EECS332 Digital Image Analysis

Motion Estimation

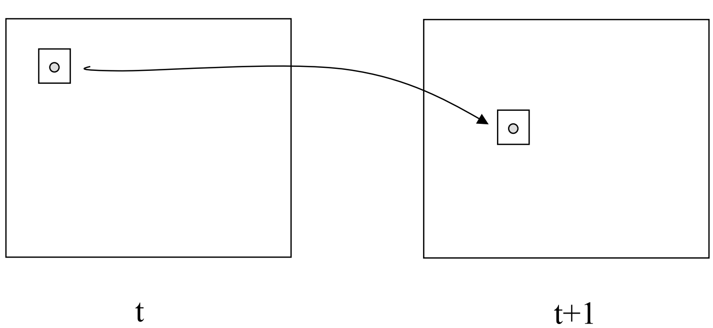
Ying Wu

Electrical Engineering & Computer Science
Northwestern University
Evanston, IL 60208

http://www.ece.northwestern.edu/~yingwu
yingwu@ece.northwestern.edu

Motivations

- If something is moving in video, can you keep tracking its movements?
- The problem:



2

Outline

- Motivation
- Basic questions
- Exhaustive search
- Flow-constraint equation
- Gradient-based search

Basic questions

- Matching $\leftarrow$ what are the criteria for matching?
- Searching $\leftarrow$ how to find the best match?

Matching Criterion

- SSD (sum of squared difference)

$$D = \sum_x \sum_y [I(x,y) - T(x,y)]^2$$

- Cross-correlation

$$C = \sum_x \sum_y I(x,y)T(x,y)$$

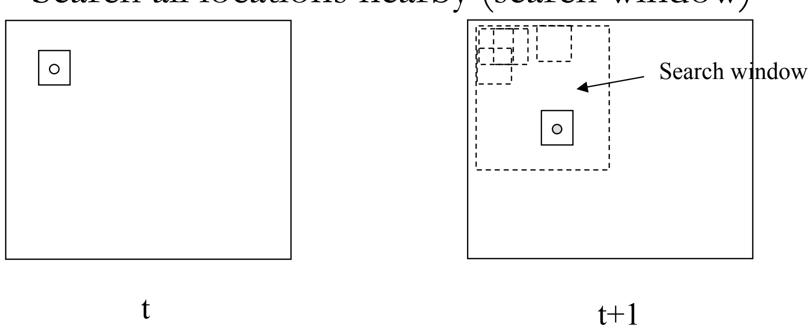
- Normalized cross-correlation

$$N = \frac{\sum_x \sum_y [I(x,y) - \bar{I}][T(x,y) - \bar{T}]}{\sqrt{[\sum_x \sum_y (I(x,y) - \bar{I})^2][\sum_x \sum_y (T(x,y) - \bar{T})^2]}}$$

5

Exhaustive Search

- Search all locations nearby (search window)



Pros:

✓ easy to implement

Cons:

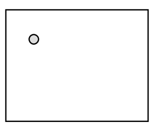
✓ computationally intensive

✓ can not handle rotation

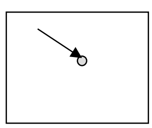
6

Gradient-based search

- Let I(x,y, τ) → current image
- Let I(x,y,0) → reference image or template
- For now, we assume a pure translational motion
- Constant Brightness Constraint:
  - i.e., I(x,y,0) = I(x+u, y+v, τ) $\forall (x,y) \in R$
  - where (u,v) is the displacement



I(x,y,0)



I(x+u,y+v, τ)

$$\boxed{\begin{aligned}(u^*, v^*) &= \arg\min_{(u,v)} D(u,v) \\ &= \arg\min_{(u,v)} \sum_x \sum_y [I(x+u, y+v, \tau) - I(x,y,0)]^2\end{aligned}}$$

