Computer Vision
for HCI

Motion

1

Motion

- Changing scene may be observed in a sequence of images
- Changing pixels in image sequence provide important features for object detection and activity recognition

2

2

General Cases of Motion

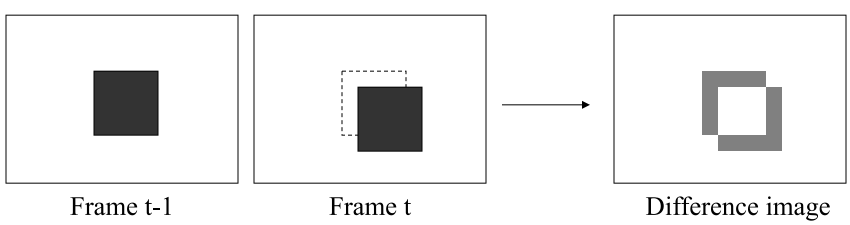
- Still camera, single moving object, constant background
  - Simplest case, motion sensors for security
- Still camera, several moving objects, constant background
  - Tracking, multi-person event analysis
- Moving camera, relatively constant scene
  - Egomotion, panning to provide wider panoramic view
- Moving camera, several moving objects
  - Most difficult, robot navigating through heavy traffic

3

3

Image Differencing

- Simplest means of motion detection
- Detect absolute value of difference between frame t-1 and t
  - Threshold result
- Only motion "presence", no direction



4

4

Single, Constant Threshold?

- Image differencing:

$$\Delta I = \begin{cases} 1 & |I_t - I_{t-1}| \geq \tau \\ 0 & else \end{cases}$$

  - Appropriate threshold perhaps not a constant value
    - Set too low and noise will appear in the difference image
    - Set too high and will not work in all situations

For **human perception**:
The greater the luminance, the more luminance change that is needed to “perceive a change” (Weber’s Law)

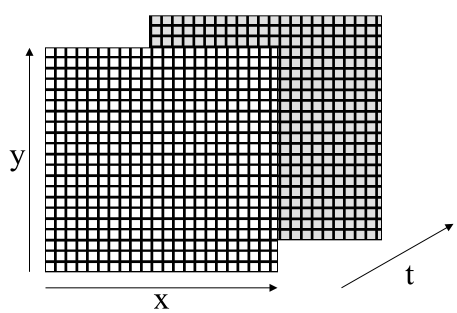
5

Optic Flow

- Image sequence $f(x, y, t)$
- Assume with small motion change (of patch), no change in gray levels

Brightness constancy constraint:

$$f(x, y, t) = f(x + dx, y + dy, t + dt)$$



6

6

Optic Flow

$$f(x,y,t) = f(x+dx, y+dy, t+dt)$$

Taylor series expansion (r.h.s.)

$$f(x,y,t) = f(x,y,t) + \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial t}dt$$

$$\downarrow$$

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

Divide by *dt*

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

7

7

Optic Flow

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

Gradients can be computed from images

(keep proper gradient **<u>normalization factor</u>**!!!)

$$\frac{\partial f}{\partial x} = f_x \Rightarrow \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} / 8 \qquad \frac{\partial f}{\partial y} = f_y \Rightarrow \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} / 8$$

Compute in I_t

$$\frac{\partial f}{\partial t} = f_t \Rightarrow F(I_t - I_{t-1})$$

e.g., *F*( ) could use 3x3 smoothing filter applied to each image before taking the difference at center pixel location

8

8

Optic Flow

******For motion <u>from</u> I_{t-1} <u>to</u> I_t**

Image (I_t)

0	**0**	**0**	**0**	**1**
0	**0**	**0**	**1**	**1**
0	**0**	**1**	**1**	**1**
0	**1**	**1**	**1**	**1**
1	**1**	**1**	**1**	**1**

$$f_x = [0(-1) + 0(0) + 1(1) + 0(-2) + 1(0) + 1(2) + 1(-1) + 1(0) + 1(1)]/8 = 3/8$$

$$f_y = [0(-1) + 0(-2) + 1(-1) + 0(0) + 1(0) + 1(0) + 1(1) + 1(2) + 1(1)]/8 = 3/8$$

$$f_t = F(I_t - I_{t-1})$$

or

$$f_t = F(smooth(I_t) - smooth(I_{t-1}))$$

9

9

Optic Flow

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

or

$$f_x u + f_y v + f_t = 0$$

Goal is to compute image motion

But only <u>one</u> equation and <u>two</u> unknowns!?!

