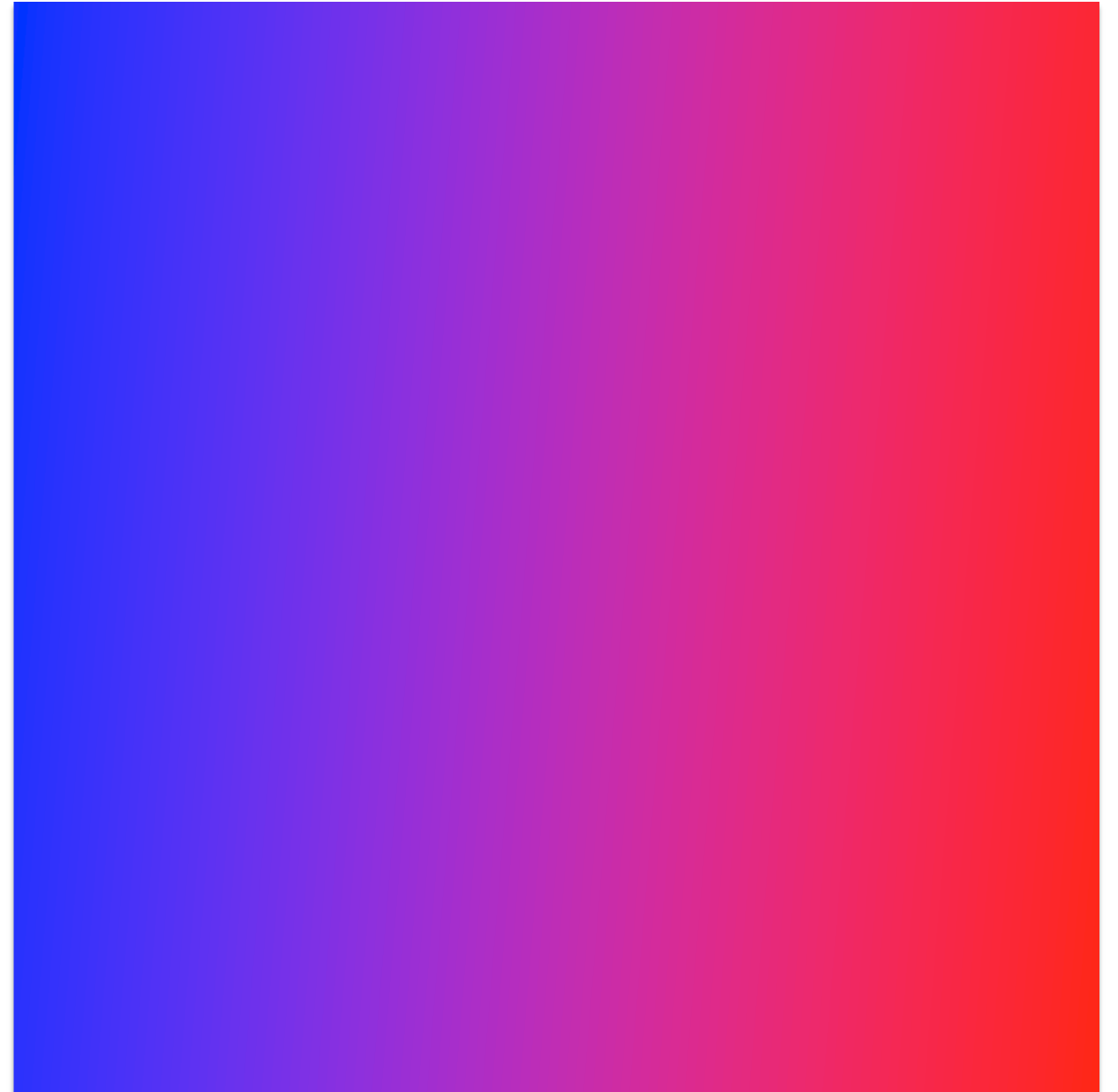
?

$$\sum_{ij} (u_{ij} - u_{i+1,j})^2$$

Which flow field optimizes the objective? $\min_{\mathbf{u}} (u_{i,j} - u_{i+1,j})^2$



big



small

Key idea

(of Horn-Schunck optical flow)

Enforce

brightness constancy

Enforce

smooth flow field

to compute optical flow

bringing it all together...