7

Flow Constraint Equation

- We perform Taylor expansion of I(x+u,y+v,t) with respect to (x,y,0):

$$I(x+u, y+v, \tau) = I(x,y,0) + \frac{\partial I(x,y,0)}{\partial x}u + \frac{\partial I(x,y,0)}{\partial y}v + \frac{\partial I(x,y,0)}{\partial t}\tau + O(t^2)$$

- denote

$$\frac{\partial I(x,y,0)}{\partial x} = I_x, \quad \frac{\partial I(x,y,0)}{\partial y} = I_y, \quad \frac{\partial I(x,y,0)}{\partial t} = I_t$$

- Since I(x+u,y+v, τ)= I(x,y,0)

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

8

Solution

$$\Rightarrow \quad D(u,v) = \sum_x \sum_y (I_x u + I_y v + I_t \tau)^2$$

We have
$$\nabla D(u,v) = \begin{bmatrix} \sum_x \sum_y (I_x u + I_y v + I_t \tau) I_x \\ \sum_x \sum_y (I_x u + I_y v + I_t \tau) I_y \end{bmatrix} = 0$$

Easy to see
$$\sum_x \sum_y \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\tau \sum_x \sum_y \begin{bmatrix} I_x I_t \\ I_y I_t \end{bmatrix}$$

$$\Rightarrow \quad \begin{bmatrix} u \\ v \end{bmatrix} = -\tau \left(\sum_x \sum_y \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right)^{-1} \left(\sum_x \sum_y \begin{bmatrix} I_x I_t \\ I_y I_t \end{bmatrix} \right)$$

9

Isn’t it nice?

$$\begin{bmatrix} u \\ v \end{bmatrix} = -\tau \left(\sum_x \sum_y \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right)^{-1} \left(\sum_x \sum_y \begin{bmatrix} I_x I_t \\ I_y I_t \end{bmatrix} \right)$$

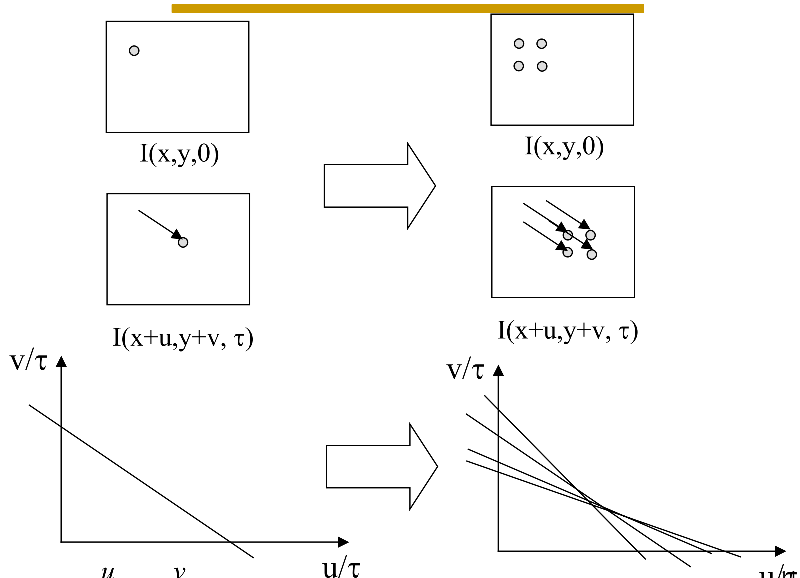
obtained from the template alone

I_t is image difference

- This is a closed form solution.
- One important thing → τ
  - We can not determine τ!
  - Thus, we can only solve u/τ and v/τ → velocities
  - i.e., this only provides a direction to search

10

Another explanation



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

Doesn't it look familiar?

A familiar solution

$$\begin{cases} I_{x1}u + I_{y1}v + I_{t1}\tau = 0 \\ I_{x2}u + I_{y2}v + I_{t2}\tau = 0 \\ \vdots \\ I_{xN}u + I_{yN}v + I_{tN}\tau = 0 \end{cases} \Rightarrow \begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \vdots & \vdots \\ I_{xN} & I_{yN} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\tau \begin{bmatrix} I_{t1} \\ I_{t2} \\ \vdots \\ I_{tN} \end{bmatrix}$$

Sounds familiar?