10

10

Aggregate Optic Flow

One solution: Over-constrain by checking over small "patch" of pixels

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

$$E = \sum_i (f_{xi}u + f_{yi}v + f_{ti})^2$$

$$\frac{\partial E}{\partial u} = \sum_i (f_{xi}^{\ 2}u + f_{xi}f_{yi}v + f_{xi}f_{ti}) = 0$$

$$\frac{\partial E}{\partial v} = \sum_i (f_{yi}f_{xi}u + f_{yi}^{\ 2}v + f_{yi}f_{ti}) = 0$$

$$\begin{bmatrix} \sum_i f_{xi}^{\ 2} & \sum_i f_{xi}f_{yi} \\ \sum_i f_{yi}f_{xi} & \sum_i f_{yi}^{\ 2} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} \sum_i f_{xi}f_{ti} \\ \sum_i f_{yi}f_{ti} \end{bmatrix}$$

11

Perform least squares to solve

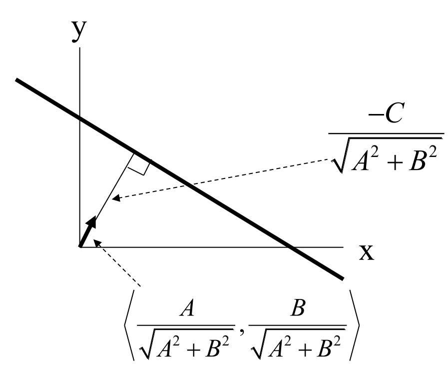
11

"Can I do anything with this equation itself?"

$$f_x u + f_y v + f_t = 0$$

Consider "Normal Form" of Straight Line Equation:

$$\frac{A}{\sqrt{A^2+B^2}}x + \frac{B}{\sqrt{A^2+B^2}}y + \frac{C}{\sqrt{A^2+B^2}} = 0$$



12

12

“**Normal** Optic Flow”

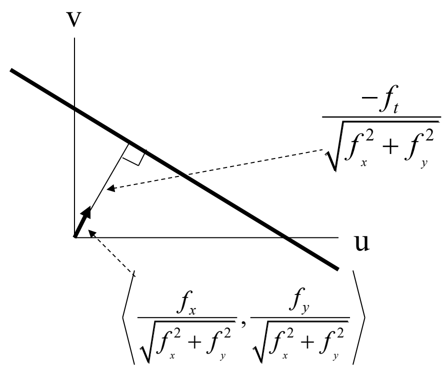
$$f_x u + f_y v + f_t = 0$$

$$\frac{f_x}{\sqrt{f_x^2 + f_y^2}} u + \frac{f_y}{\sqrt{f_x^2 + f_y^2}} v + \frac{f_t}{\sqrt{f_x^2 + f_y^2}} = 0$$

Can determine only one component of flow:

<u>Magnitude</u> of flow in the <u>direction of the brightness gradient</u> (perpendicular to brightness contour)

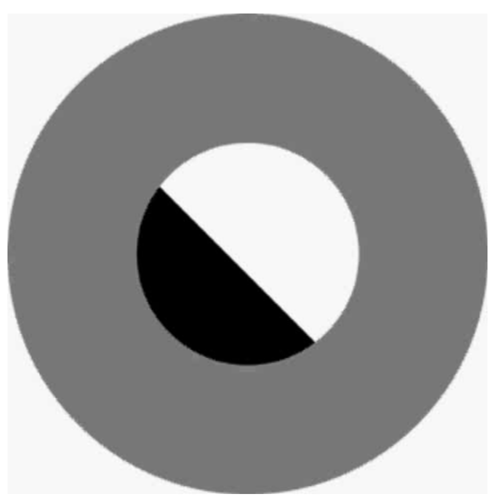
“Normal Flow”



13

13

Aperture Problem



14

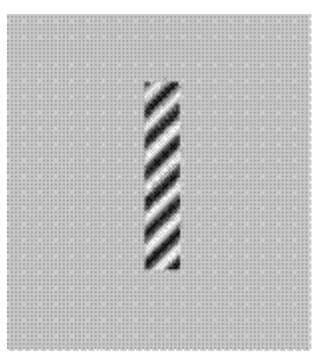
14

Aperture Problem

15

15

Barber Pole Illusion



16

16

Weighted Aggregate Normal Flow

- Weight the computation of the normal flow magnitude by the strength of gradients in local region (smoothes result)

