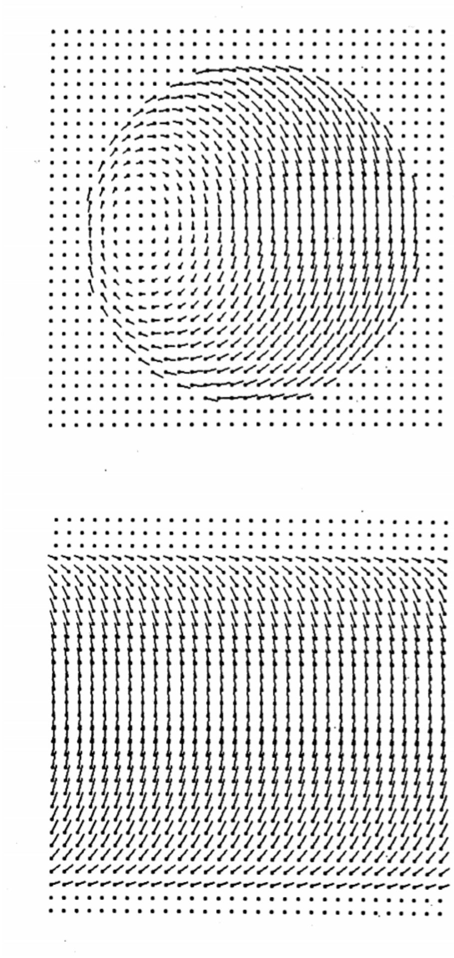
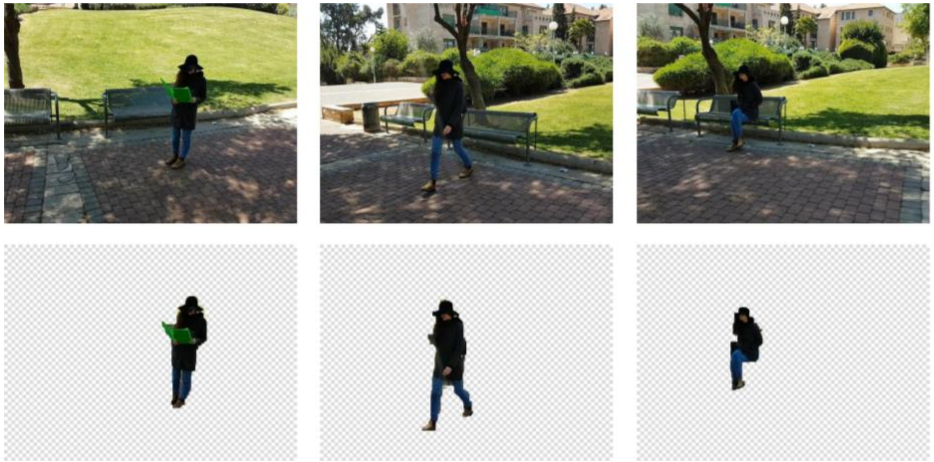
Class 8

Optical Flow

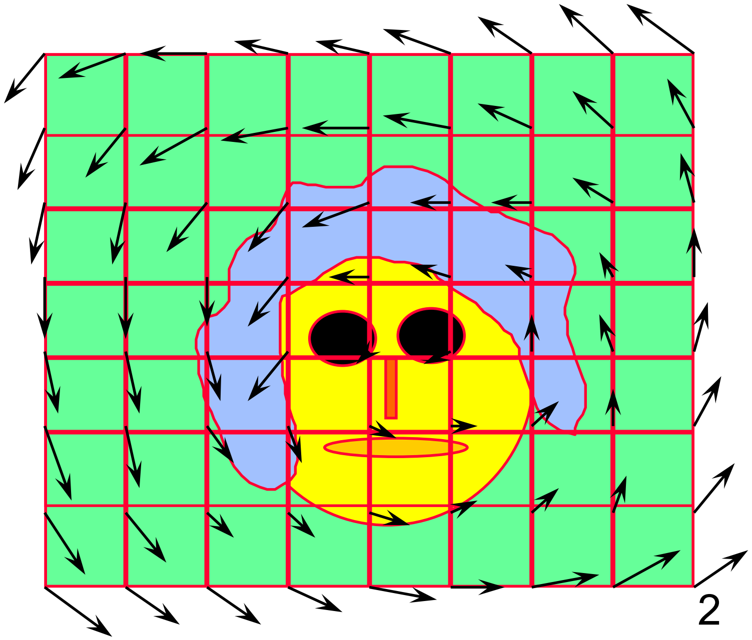


Change Detection

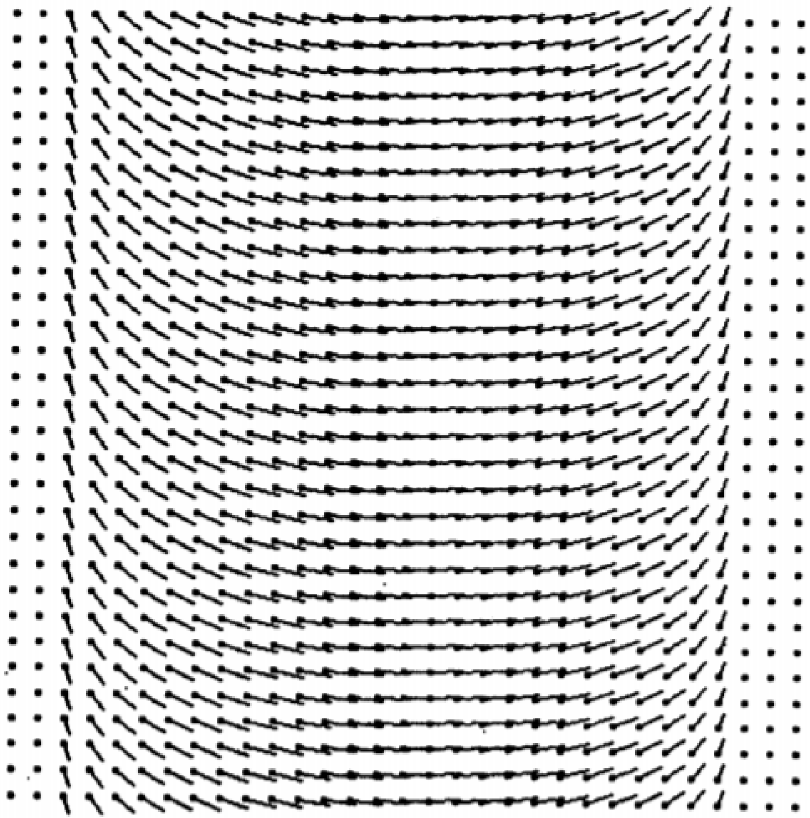


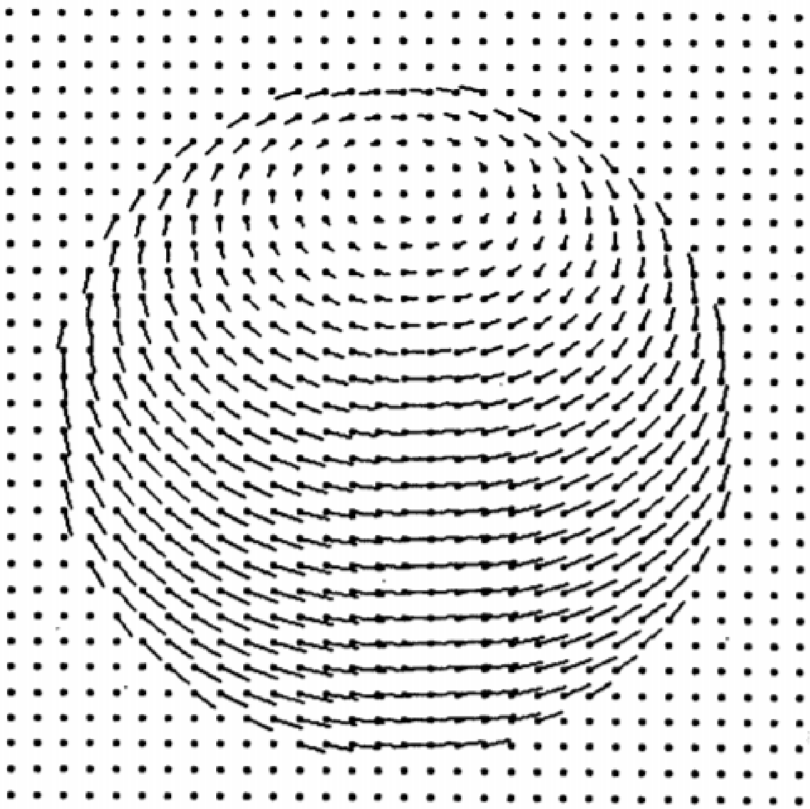
Optical Flow

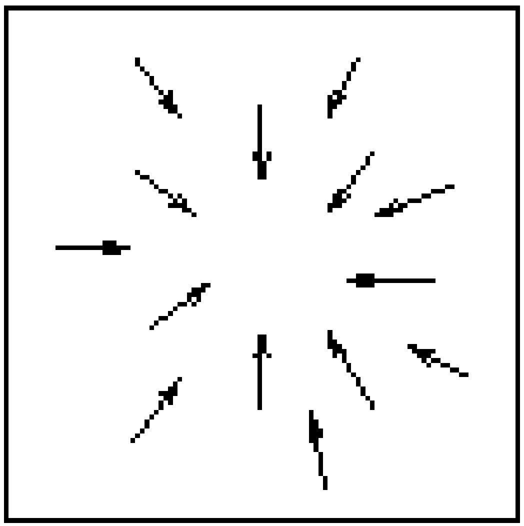
- Pixel motion between consecutive frames:
  - Caused by camera or object motion
- Introduced by James J. Gibson 1940