Ax=b

You can easily figure out the solution now.

12

Handling rotation?

- Assume a pure rotation

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

objective function

$$D(\theta) = \sum_x \sum_y \left[I(R(\theta)\begin{bmatrix} x \\ y \end{bmatrix}, \tau) - I(x, y, 0) \right]^2$$

Taylor expansion

$$I(R(\theta)\begin{bmatrix} x \\ y \end{bmatrix}, \tau) = I(x, y, 0) + \frac{\partial I}{\partial \theta}\theta + \frac{\partial I}{\partial t}\tau + o(t^2)$$

where $\frac{\partial I}{\partial \theta} = -\frac{\partial I}{\partial x} y + \frac{\partial I}{\partial y} x = I_\theta$

derivative

$$D(\theta) = \sum_x \sum_y (I_\theta \theta + I_t \tau)^2 \quad \Rightarrow \quad \nabla D(\theta) = \sum_x \sum_y (I_\theta \theta + I_t \tau) I_\theta$$

solution

$$\Rightarrow \quad \theta = -\tau \frac{\sum_x \sum_y I_\theta I_t}{\sum_x \sum_y I_\theta^2}$$

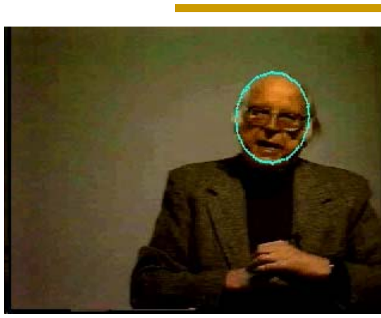
13

Face/Head Tracking

Ying Wu
yingwu@ece.northwestern.edu

Electrical and Computer Engineering
Northwestern University, Evanston, IL
http://www.ece.northwestern.edu/~yingwu

Tracking Heads?



Courtesy of Y. Wu, 2001

- The task:
  Localize faces and track them in image sequences
- Challenges:
  Lighting, occlusion, rotation, etc.

Outline

- ✓ Motivation
- ✓ What is tracking?
- ✓ One solution (Birchfield_CVPR98)
- ✓ Other methods and open issues

Motivation

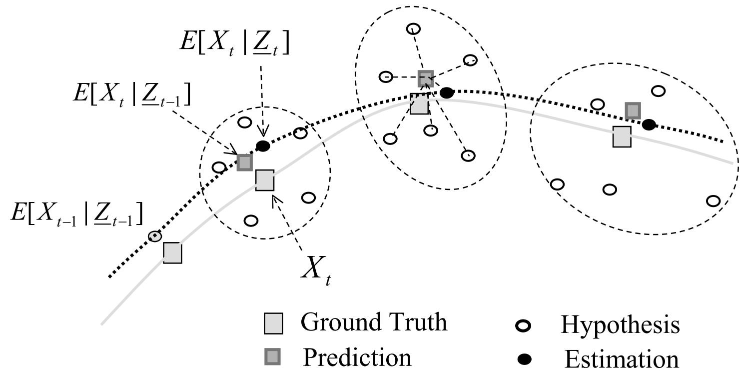
- Why tracking?
  - The complexity of face detection
    - ✓ scan all the pixel positions and several scales
  - The limitation of face detection
    - ✓ hard to handle out-of-plane rotation
  - Can we maintain the identity of the faces?
    - ✓ although face recognition is the ultimate solution for this, we may not need it, if not necessary
- Objectives
  - fast (frame-rate) face/head localization
  - handle 360^o out-of-plane rotation