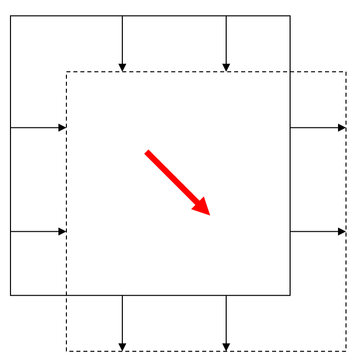
$$NF_{mag} = \frac{-f_t}{\sqrt{f_x^2 + f_y^2}}$$

$$wNF_{mag} = \frac{-\sum_i \left(\sqrt{f_{xi}^2 + f_{yi}^2}\right) \cdot f_{ti}}{\sum_i \left(\sqrt{f_{xi}^2 + f_{yi}^2}\right) \cdot \sqrt{f_{xi}^2 + f_{yi}^2}} = \frac{-\sum_i \left(\sqrt{f_{xi}^2 + f_{yi}^2}\right) \cdot f_{ti}}{\sum_i \left(\sqrt{f_{xi}^2 + f_{yi}^2}\right)^2}$$

17

17

Normal Flow vs. Desired Optic Flow for Moving Box



18

18

Motion

Hierarchical Motion Estimation

19

Optical Flow

- Assumptions for computation of optical flow $f_x u + f_y v + f_t = 0$
  - Small motion change (of patch)
  - No change in gray levels after moving (brightness constancy constraint)
- Not entirely realistic for real-world motions in video
  - Usually larger motion (even at 30+ frames/sec)

20

20

Hierarchical Motion Estimation

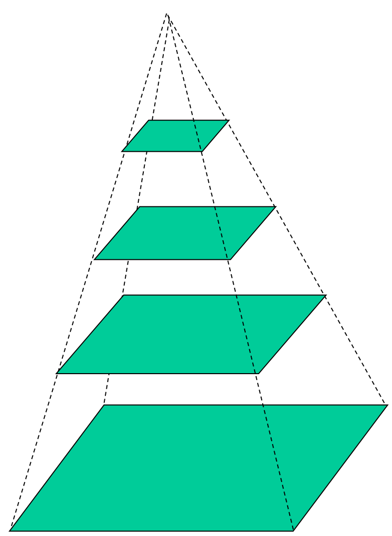
- Bergen J., and Hingorani R. "*Hierarchical Motion-Based Frame Rate Conversion*", 1990
- Bergen J., Anandan P., Hanna K., and Hingorani R. "*Hierarchical Model-Based Motion Estimation*", 1992
- Key feature of framework
  - Coarse-to-fine refinement strategy
    - Local model (used in estimation process)
    - Global model (constrains overall structure of motion)

21

21

Hierarchical Motion Estimation

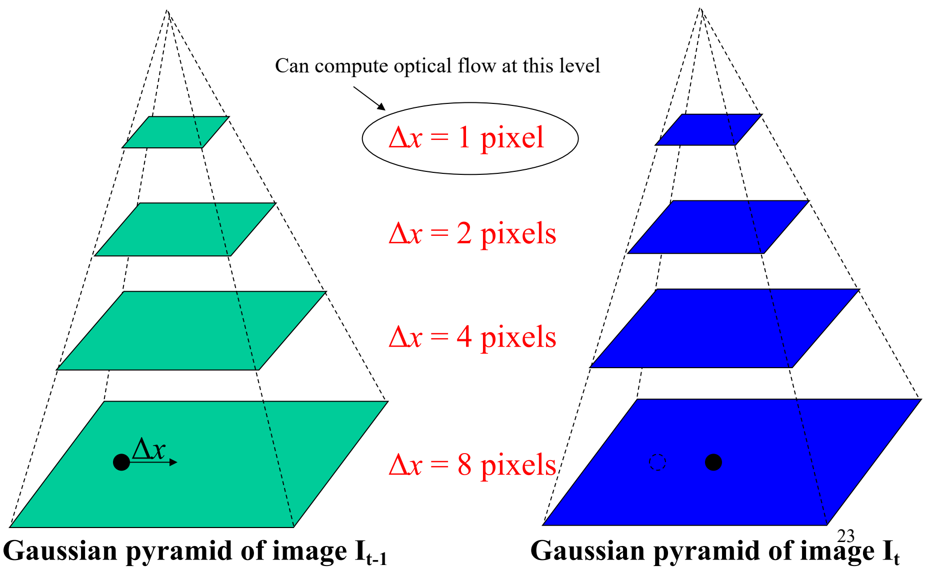
- Key element is use of <u>multi-resolution</u> image representation to allow **optical flow** constraint equation **even when motion is <u>fairly large</u>**



22

22

Displacement Reduction in Pyramid



23

Smoothness

- Pyramid construction smoothens out discontinuities to provide better gradient-based estimations
  - Smooths over boundaries, but dealt with in later estimations

24

24

Displacement Reduction in Pyramid

- Can compute optical flow at top level of pyramid, but …
  - Low image resolution (reduced size), so not true answer for original sized image
  - Motion may even start at lower (larger) levels
- This only provides an initial estimate (to be refined)