Horn-Schunck optical flow

$$\min_{\mathbf{u}, \mathbf{v}} \sum_{i, j} \left\{ \begin{array}{l} \text{smoothness} \\ E_s(i, j) \end{array} + \begin{array}{l} \text{brightness constancy} \\ \lambda E_d(i, j) \end{array} \right\}$$

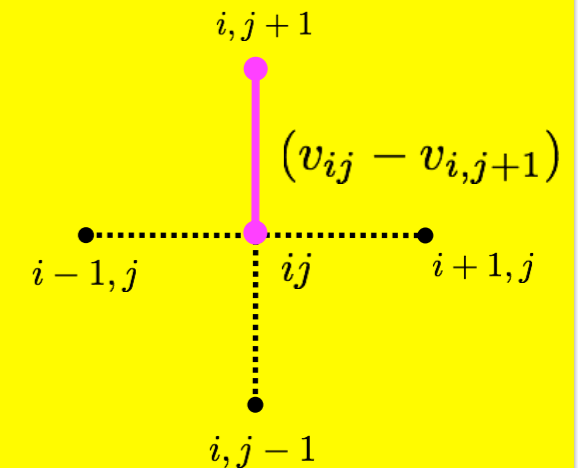
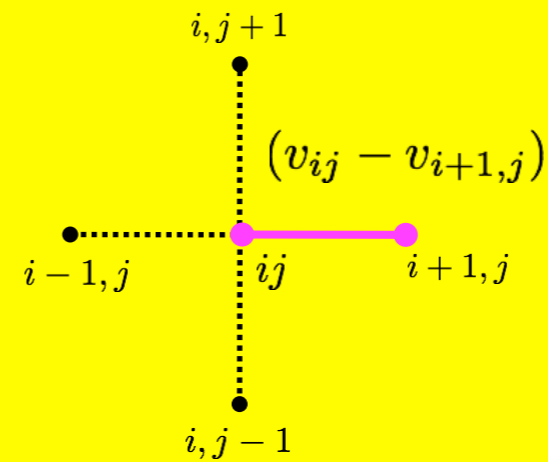
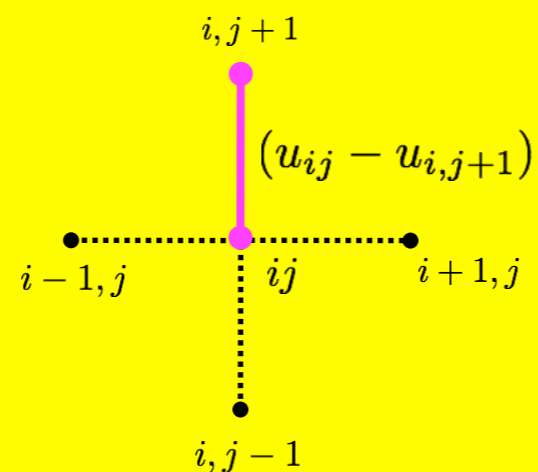
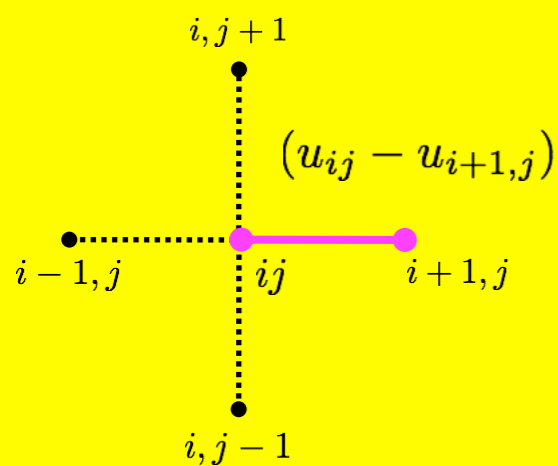
weight

HS optical flow objective function

Brightness constancy $E_d(i, j) = \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2$

Smoothness

$$E_s(i, j) = \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right]$$



How do we solve this minimization problem?

$$\min_{\mathbf{u}, \mathbf{v}} \sum_{i, j} \left\{ E_s(i, j) + \lambda E_d(i, j) \right\}$$

How do we solve this minimization problem?

$$\min_{\mathbf{u}, \mathbf{v}} \sum_{i, j} \left\{ E_s(i, j) + \lambda E_d(i, j) \right\}$$

Compute partial derivative, derive update equations
(gradient decent!)

Compute the partial derivatives of this huge sum!

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

smoothness term

brightness constancy

Compute the partial derivatives of this huge sum!

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

it's not so bad...

$$\frac{\partial E}{\partial u_{kl}} =$$

how many u terms depend on k and l?

Compute the partial derivatives of this huge sum!

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

it's not so bad...

$$\frac{\partial E}{\partial u_{kl}} =$$

how many u terms depend on k and l?

FOUR from smoothness

ONE from brightness constancy

Compute the partial derivatives of this huge sum!

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

it's not so bad...

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

how many u terms depend on k and l?