Visual Tracking

Four Elements

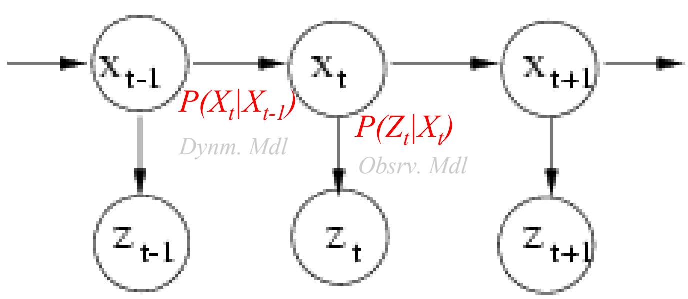
- Infer target states in video sequences
- Target states vs. image observations
- Visual cues and modalities
- Four elements
  - *Target representation* X
  - *Observation representation* Z
  - *Hypotheses measurement* $p(Z_t|X_t)$
  - *Hypotheses generating* $p(X_t|X_{t-1})$

Visual Tracking



$$(E[X_{t-1} \mid \underline{Z}_{t-1}, Z_t]) \Rightarrow E[X_t \mid \underline{Z}_t]$$

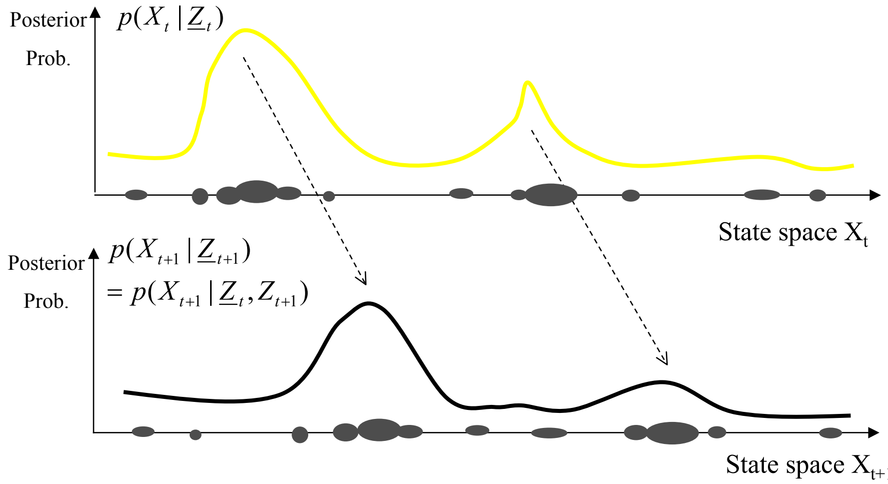
Formulating Visual Tracking



$$p(X_{t+1} \mid \underline{Z}_{t+1}) \propto p(Z_{t+1} \mid X_{t+1}) p(X_{t+1} \mid \underline{Z}_t)$$

$$p(X_{t+1} \mid \underline{Z}_t) = \int p(X_{t+1} \mid X_t) p(X_t \mid \underline{Z}_t) dX_t$$

Tracking as Density Propagation



One Solution

(Birchfield_CVPR98)

- Framework
- Search strategy
- Edge cue
- Color cue

Framework

- s = (x,y,σ)
- Tracking is treated as a local search based on the prediction

$$\mathbf{s}^* = \arg\max_{\mathbf{s}_i \in S}\{\bar{\phi}_g(\mathbf{s}_i) + \bar{\phi}_c(\mathbf{s}_i)\},$$

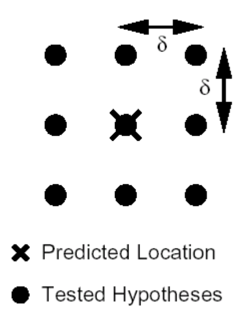
hypotheses

Edge matching score

color matching score

Search Strategy

- Local exhaustive search



δ is the search step size

- Do you have better ideas?

Edge Cue

- Method I

$$\phi_g(\mathbf{s}) = \frac{1}{N_\sigma} \sum_{i=1}^{N_\sigma} |\mathbf{g_s}(i)|,$$

The the magnitude of the gradient at perimeter pixel i of the ellipse s.