25

25

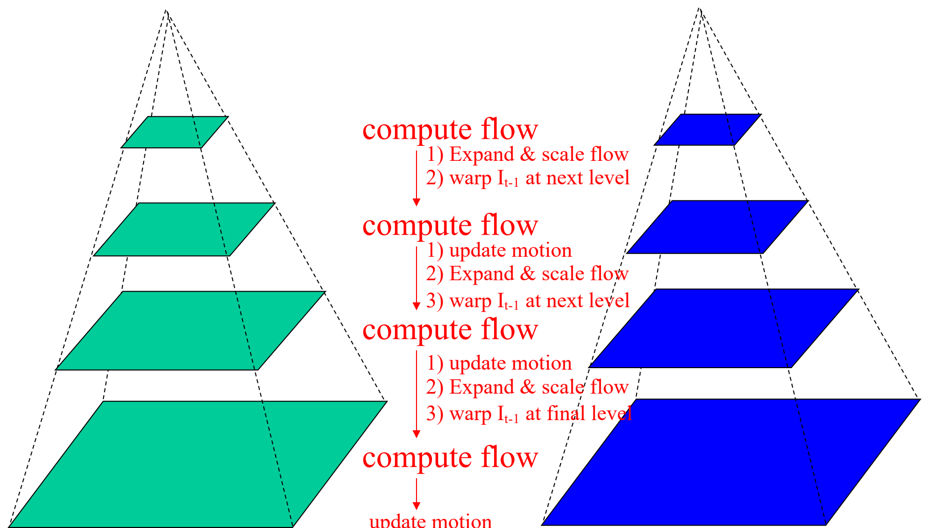
Hierarchical Motion Estimation

- Overview of approach for pair of images (I_{t-1}, I_t)
  - Generate multi-resolution Gaussian pyramid of images
    - Iteratively reduce image
  - Perform local estimation of displacements
    - Compute optical flow
    - Start at smallest pyramid level
  - Expand flow and warp image (by the flow) at next level below, then compute local flow estimation and update
    - Compensates for previously estimated displacements
  - Iteratively refine optical flow

26

26

Course-to-Fine Motion Estimation



Gaussian pyramid of image I_{t-1} **Gaussian pyramid of image I_t**

27

27

Computing Optic Flow

Aggregate flow method

$$\frac{dx}{dt} = u \qquad E = \sum_i (f_{xi}u + f_{yi}v + f_{ti})^2$$

$$\frac{dy}{dt} = v \qquad \frac{\partial E}{\partial u} = \sum_i (f_{xi}{}^2 u + f_{xi} f_{yi} v + f_{xi} f_{ti}) = 0$$

$$\frac{\partial E}{\partial v} = \sum_i (f_{yi} f_{xi} u + f_{yi}{}^2 v + f_{yi} f_{ti}) = 0$$

$$\begin{bmatrix} \sum_i f_{xi}{}^2 & \sum_i f_{xi} f_{yi} \\ \sum_i f_{yi} f_{xi} & \sum_i f_{yi}{}^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum_i f_{xi} f_{ti} \\ \sum_i f_{yi} f_{ti} \end{bmatrix}$$

28

Least squares

28

Expanding & Scaling Motion

- The flow (u, v) is expanded to match the size of the next (bigger) pyramid image
  - Using 2x spatial expand operations (see upsampling in pyramid lecture)
  - Also **must expand the flow <u>magnitude</u>**, as the vectors are in pixel units (and image was just <u>doubled</u> in size)

u' = expand(u) * **2**

v' = expand(v) * **2**

29

29

Image Warping

- Given image I_{t-1} at next (larger) pyramid level below and the corresponding (u',v') motion estimate expanded down for that level, use motion vectors to warp I_{t-1} into W
- W and the corresponding level I_t should appear to "be close"
  - Enough to compute valid optical flow

30

30

Matlab

```
function [warpIm]=warp(Im, u, v)

[M, N]=size(Im);

[x, y]=meshgrid(1:N,1:M);

warpIm=interp2(x, y, Im, x-u, y-v);

% Matlab returns a NaN in places outside the first image
I=find(isnan(warpIm));
warpIm(I)=zeros(size(I));
```

31

31

Updating Motion

- Compute new, local motion flow at this level
  - From W to I_t
- **Refine** previous flow estimate (that was expanded/scaled and used from previous level for W)

$$u = u' + \text{local\_}u$$
$$v = v' + \text{local\_}v$$

Updated motion estimate at this level

Motion using warped image W and I_t at this level

Expanded motion from previous level (used for warping this level into W)