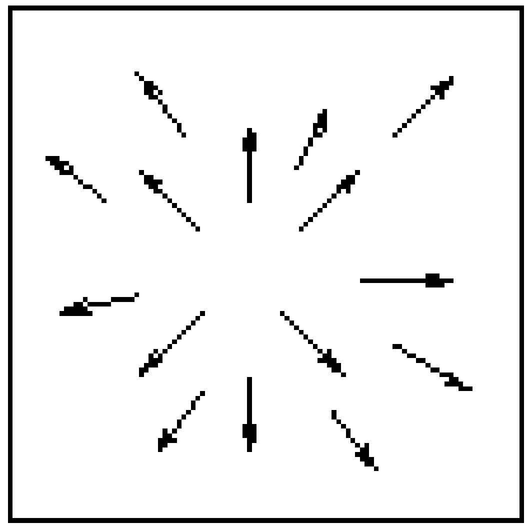


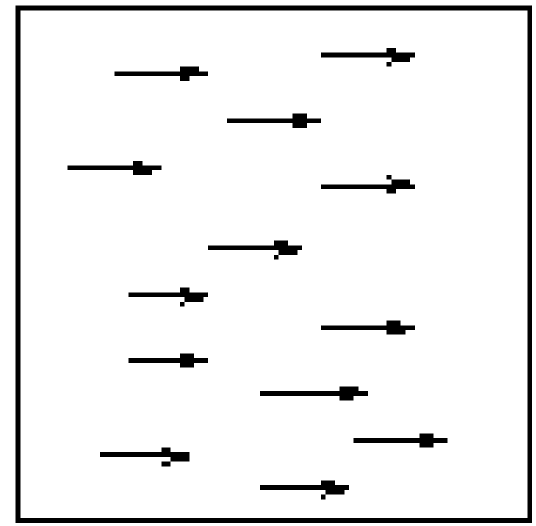
What is Moving and How?



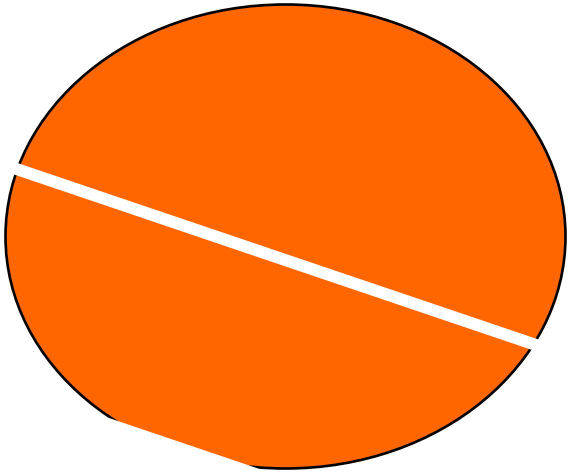



Zoom out


Zoom in


Pan Right to Left

Aperture Problem



Optical Flow: Gradient Based

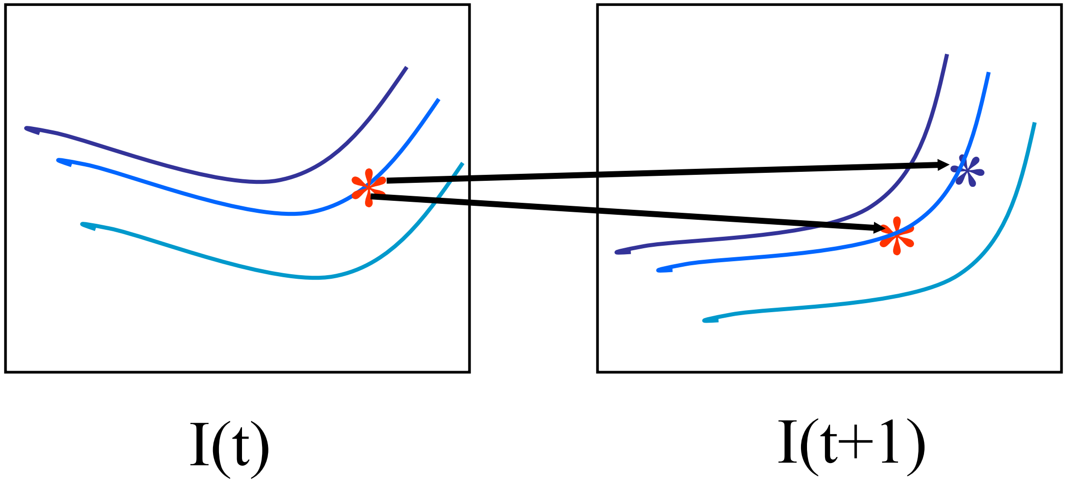
Assumptions:

- The movement is small
- Brightness constancy assumption (BCA):
  - The intensity of a given object point does not change between frames
  - $I(x+dx, y+dy, t+dt) = I(x, y, t)$

הבינתחומי הרצליה
IDC HERZLIYA

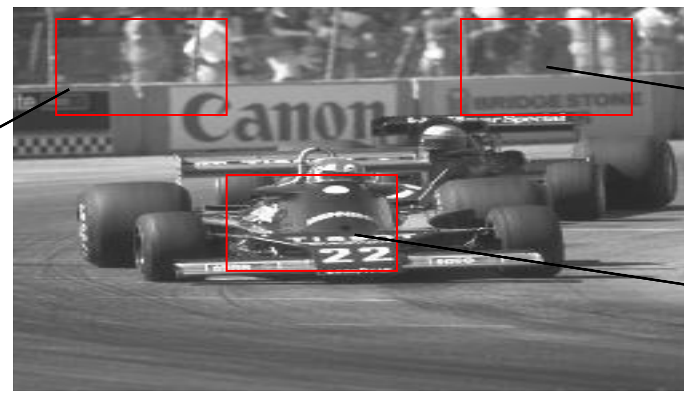
Ambiguity

- $I(x+dx, y+dy, t+dt) = I(x, y, t)$
- Brightness constancy assumption: insufficient!



Solution of Ambiguity

- Local constraint
  - Assume constant motion in a small local window



- How to use local constraint?
  - Naïve search: expensive!

IDC HERZLIYA הבינתחומי הרצליה

Gradient Based Method

- Relates spatial and temporal gradients
- Assumptions:
  - First order approximation of the flow: $u(p)$ and $v(p)$ are small
  - $u(p)$ and $v(p)$ are constant (or smooth) in a small neighborhood of p

An under-determined problem (aperture Problem)

Lucas–Kanade Algorithm

- Optical flow computation
- Gradient based algorithm

Optical Flow Equation

- Taylor Series for a pixel $p = (p_x, p_y)$

$$I(p_x + dx, p_y + dy, t + dt)$$

$$= I(p_x, p_y, t) + \frac{\partial I}{\partial x}(p)dx + \frac{\partial I}{\partial y}(p)dy + \frac{\partial I}{\partial t}(p)dt + \cancel{\ldots}$$

- Brightness constancy assumption:
  - $I(p_x + dx, p_y + dy, t + dt) = I(p_x, p_y, t)$
  - $\frac{\partial I}{\partial x}(p)dx + \frac{\partial I}{\partial y}(p)dy + \frac{\partial I}{\partial t}(p)dt = 0$
  - Notation: $\boxed{I_x(p)dx + I_y(p)dy = -I_t(p)dt}$

IDC HERZLIYA

Optical Flow Equation

- $I_x(p)dx + I_y(p)dy = -I_t(p)dt$
- Divide by dt
  - $I_x(p)\frac{dx}{dt} + I_y(p)\frac{dy}{dt} = -I_t(p)$
  - $\boxed{I_x(p)u(p) + I_y(p)v(p) = -I_t(p)}$
- Matrix notation: $\left(I_x(p), I_y(p)\right)\begin{pmatrix} u(p) \\ v(p) \end{pmatrix} = -I_t(p)$

Unknowns

הבינתחומי הרצליה
IDC HERZLIYA

Local Constant Flow

- For each p_i we have:

$$\left(I_x(p_i), I_y(p_i)\right)\begin{pmatrix} u(p_i) \\ v(p_i) \end{pmatrix} = -I_t(p_i)$$

- Let $w(p_0)$ be a small patch around p_0
- Assume constant motion $\forall p_i \in w(p_0)$:

$$\begin{pmatrix} I_x(p_1), I_y(p_1) \\ I_x(p_2), I_y(p_2) \\ I_x(p_3), I_y(p_3) \end{pmatrix}\begin{pmatrix} u(p_0) \\ v(p_0) \end{pmatrix} = -\begin{pmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_k) \end{pmatrix}$$

Local Constant Flow

Let $p_1 \dots p_k \in w(p_0)$:

$$\underbrace{\begin{pmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_k) & I_y(p_k) \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} u(p_0) \\ v(p_0) \end{pmatrix} = - \underbrace{\begin{pmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_k) \end{pmatrix}}_{\mathbf{b}}$$

Is there a problem?