# of pixels on the perimeter of the ellipse

- Method II

$$\phi_g(\mathbf{s}) = \frac{1}{N_\sigma} \sum_{i=1}^{N_\sigma} |\mathbf{n}_\sigma(i) \cdot \mathbf{g_s}(i)|,$$

unit vector normal to the ellipse at pixel i.

- Which is better?

Normalization

$$\bar{\phi}_g(\mathbf{s}) = \frac{\phi_g(\mathbf{s}) - \min_{\mathbf{s}_i \in S} \phi_g(\mathbf{s}_i)}{\max_{\mathbf{s}_i \in S} \phi_g(\mathbf{s}_i) - \min_{\mathbf{s}_i \in S} \phi_g(\mathbf{s}_i)}.$$

- Why do we need normalization?
- How good is it?

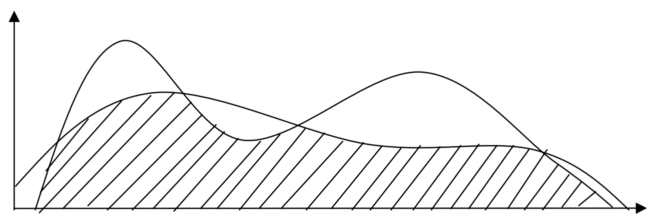
Color Cue

- Histogram intersection

$$\phi_c(\mathbf{s}) = \frac{\sum_{i=1}^{N} \min(I_{\mathbf{s}}(i), M(i))}{\sum_{i=1}^{N} I_{\mathbf{s}}(i)},$$

# of bins

Model histogram



Color Cue

- Color space
  - B-G
  - G-R
  - R+G+B (why do we need that)
- 8 bins for B-G and G-R, 4 for R+G+B
- Training the model histogram
- Normalization

$$\bar{\phi}_c(\mathbf{s}) = \frac{\phi_c(\mathbf{s}) - \min_{\mathbf{s}_i \in S} \phi_c(\mathbf{s}_i)}{\max_{\mathbf{s}_i \in S} \phi_c(\mathbf{s}_i) - \min_{\mathbf{s}_i \in S} \phi_c(\mathbf{s}_i)}.$$

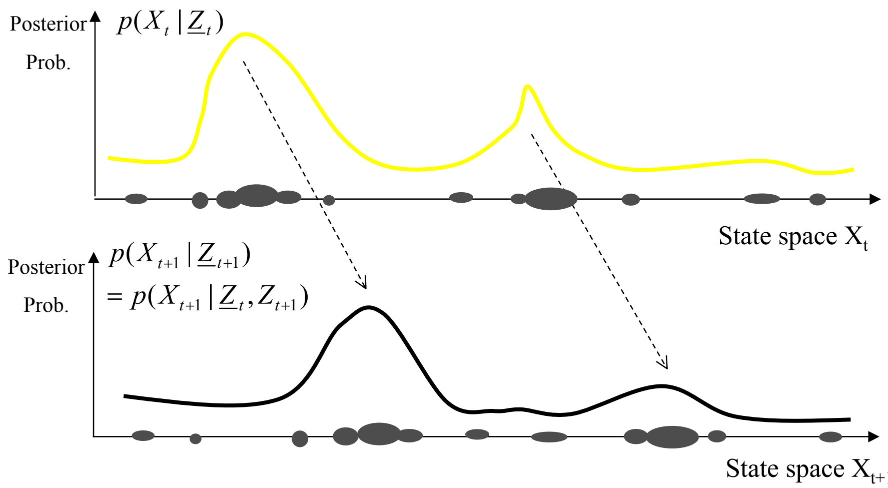
Comments

- Can the rotation be handled?
- Can the scaling issue be handled?
- Is the search strategy good enough?
- Is the color module good?
- Is the motion prediction enough?
- Is the combination of the two cues good?
- Can it handle occlusion?
- Can it cope with multiple faces
  - Coalesce
  - Switch ID