FOUR from smoothness

ONE from brightness constancy

Compute the partial derivatives of this huge sum!

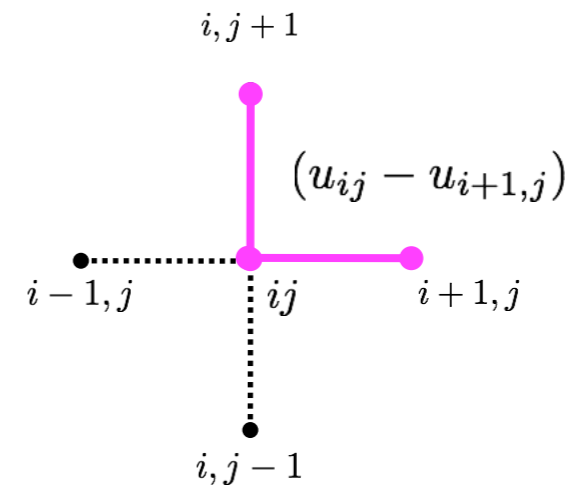
$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$



$$(u_{ij}^2 - 2u_{ij}u_{i+1,j} + u_{i+1,j}^2)$$

$$(u_{ij}^2 - 2u_{ij}u_{i,j+1} + u_{i,j+1}^2)$$

(variable will appear four times in sum)



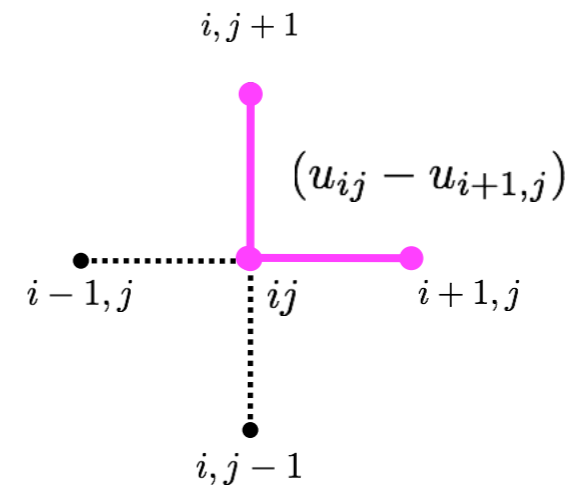
Compute the partial derivatives of this huge sum!

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$



$$(u_{ij}^2 - 2u_{ij}u_{i+1,j} + u_{i+1,j}^2)$$

$$(u_{ij}^2 - 2u_{ij}u_{i,j+1} + u_{i,j+1}^2)$$



(variable will appear four times in sum)

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

$$\frac{\partial E}{\partial v_{kl}} = 2(v_{kl} - \bar{v}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_y$$

short hand for
local average

$$\bar{u}_{ij} = \frac{1}{4} \left\{ u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} \right\}$$

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

$$\frac{\partial E}{\partial v_{kl}} = 2(v_{kl} - \bar{v}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_y$$

Where are the extrema of E?

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

$$\frac{\partial E}{\partial v_{kl}} = 2(v_{kl} - \bar{v}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_y$$

Where are the extrema of E?

(set derivatives to zero and solve for unknowns u and v)

$$(1 + \lambda I_x^2)u_{kl} + \lambda I_x I_y v_{kl} = \bar{u}_{kl} - \lambda I_x I_t$$

$$\lambda I_x I_y u_{kl} + (1 + \lambda I_y^2)v_{kl} = \bar{v}_{kl} - \lambda I_y I_t$$

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

$$\frac{\partial E}{\partial v_{kl}} = 2(v_{kl} - \bar{v}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_y$$

Where are the extrema of E?

(set derivatives to zero and solve for unknowns u and v)

$$(1 + \lambda I_x^2)u_{kl} + \lambda I_x I_y v_{kl} = \bar{u}_{kl} - \lambda I_x I_t$$

$$\lambda I_x I_y u_{kl} + (1 + \lambda I_y^2)v_{kl} = \bar{v}_{kl} - \lambda I_y I_t$$

this is a linear system **$\mathbf{Ax} = \mathbf{b}$** *how do you solve this?*

ok, take a step back, why are we doing all this math?

We are solving for the optical flow (u,v) given two constraints

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

smoothness

brightness constancy

We need the math to minimize this
(back to the math)

Partial derivatives of Horn-Schunck objective function E:

$$\frac{\partial E}{\partial u_{kl}} = 2(u_{kl} - \bar{u}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_x$$

$$\frac{\partial E}{\partial v_{kl}} = 2(v_{kl} - \bar{v}_{kl}) + 2\lambda(I_x u_{kl} + I_y v_{kl} + I_t)I_y$$

Where are the extrema of E?