That is: $A \begin{pmatrix} u(p_0) \\ v(p_0) \end{pmatrix} = b \quad \Rightarrow \quad \begin{pmatrix} u(p_0) \\ v(p_0) \end{pmatrix} = A^+ b$

$$\boxed{A^+ = (A^T A)^{-1} A^T}$$

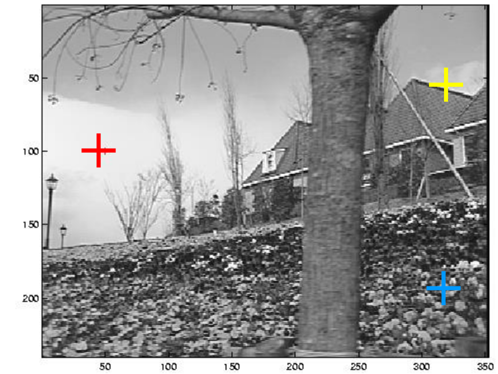
Algebra: definition of C

- $A = \begin{pmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ & \vdots & \\ I_x(p_k) & I_y(p_k) \end{pmatrix}$

- $A^T A = \begin{pmatrix} I_x(p_1) & I_x(p_2) \dots I_x(p_k) \\ I_y(p_1) & I_y(p_2) \dots I_y(p_k) \end{pmatrix} \begin{pmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ & \vdots & \\ I_x(p_k) & I_y(p_k) \end{pmatrix}$

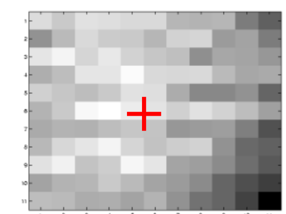
$$= \begin{pmatrix} \sum I_x^2(p_i) & \sum I_x(p_i) I_y(p_i) \\ \sum I_x(p_i) I_y(p_i) & \sum I_x^2(p_i) \end{pmatrix}$$

Cases

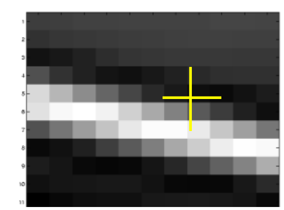


$$A\begin{pmatrix}u(p_0)\\v(p_0)\end{pmatrix}=b \quad\longrightarrow\quad A^TA\begin{pmatrix}u(p_0)\\v(p_0)\end{pmatrix}=A^Tb$$

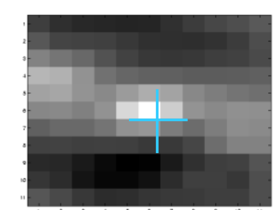
Let $C=A^TA=\begin{pmatrix}\sum I_x^2(p_i) & \sum I_x(p_i)I_y(p_i)\\ \sum I_x(p_i)I_y(p_i) & \sum I_y^2(p_i)\end{pmatrix}$



- $rank(C) = 0$ blank wall problem



- $rank(C) = 1$ aperture problem
- $rank(C) = 2$ enough texture



The Algorithm

- Smooth the image in the special domain
- Smooth the image in the temporal domain (not always necessary)
- For each pixel, p_0:
  - Compute $A(p_0)$, $b(p_0)$, and $C(p_0)$
  - If $rank(C(p_0)) = 2$, compute $u(p_0)$ and $v(p_0)$ by: $\begin{pmatrix} u(p_0) \\ v(p_0) \end{pmatrix} = A^+ b$

IDC HERZLIYA הבינתחומי הרצליה

Modification

- Replace Σ in $c(p_0) = \begin{pmatrix} \sum I_x^2(p_i) & \sum I_x(p_i)I_y(p_i) \\ \sum I_x(p_i)I_y(p_i) & \sum I_y^2(p_i) \end{pmatrix}$

$p_i \in w(p_0)$

by Convolution with Gaussian:

  - $E.g., \quad \sum I_x^2(p_i) \Rightarrow (G * I_x^2)(p_0)$

$(G * I_x^2)$ is a matrix

- $c(p_0) = \begin{pmatrix} (G * I_x^2)(p_0) & (G * I_x I_y)(p_0) \\ (G * I_x I_y)(p_0) & (G * I_y^2)(p_0) \end{pmatrix}$

הבינתחומי הרצליה
IDC HERZLIYA

What can go wrong?

- Brightness constancy is **not** satisfied
- The motion is **not** small
- The motion is **not** translation
- A point does **not** move like its neighbors
  - window size is too large
  - what is the ideal window size?

Next

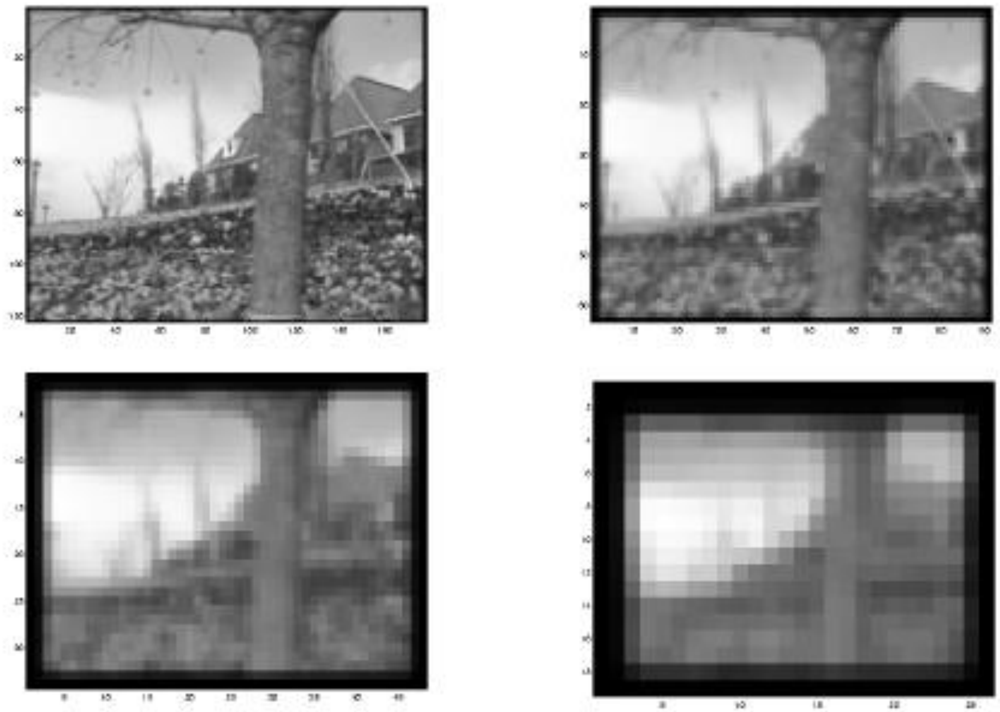
- Dealing with large motion
  - Use a pyramid of OF
- More general motion:
  - Affine rather than just translation
- Global solutions

Next

- Dealing with large motion
  - Use a pyramid of OF
- More general motion:
  - Affine rather than just translation
- Global solutions

Large Motion

- Reduce the Resolution
- What may be the problem?