32

32

Other 2-D Motion Approaches

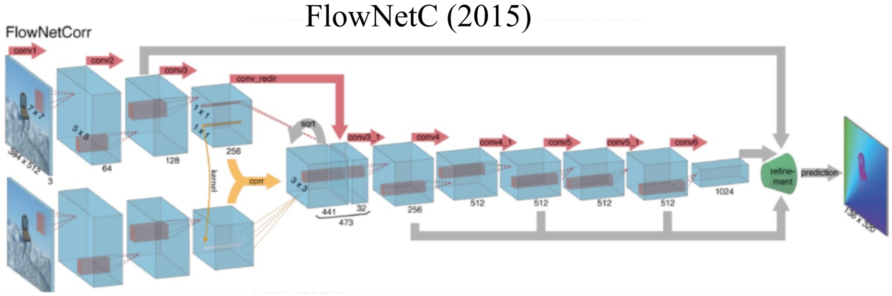
- Many, many, many other approaches
  - Add smoothness terms to error function
  - Robust statistics
  - …

33

33

Optical Flow with Deep Learning

FlowNetC (2015)



Uses two separate, yet identical, processing streams to produce meaningful representations for the two images and then combines them at a later stage (at a higher level). Incorporates a "correlation layer" that performs multiplicative patch comparisons between two feature maps. This roughly resembles the standard matching approach when one first extracts features from patches of both images and then compares those feature vectors. Lastly applies feature map expansion, and concatenates its with corresponding feature maps from the "contractive" part of the network.

34

34

Motion

Motion History Images

35

Motion Templates

- From blurred video, easily recognize the activity
- Recognize holistic “patterns of motion”
  - No tracking of structural features (hands, elbows)

[video example next slide…]

36

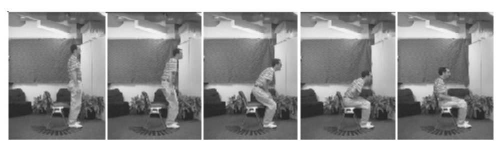
36

What is it????

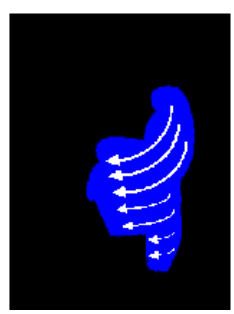
37

37

Representation Theory



- Decompose motion
- The “where”
  - Spatial pattern of “where” motion occurred
  - Motion energy image (MEI)
- The “how”
  - Progression of “how” the motion is moving
  - Motion history image (MHI)



38

38

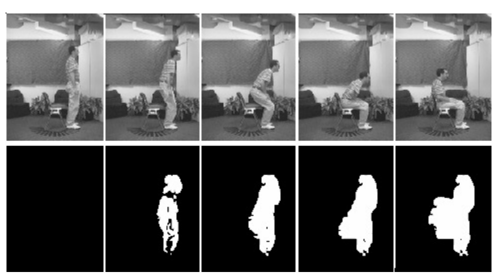
Temporal Template

- MHI (and MEI) is a static image
  - Value at each pixel is some function of the motion at that pixel
  - MHI: pixel records temporal history
  - MEI: pixel records presence of motion
- TT = [MEI, MHI]

39

39

Cumulative Motion Images



- Cumulative motion presence
  - Image differencing
- Sweeps out particular region
- Shape can be used to suggest action and view

40

40

Across View Angle

-90° -70° -50° -40° -20° 0°

41

41

Motion Energy Image (MEI)

- Cumulative motion images

$$E_\tau(x,y,t) = \bigcup_{i=0}^{\tau-1} D(x,y,t-i)$$

- Duration τ defines temporal extent

42

42

Motion History Image (MHI)

- Represent “how” motion is moving
- Pixel intensity is function of temporal history at that point
- Simple replace-and-decay operator with timestamp $\boldsymbol{\tau}$

$$MHI_{\delta}(x,y) = \begin{cases} \tau & \text{if } \Psi(I(x,y)) \neq 0 \\ 0 & \text{else if } MHI_{\delta}(x,y) < \tau - \delta \end{cases}$$

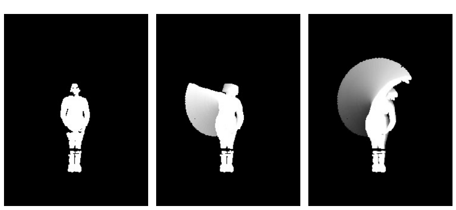
Note: later will normalize MHI to values (0-1) for matching