(set derivatives to zero and solve for unknowns u and v)

$$(1 + \lambda I_x^2)u_{kl} + \lambda I_x I_y v_{kl} = \bar{u}_{kl} - \lambda I_x I_t$$

$$\lambda I_x I_y u_{kl} + (1 + \lambda I_y^2)v_{kl} = \bar{v}_{kl} - \lambda I_y I_t$$

$$\mathbf{Ax} = \mathbf{b} \quad \text{how do you solve this?}$$

$$(1 + \lambda I_x^2)u_{kl} + \lambda I_x I_y v_{kl} = \bar{u}_{kl} - \lambda I_x I_t$$

$$\lambda I_x I_y u_{kl} + (1 + \lambda I_y^2)v_{kl} = \bar{v}_{kl} - \lambda I_y I_t$$

Recall $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} = \frac{\text{adj}\mathbf{A}}{\det \mathbf{A}}\mathbf{b}$

$$(1 + \lambda I_x^2)u_{kl} + \lambda I_x I_y v_{kl} = \bar{u}_{kl} - \lambda I_x I_t$$

$$\lambda I_x I_y u_{kl} + (1 + \lambda I_y^2)v_{kl} = \bar{v}_{kl} - \lambda I_y I_t$$

Recall $\mathbf{x} = \mathbf{A}^{-1} \mathbf{b} = \frac{\text{adj} \mathbf{A}}{\det \mathbf{A}} \mathbf{b}$

Same as the linear system:

$$\underbrace{\{1 + \lambda(I_x^2 + I_y^2)\}}_{(\det \mathbf{A})} u_{kl} = (1 + \lambda I_x^2) \bar{u}_{kl} - \lambda I_x I_y \bar{v}_{kl} - \lambda I_x I_t$$

$$\underbrace{\{1 + \lambda(I_x^2 + I_y^2)\}}_{(\det \mathbf{A})} v_{kl} = (1 + \lambda I_y^2) \bar{v}_{kl} - \lambda I_x I_y \bar{u}_{kl} - \lambda I_y I_t$$

$$\{1 + \lambda(I_x^2 + I_y^2)\}u_{kl} = (1 + \lambda I_x^2)\bar{u}_{kl} - \lambda I_x I_y \bar{v}_{kl} - \lambda I_x I_t$$

$$\{1 + \lambda(I_x^2 + I_y^2)\}v_{kl} = (1 + \lambda I_y^2)\bar{v}_{kl} - \lambda I_x I_y \bar{u}_{kl} - \lambda I_y I_t$$

Rearrange to get update equations:

$$\hat{u}_{kl} = \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x$$

new value
old average

$$\hat{v}_{kl} = \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

Recall: $\min_{\mathbf{u}, \mathbf{v}} \sum_{i,j} \left\{ E_s(i, j) + \lambda E_d(i, j) \right\}$

When lambda is small (lambda inverse is big)...

$$\hat{u}_{kl} = \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x$$

new value old average

$$\hat{v}_{kl} = \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

Recall: $\min_{\mathbf{u}, \mathbf{v}} \sum_{i,j} \left\{ E_s(i, j) + \lambda E_d(i, j) \right\}$

When lambda is small (lambda inverse is big)...

$$\hat{u}_{kl} = \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x$$

new value
old average

$$\hat{v}_{kl} = \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

goes to zero
goes to zero

Recall: $\min_{\mathbf{u}, \mathbf{v}} \sum_{i,j} \left\{ E_s(i,j) + \lambda E_d(i,j) \right\}$

When lambda is small (lambda inverse is big)...

$$\hat{u}_{kl} = \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x$$

new value
old average

$$\hat{v}_{kl} = \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

...we only care about smoothness.

ok, take a step back, why did we do all this math?

We are solving for the optical flow (u,v) given two constraints

$$\sum_{ij} \left\{ \frac{1}{4} \left[(u_{ij} - u_{i+1,j})^2 + (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i+1,j})^2 + (v_{ij} - v_{i,j+1})^2 \right] + \lambda \left[I_x u_{ij} + I_y v_{ij} + I_t \right]^2 \right\}$$

smoothness brightness constancy

We needed the math to minimize this
(now to the algorithm)

Horn-Schunck Optical Flow Algorithm

1. Precompute image gradients I_y I_x
2. Precompute temporal gradients I_t
3. Initialize flow field $\mathbf{u} = \mathbf{0}$
 $\mathbf{v} = \mathbf{0}$
4. While not converged

Compute flow field updates for each pixel:

$$\hat{u}_{kl} = \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x \quad \hat{v}_{kl} = \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

Just 8 lines of code!

References

Basic reading:

- Szeliski, Section 8.4.