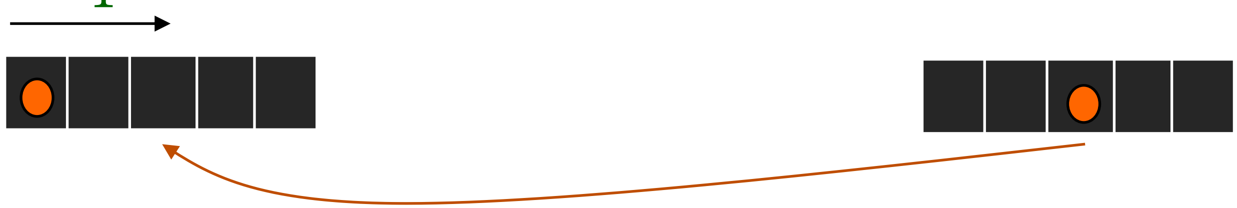
Taken from Dyer

Solution: Use a Pyramid

- 1D: basic idea

$$v = v`_1 + v_2 = 4 + 1 = 5$$

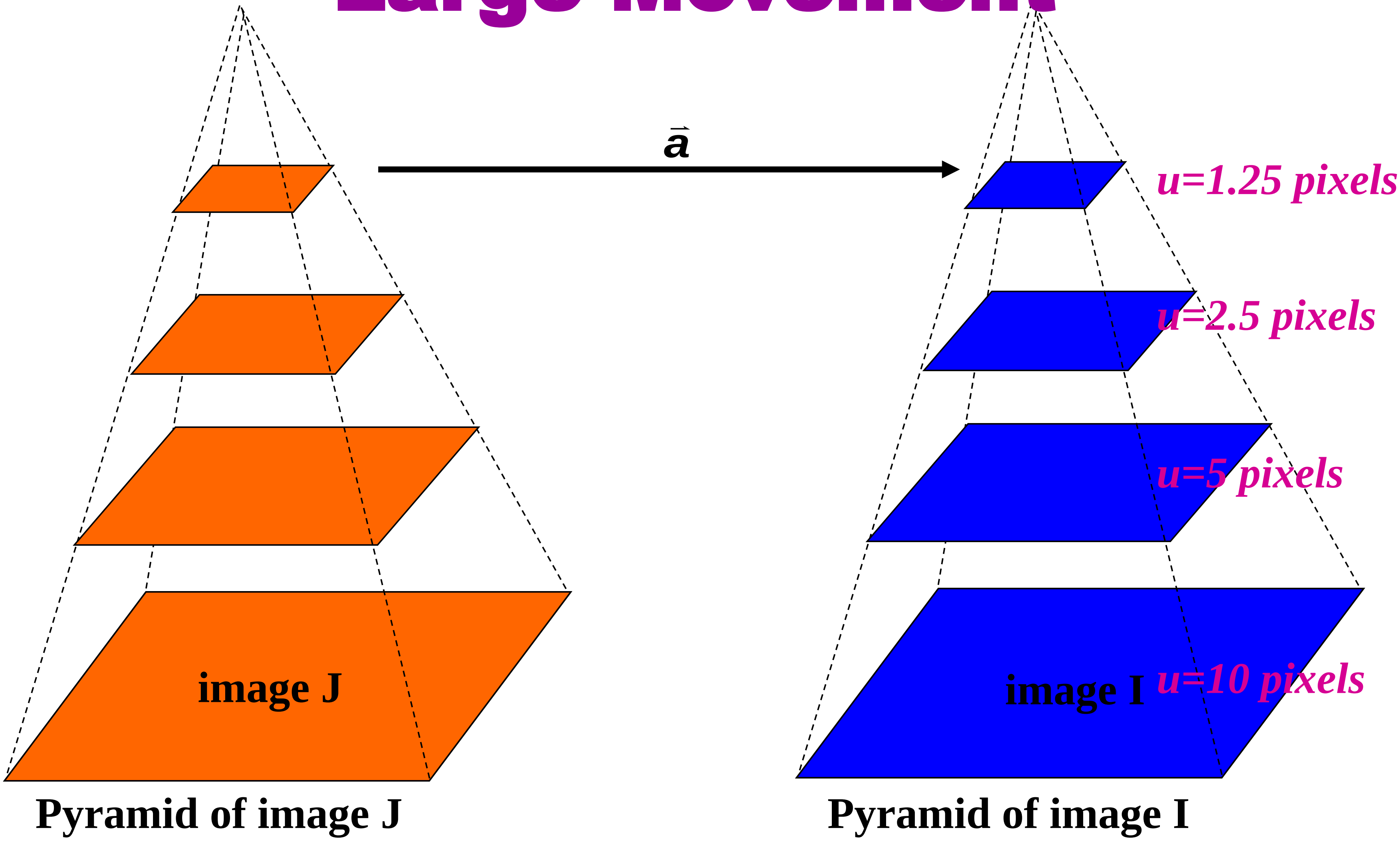
$v_1 = 2$

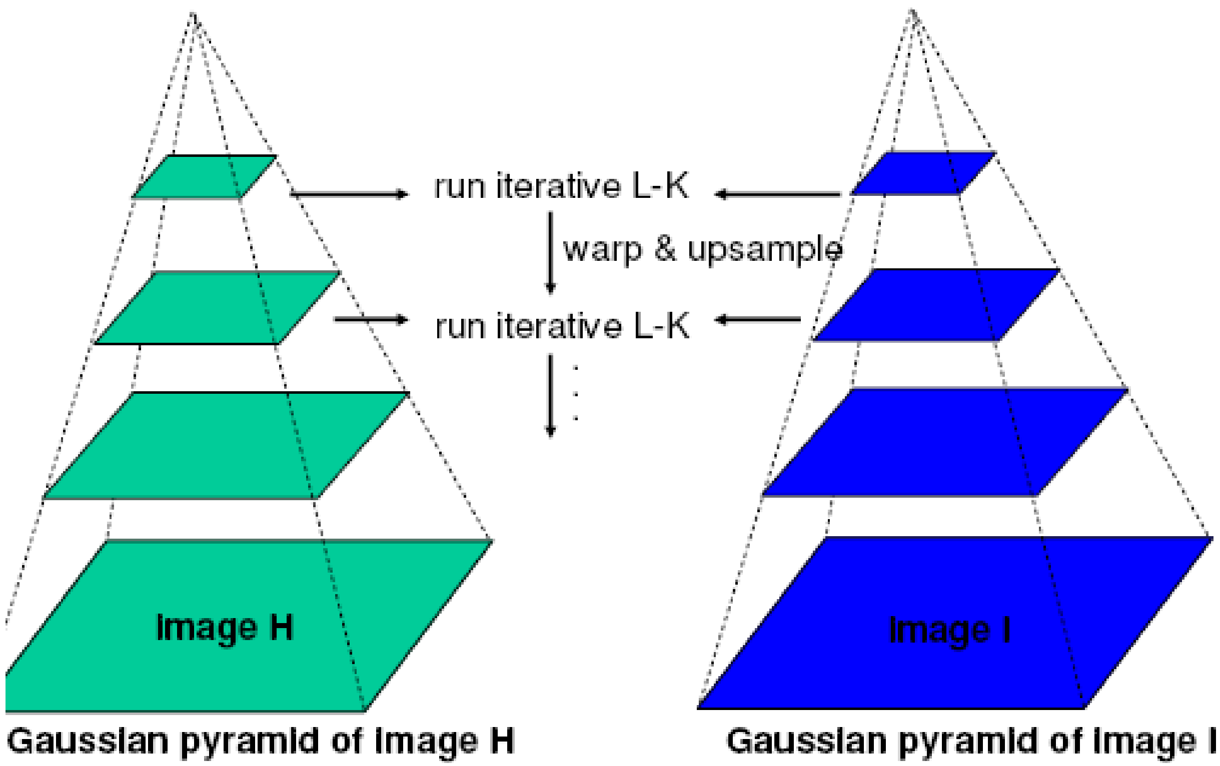
Resize  Resize

Back to the original image
$v`_1 = 4$, warp -4

$v_2 = 1$ $v = 5$, too large!

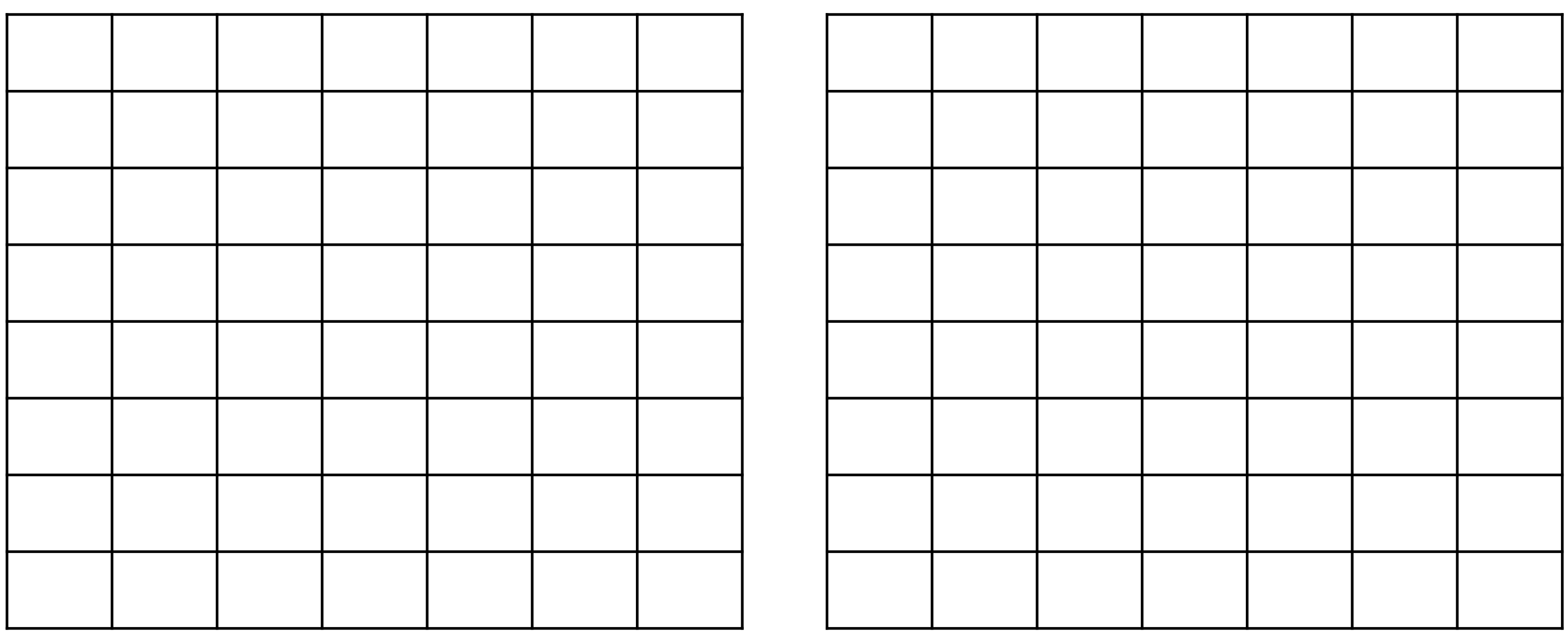
Large Movement




run iterative L-K
warp & upsample
run iterative L-K
Image H
Image I
Gaussian pyramid of image H
Gaussian pyramid of image I