43

43

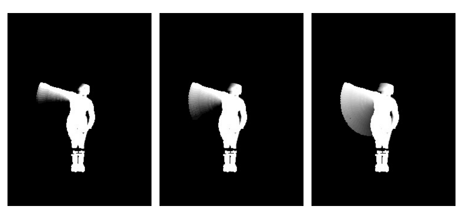
Silhouette Differences

FRAME-1 FRAME-35 FRAME-70



MHI-1 MHI-35 MHI-70

Changing delay $\boldsymbol{\delta}$

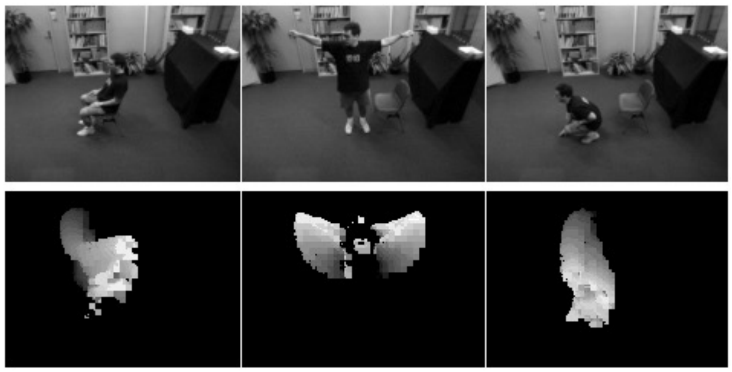


$\boldsymbol{\delta}$ = 0.25 $\boldsymbol{\delta}$ = 0.5 $\boldsymbol{\delta}$ = 1

44

44

vs. Image Differencing



45

45

Matching Templates

- First normalize template to (0-1)
  - See next slide!
- Invariant to viewing condition?
  - Want scale and translation invariance
- Feature vector of 7 Similitude moments
  - Vector for MEI
  - Vector for MHI
- Mahalanobis distance (squared) metric to models $MD = (x-m)^T K^{-1} (x-m)$

46

46

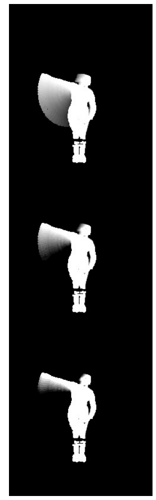
MHI (Re)Normalization

Want "fade to black" for time duration used

For a pixel having motion that
occurred at any time t:

$$\max\left(0, \frac{t - [\min(t) - \Delta t]}{\max(t) - [\min(t) - \Delta t]}\right)$$

$$\max\left(0, \frac{\overbrace{frame\#}^{\text{frame \# at pixel}} - [\min(frame\#) - 1]}{\underbrace{\max(frame\#)}_{\substack{\text{The most current frame \#} \\ \text{(most recent)}}} - [\underbrace{\min(frame\#)}_{\substack{\text{The oldest frame \# that still} \\ \text{want to keep/display}}} - 1]}\right)$$