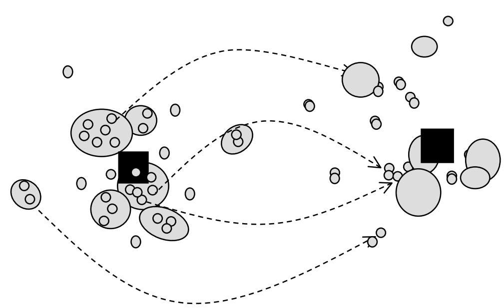
Other Solutions

- Condensation algorithm
- 3D head tracking

Tracking as Density Propagation



Sequential Monte Carlo

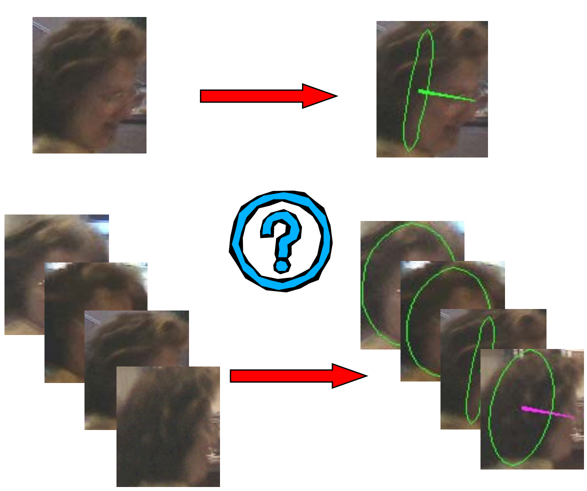


- $P(X_t|Z_t)$ is represented by a set of weighted samples
- Sample weights are determined by $P(Z_t^{(n)}|X_t^{(n)})$
- Hypotheses generating is controlled by $P(X_t|X_{t-1})$

Challenge to Condensation

- Curse of dimensionality
  - What to track?
    - ✓ *Positions, orientations*
    - ✓ *Shape deformation*
    - ✓ *Color appearance changing*
  - The dimensionality of X
  - The number of hypotheses grows exponentially

3D Face Tracking: The Problem



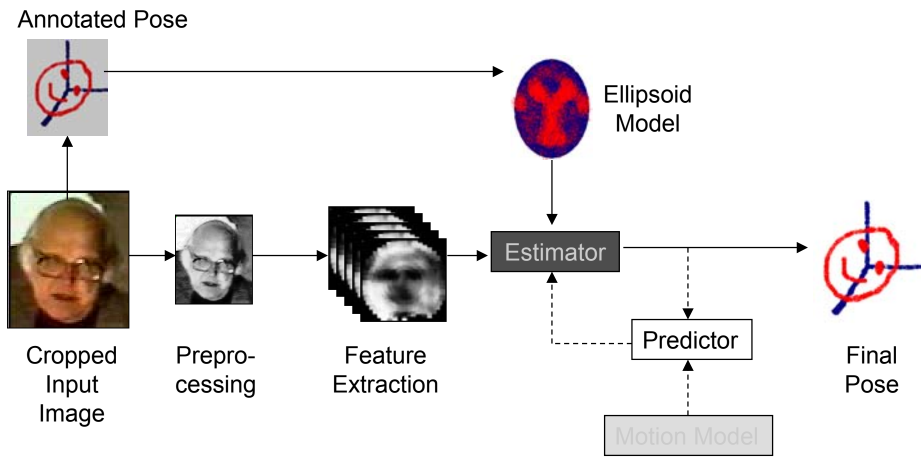
The goal:

Estimate and track 3D head poses

The challenges:

- ✓ Side view
- ✓ Back view
- ✓ Poor illumination
- ✓ Low resolution
- ✓ Different users

3D Face Tracking: A Solution



Courtesy of Y. Wu and K. Toyama, 2000

3D Face Tracking: some results