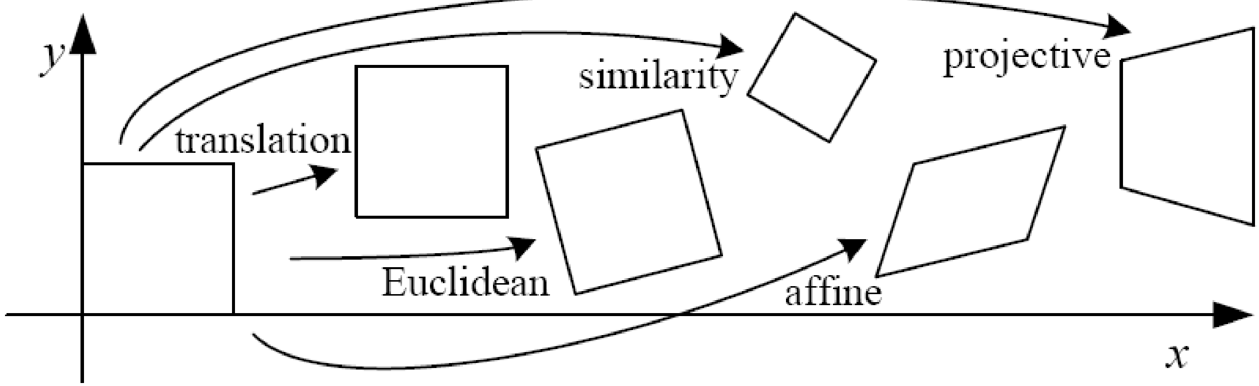
Implementation

- Warp function:
  - Generate a new image based on OF
  - How?



הבינתחומי הרצליה
IDC HERZLIYA

2D Motion Models



Name	Matrix	# D.O.F.	Preserves:	Icon
translation	$\begin{bmatrix} \boldsymbol{I} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	2	orientation $+\cdots$	
rigid (Euclidean)	$\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	3	lengths $+\cdots$	
similarity	$\begin{bmatrix} s\boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	4	angles $+\cdots$	
affine	$\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$	6	parallelism $+\cdots$	
projective	$\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

These transformations are closed under composition and inverse is a member

Taken from: http://graphics.cs.cmu.edu/courses/15-463/2006_fall/www/Lectures/direct-alignment.pdf

Example: Affine Motion

- Affine, $p = (x, y)^T$: $Ap + T$
  - $u(p) = u(x, y) = a_1 + a_2 x + a_3 y$
  - $v(p) = v(x, y) = a_4 + a_5 x + a_6 y$
- Brightness constancy assumption:
  - $I_x(p)u(p) + I_y(p)v(p) = -I_t(p)$
  - $-I_t(x, y) =$
    $$I_x(x, y)(a_1 + a_2 x + a_3 y) + I_y(x, y)(a_4 + a_5 x + a_6 y)$$
- Least Square Minimization: search for a_i

IDC HERZLIYA הבינתחומי הרצליה

Cont.

$$B\vec{a} = -\vec{I}_t$$

- Search for $\vec{a} = (a_1, \dots a_6)$ to minimize the least square of the BC assumption

$$Err(\vec{a}) = \sum((a_1 + a_2 x + a_3 y)I_x + (a_4 + a_5 x + a_6 y)I_y + I_t)^2$$

- Matrix notation:

$$\underbrace{\begin{pmatrix} I_x(p_1), x_1 I_x(p_1), y_1 I_x(p_1), I_y(p_1), x_1 I_y(p_1), y_1 I_y(p_1) \\ \vdots \\ I_x(p_k), x_k I_x(p_k), y_k I_x(p_k), I_y(p_k), x_k I_y(p_k), y_k I_y(p_k) \end{pmatrix}}_{\substack{\mathrm{B} \\ k \times 6}} \underbrace{\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_6 \end{pmatrix}}_{\vec{a}} = -\underbrace{\begin{pmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_k) \end{pmatrix}}_{\vec{I_t}}$$

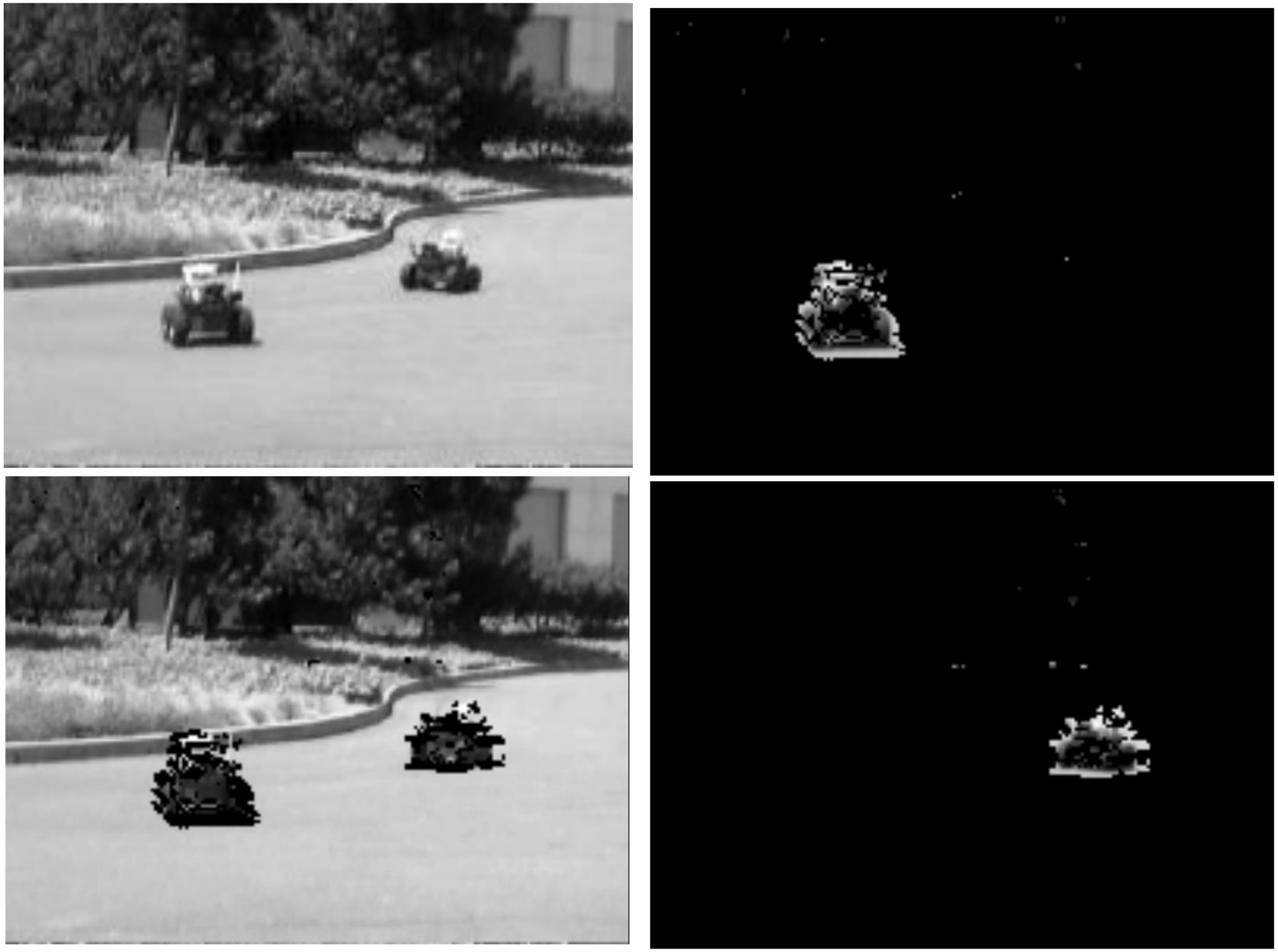
Cont.

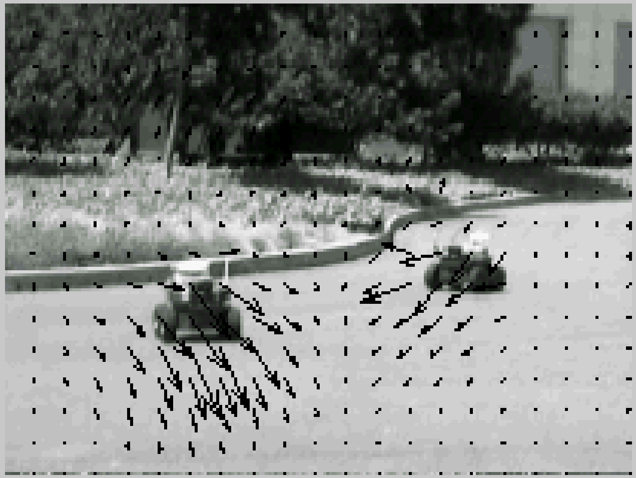
- We can compute: B and $\vec{I}_t$
- We want to solve for $\vec{a}$:
  - $\underset{k \times 6}{B}\vec{a} = -\vec{I}_t$
  - $\underset{6 \times 6}{B^T B}\vec{a} = -\underset{6 \times k}{B^T}\vec{I}_t$
  - $\vec{a} = -(B^T B)^{-1} B^T \vec{I}_t = -\underset{6 \times k}{B^+}\vec{I}_t$

Is there a problem?