Duration

Long

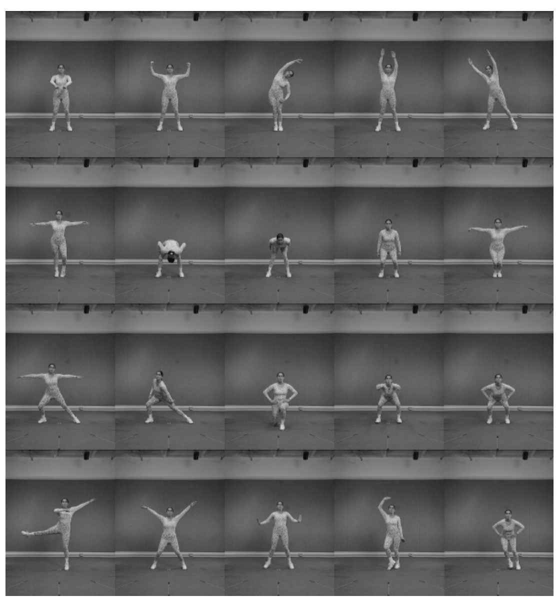
Medium

Short

47

47

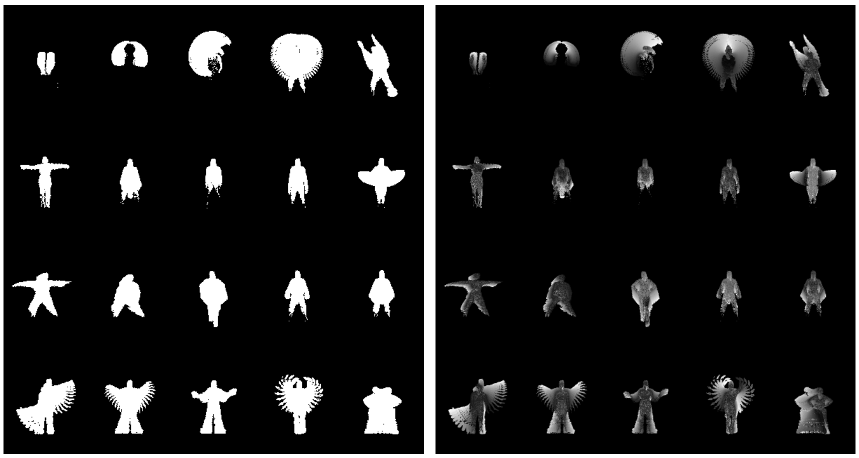
Aerobics Data



48

48

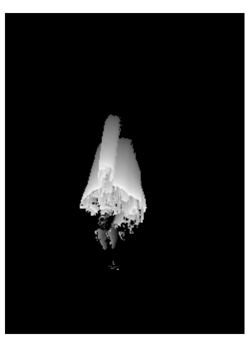
MEIs and MHIs



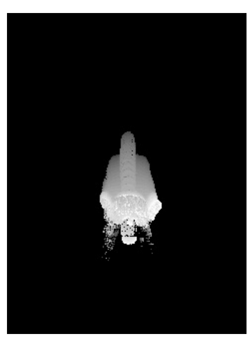
49

49

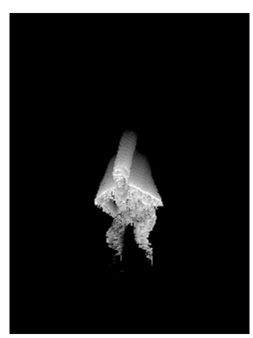
Confusion Difficulties



Test input:
Move 13 at 30^{o}



Closest match:
Move 6 at 0^{o}



Correct match

50

50

Subject Variances and Motion Calculation

Test input: Move 16 at 30°

Closest match: Move 15 at 0°

Correct match

Bad motion calculation in test input, and different subject performances

51

51

Temporal Segmentation

- Maximum and minimum duration of actions

  $\tau_{\min} - \tau_{\max}$
- Backward searching time window
  - Compute $MHI_{\tau_{\max}}(x, y, t)$
  - Normalize (0-1), match to models
  - Compute $MHI_{\tau - \Delta\tau}(x, y, t)$
    - Just threshold previous un-normalized version! (removes older parts)
  - Normalize (0-1), Match to models
  - Keep going until $\tau - \Delta\tau = \tau_{\min}$

52

52

Motion Gradients

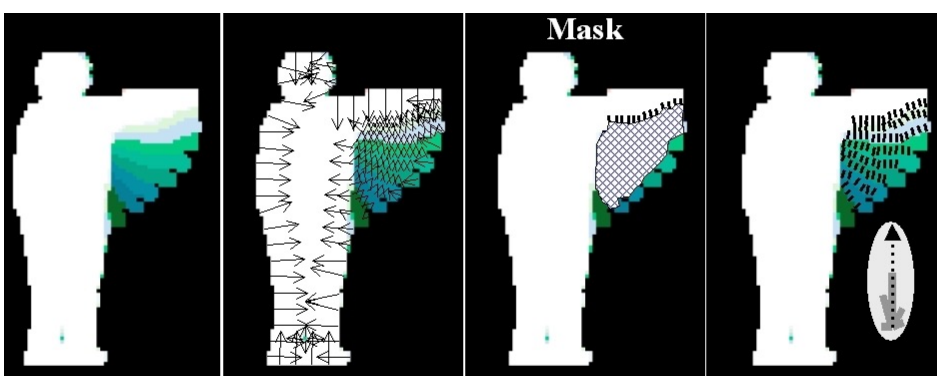
- Can perceive “direction of motion” in intensity fading of motion template
  - Downward, backward swipe
- Convolve gradient masks with template
  - Intensity gradient similar to motion flow (normal flow)

$$F_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad F_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \end{bmatrix}$$

53

53

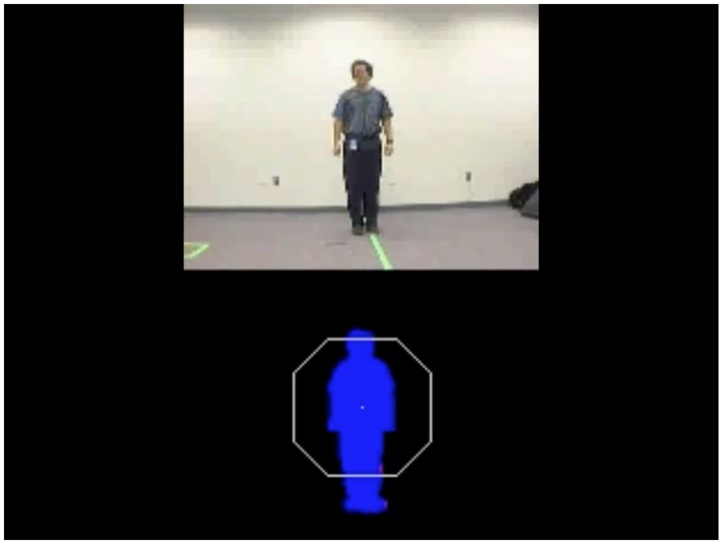
Motion Gradients



54

54

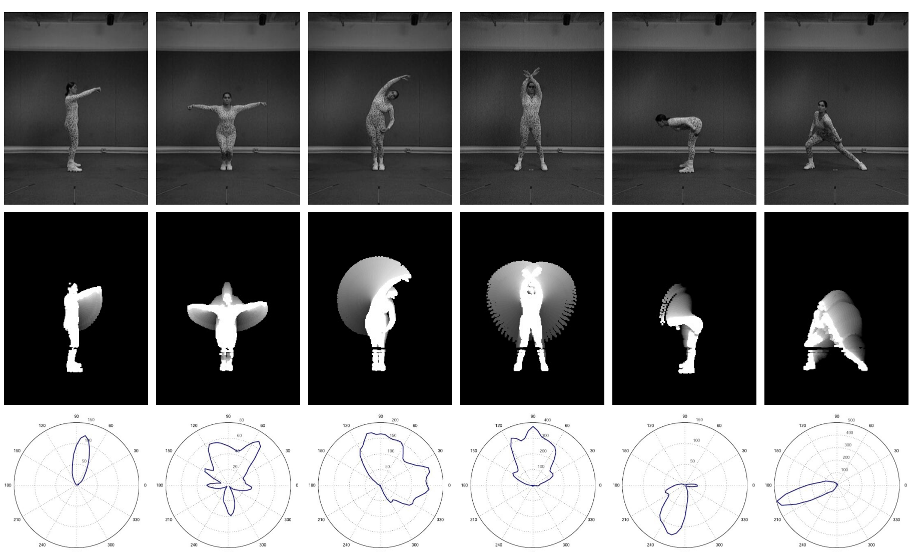
Demonstration



55

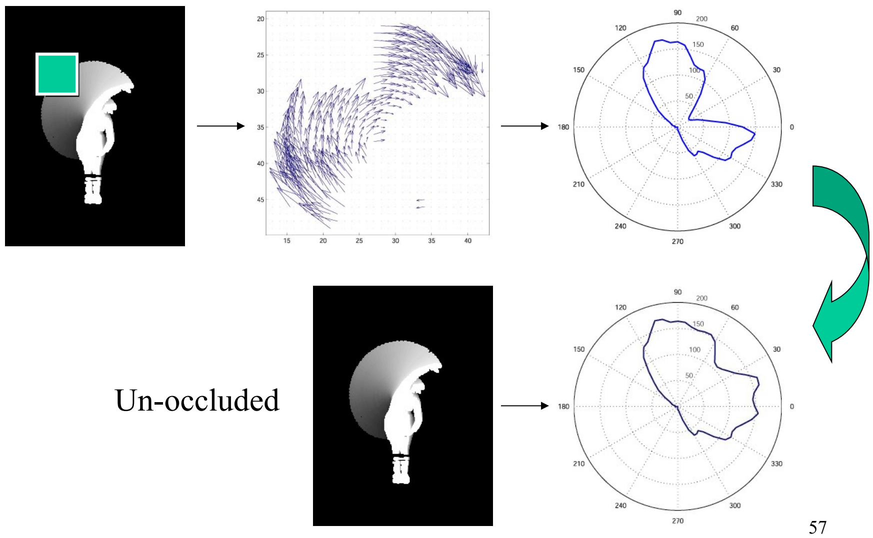
55

Motion Orientation Histograms



56

Effect of Occlusion



57

57

Summary

- Changing pixels in image sequence provide important features for motion analysis
- Optic flow
  - Brightness constancy constraint
  - Under-constrained and aperture problem
  - Aggregate optic flow
- Normal flow
  - Magnitude of flow in the direction of the brightness gradient (perpendicular to brightness contour)

58

58

Summary (con’ t)

- Multi-resolution motion estimation
  - Coarse-to-fine refinement strategy
- Permits use of optical flow constraint equation even when motion is fairly large
- Key steps
  - Generate multi-resolution Gaussian pyramid of images
  - Perform local estimation of motion displacements
  - Expand flow and warp image at next level, then compute local flow estimation
  - Update global motion estimation

59

59

Summary (con’ t)

- Motion Templates
  - MEI and MHI
  - Temporal accumulator and decay operator
- Recognition
  - Compute and match first seven scale- and translation-invariant moments from entire MHI (and MEI)
    - Also from a binary MHI
- Motion Gradients
  - Direction of motion from intensity gradient

60

60