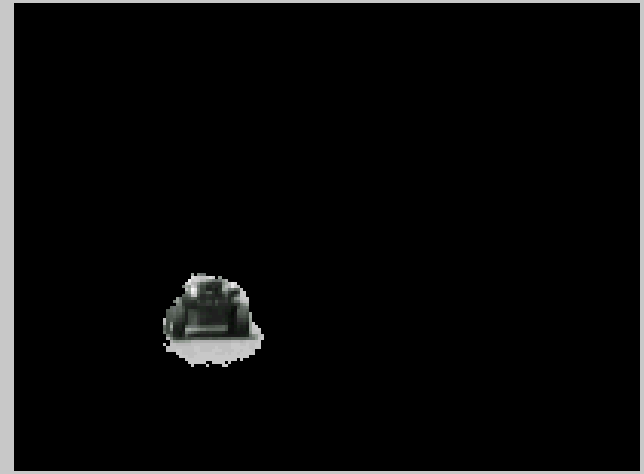
Solve it!

Results: affine motion segmentation



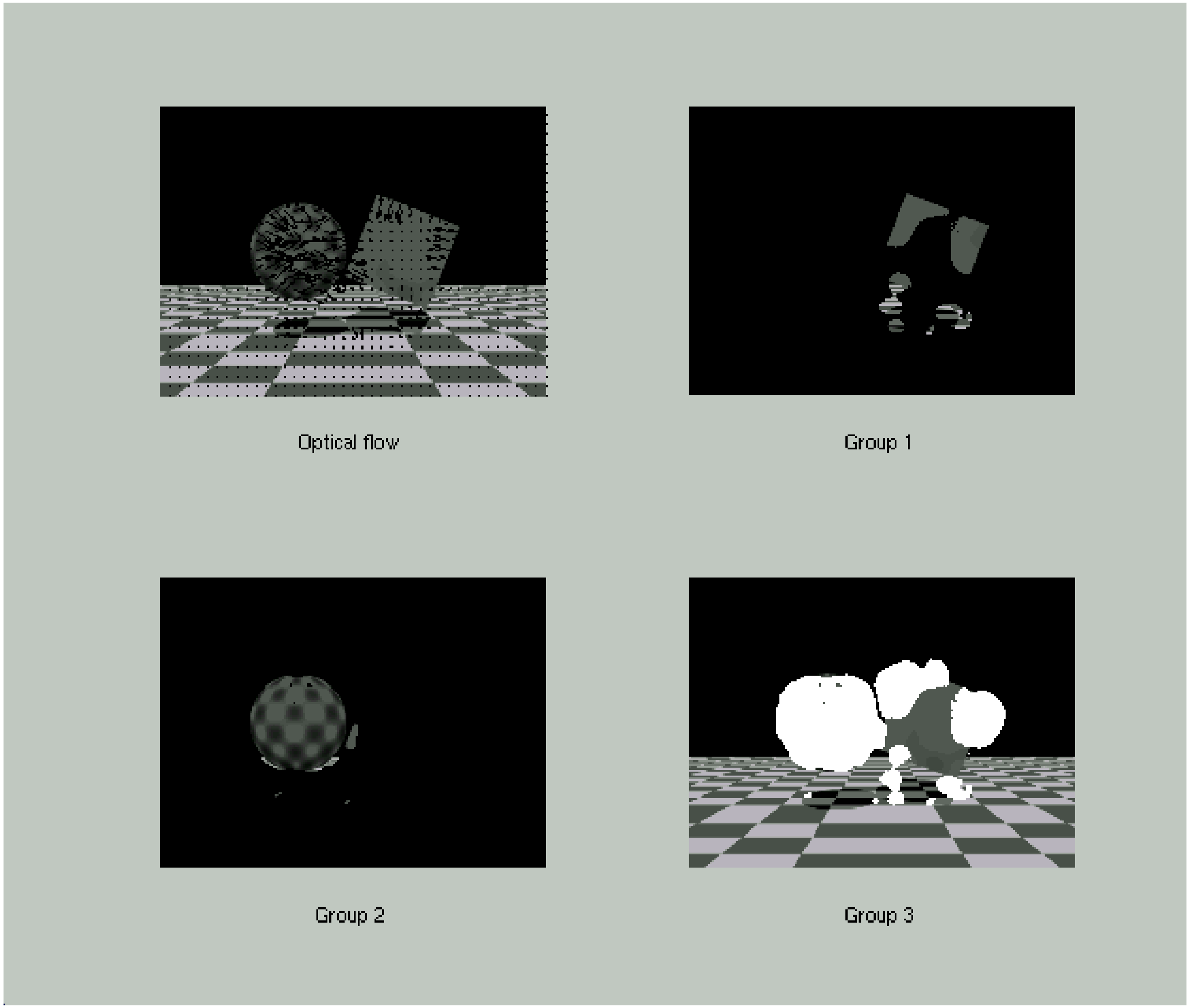


Optical flow



Group 1


Optical flow
Group 1
Group 2
Group 3


Optical flow
Group 1
Group 2
Group 3

IDC HERZLIYA
הבינתחומי הרצליה

Global Solutions

- Knowledge about the scene
  - For example: smooth, piecewise smooth, planar …
- Knowledge about the motion
  - For example: rigid, piecewise rigid, smooth, piecewise smooth…
- Use global optimization

Example: Global Solutions

- Minimize the functional:

$$\underbrace{\iint_{x\,y} (u \cdot I_x + v \cdot I_y + I_t)^2 dxdy}_{\text{Brightness Constancy assumption}} + \underbrace{\iint_{x\,y} \left(\frac{dv}{dx}\right)^2 + \left(\frac{dv}{dy}\right)^2 + \left(\frac{du}{dx}\right)^2 + \left(\frac{du}{dy}\right)^2 dxdy}_{\text{Smoothness of the optical flow}}$$

- Solve *u(x,y)* and *v(x,y)* using any optimization method (not in this course)
- See classic paper by Horn & Schunk 1980

הבינתחומי הרצליה
IDC HERZLIYA

Possible Applications

- Segmentation
- Object tracking
- 3D shape reconstruction
- Align images (mosaics)
- MPEG compression
- Super-resolution
- Correct for camera jitter (stabilization)
- Many more...

Summary

- Some psychophysics
- Optical flow:
  - Aperture problem
  - Lucas Kanade algorithm
- Next:
  - Change detection
  - Tracking

Next

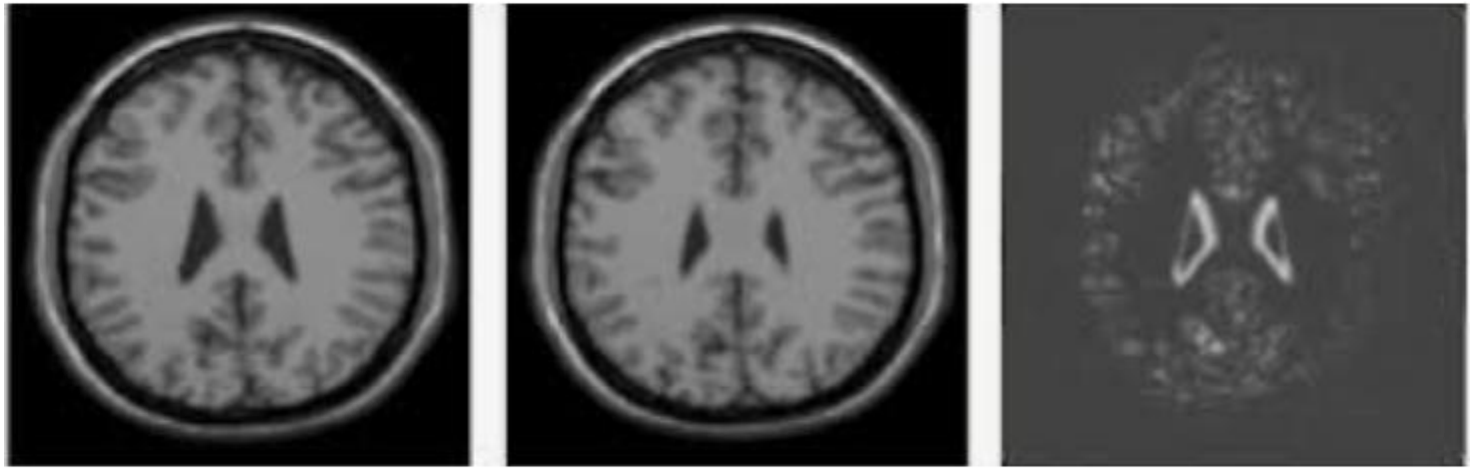
- Detect changes in natural scenes

Motion/Change Detection

- Surveillance
- Attention
- Pre-processing
  - Segmentation
  - Object recognition
  - Tracking
  - Action recognition
  - …

Large Scale

Medical Imaging



Object Segmentation

- Using background subtraction

Another Application

הבינתחומי הרצליה
IDC HERZLIYA

Making life simple

Borrowed from Alexei Efros, CMU, Fall 2005

Simple Change Detection

- Problem: detect the moving regions in the scene
- Input: a sequence of frames taken from a fixed camera
  (e.g., by a surveillance camera)
- Output: for *each pixel* determine if it belongs to the foreground or background
- Extension:
  - Moving camera
  - Segmentation
  - Regions rather than pixels

הבינתחומי הרצליה
IDC HERZLIYA

Background Subtraction

Input:

- A sequence of n images: $I_1, I_2, \ldots, In$
- A threshold: th

Naïve algorithm:

1. Learn the background, I_B
2. Compute the point wise difference
   $d_k(p) = |I_B(p) - I_k(p)|$
3. If $|d_k(p)| > th$ label p as moving

How?

Maybe other measure?

How to define?

Background Model: Pixel

- A single image

Which frame to use?
When is it enough?

- Statistic on a set of frames
  - Average
  - Median
  - A Gaussian
  - A (weighted) mixture of Gaussians

How to compute?
How to update?

Borrowed from Alexei Efros, CMU, Fall 2005

Average image

Median Image

Borrowed from Alexei Efros, CMU, Fall 2005

Average/Median Image

Borrowed from Alexei Efros, CMU, Fall 2005

Background Subtraction

\-

=

Borrowed from Alexei Efros, CMU, Fall 2005

Limitations

- A global threshold
  - Set a different threshold to each pixel
- Setting the threshold
  - Set it automatically
- Changing background
  - Update it
- Cope with multiple modal background
  - Keep multimodal

Background

- Take the median / average of k last frames
- Update every s frames

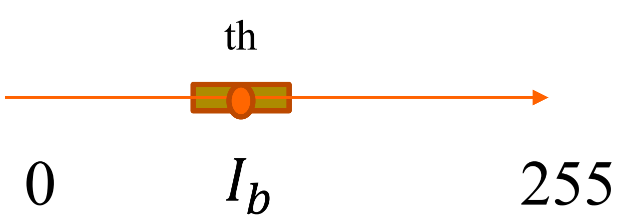
Parametric Pixel Modeling

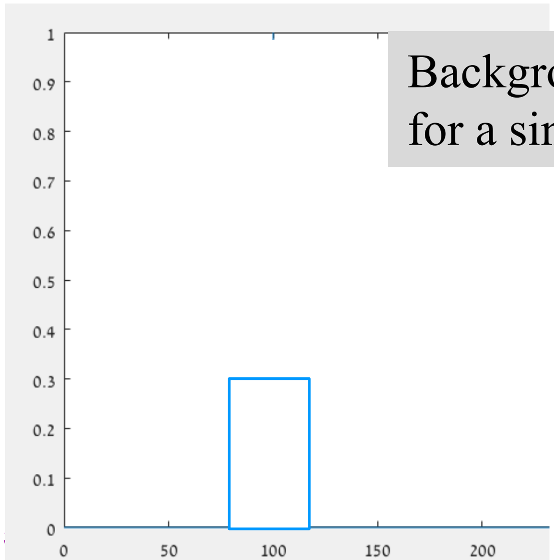
הבינתחומי הרצליה
IDC HERZLIYA

Background Distribution

- Let $P_B(x) = P(x|x \in B)$ be the probability distribution function (*pdf*) of the background for a pixel q
- Given a threshold α, and $x_t = I(q,t)$:

$$F(x_t) = \begin{cases} 1 & P_b(x_t) < \alpha \\ 0 & P_b(x_t) \geq \alpha \end{cases}$$





Background probability for a single pixel

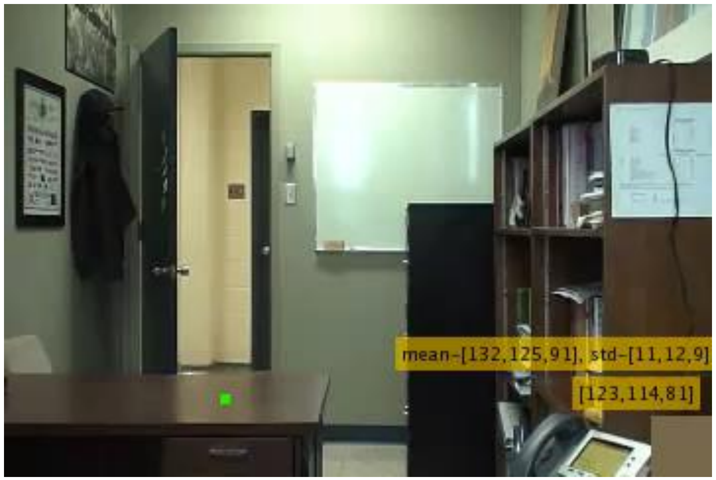
הבינתחומי הרצליה
IDC HERZLIYA

Background Distribution

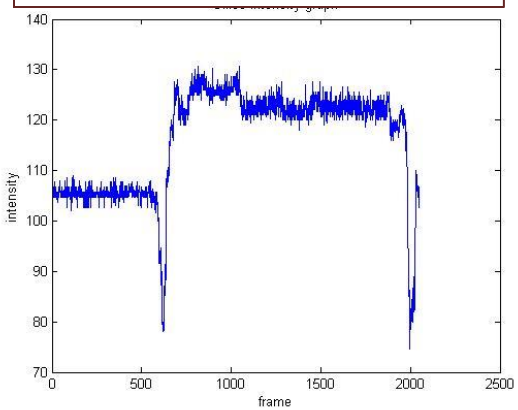
- Let q be a pixel, and $\mathrm{x_t} = I(q,t)$ be the intensity of q at time t
- Define the probability distribution function (*pdf*) of the background, using $\{I(q,t)\}$:

$$P_B(x) = P(x|x \in B)$$

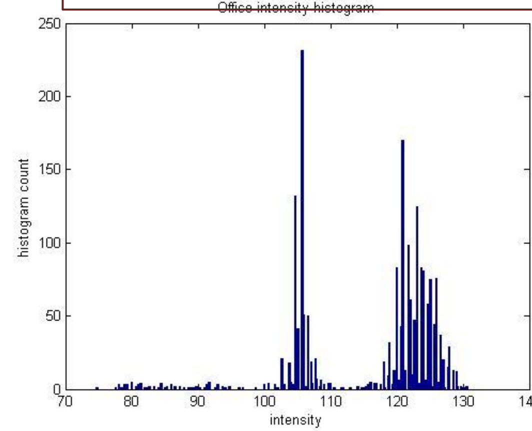
Background Pixel



Intensity: $I(q,t)$

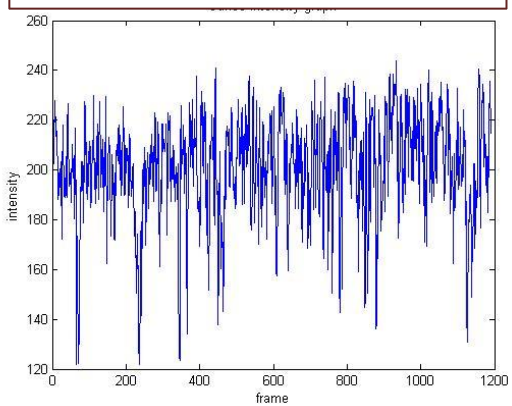


histogram

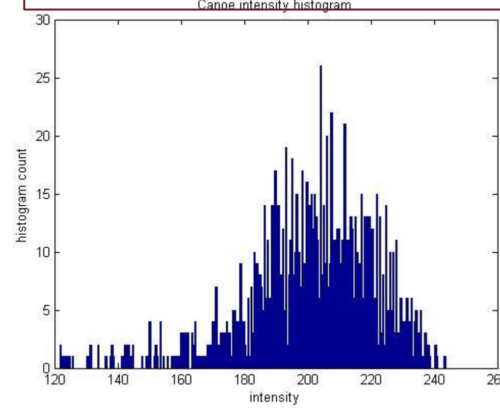


Example 2

Intensity: $I(q,t)$



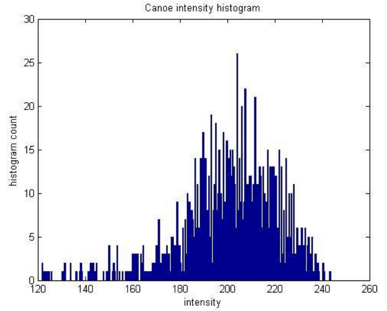
Histogram

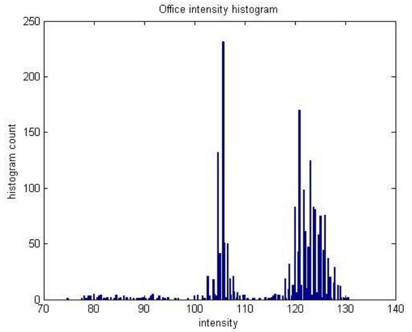


Model the *pdf*

- Let $x_t = I(q,t)$
- Regard the histogram of $\{x_t\}_{t=0}^{n}$ as a *pdf* of q background
- Parametric models:
  - 1D/3D Gaussian
  - Multi modal Gaussians
  - Others…
- Non-parametric models

Under which assumption is it true?





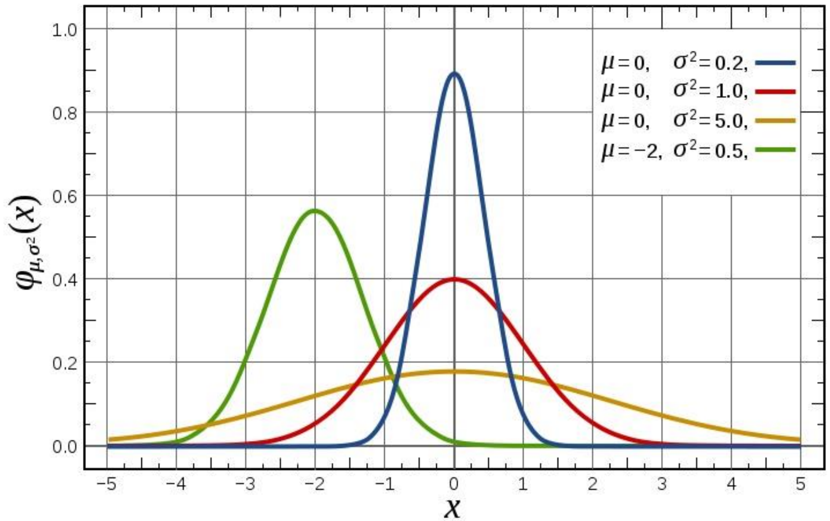
Model: 1D Gaussian

- Assume: independent Gaussian noise in the sampling process
- Parameters:
  - μ - mean (expectation)
  - σ - STD
  - σ^2 - Variance

$$G(x,\sigma) = \frac{1}{2\pi\sigma^2} \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\mu = E(x)$$

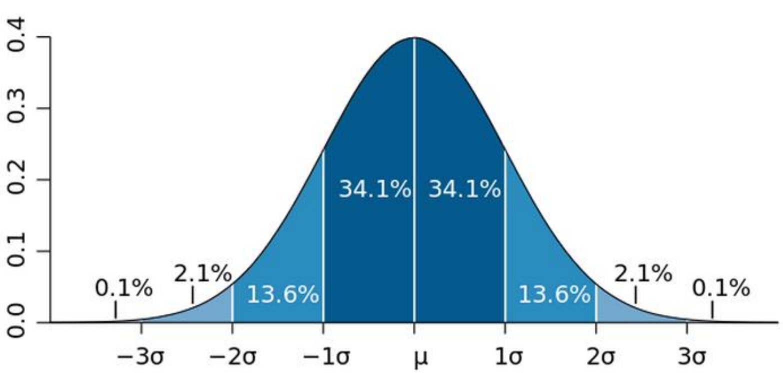
$$\sigma^2(x) = E[(x-\mu)^2]$$



How to set the threshold α?

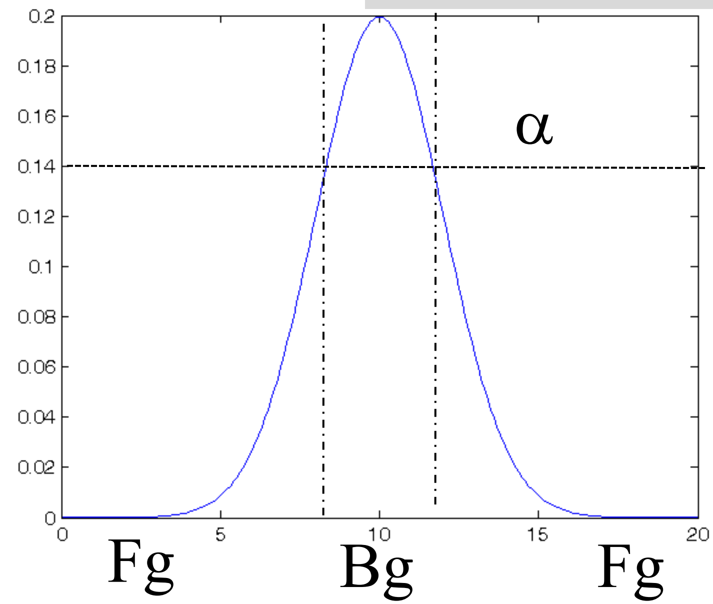
- $x_t = I(q,t)$
- $F(x_t) = \begin{cases} 1 & P_b(x_t) < \alpha \\ 0 & P_b(x_t) \geq \alpha \end{cases}$
- A common choice:

  $\alpha = 2.5\sigma_{i,t}$



Where are the failures?

Background probability for a single pixel



Gaussian Mixture Model

(based on Stauffer et. al. 1999)

- Motivation:
  - Moving background – e.g., trees
  - Single Gaussian is insufficient
- Use mixture of Gaussian:

K Depends on memory and computational power

$$P(x_t \mid B) = \sum_{i=1}^{K} w_{i,t} G(x_t, \mu_{i,t}, \sigma_{i,t})$$

  - K: number of Gaussians
  - $w_{i,t}$: weight of the i^{th} Gaussian at time t
  - $\mu_{i,t}$ *and* $\sigma_{i,t}$: the i Gaussian parameters

הבינתחומי הרצליה
IDC HERZLIYA

Issues

- How to use the set of K Gaussians
- How to initialize the Gaussian parameters:

  For a single Gaussian, i :
  - Average: $\mu_{i,t} = average(\{x_t\})$
  - Variance: $\sigma^2 = E[(x_t - \mu)^2]$

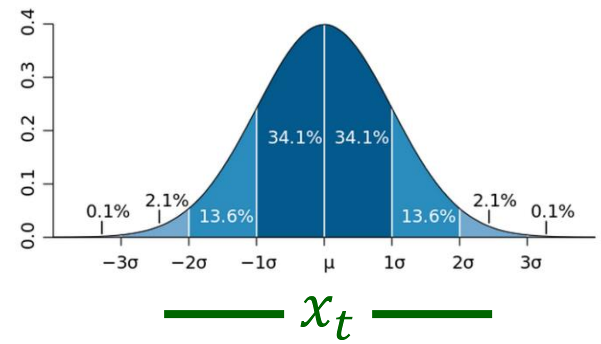
What is $\{x_t\}$

- How to update the Gaussian parameters
- Note: we first assume grey-level images

Match x_t with G_i

- Define, x_t match G_i by:

$$M(x_t, Gi) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$



- Assume G_i is a background model of q, and $x_t = I(q, t)$ then we consider x_t to be a background pixel if $M(x_t, G_i) = 1$

Using the set of K Gaussians

- Given the set B of dominant Gaussians:
  - x_t is foreground: $M(x_t, G_i) = 0\ , \forall G_i \in B$
  - x_t is background: otherwise

Weight of G_i

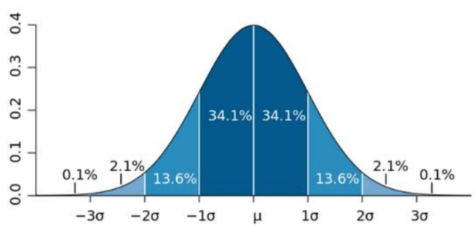
- Let w_{it} be the weight of G_i with σ_{it}
- Order the set of K Gaussians by : w_{it}/σ_{it}
  - high: more evidence & low variance
- Use it to define the set of dominant Gaussians

  (details in the paper)

Update μ_i & σ_i

$$\sigma^2 = E[(x_t - \mu)^2]$$

- Matched: $M(x_t, G_i) = 1$
  - $\mu_{i,t} = (1-\rho)\mu_{i,t-1} + \rho x_t$
  - $\sigma_{i,t}^2 = (1\text{-}\rho)\sigma_{i,t-1}^2 + \rho\left(x_t - \mu_{i,t}\right)^2$
  - ρ is the learning rate defined by a parameter α : $\rho = \alpha G(x_t \mid \mu_i, \sigma_i)$
- Unmatched: remains the same



$$M(x_t, G_i) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$

הבינתחומי הרצליה
IDC HERZLIYA

Update Weights

- Update weights of all Gaussians:
  - $w_{i,t} = (1-\alpha)w_{j,t-1} + \alpha\left(M\left(x_t, G_{j,t-1}\right)\right)$
  - α is a learning rate parameter
- Renormalize the weights $\sum_j w_{j,t} = 1$

$$M(x_t, Gi) = \begin{cases} 1 & |x_t - \mu_{i,t}| < 2.5\sigma_{i,t} \\ 0 & |x_t - \mu_{i,t}| \geq 2.5\sigma_{i,t} \end{cases}$$

When does $w_{j,t}$ increase ?

הבינתחומי הרצליה
IDC HERZLIYA

Update the Set G_i

- Given x_t such that $\forall i,\ \ M(x_t, G_i) = 0$
- Replaced G_i with smallest w_{it}/σ_{it} with a new Gaussian:
  - $\mu_{j,t} = x_t$
  - Set σ_j high

Why?

Summary

- Optical Flow:
  - Assumptions
  - Pairs of images
  - Multi scale
- Change Detection
  - Learn and model the background
  - Compare a frame to the BG
  - Mixture of Gaussians