Tracking Objects with Dynamics

Computer Vision
CS 543 / ECE 549
University of Illinois

Amin Sadeghi

some slides from Lana Lazebnik, Kristen Grauman, Deva Ramanan, Derek Hoiem
Today: Tracking Objects

Goal: Locating a moving object/part across video frames

This Class:

• Examples and Applications
• Overview of probabilistic tracking
• Kalman Filter
• Particle Filter
Tracking Examples

Traffic: http://www.youtube.com/watch?v=dlZEakHcD_o

Soccer: http://www.youtube.com/watch?v=ZqQIlItFAnxg

Face: http://www.youtube.com/watch?v=i_bZNVmhJ2o

Body: http://www.youtube.com/watch?v=gRyo-DM2F6E

Eye: http://www.youtube.com/watch?v=NCTYdUEMotg

Gaze: http://www.youtube.com/watch?v=-G6Rw5cU-1c
Further applications

- Motion capture
- Augmented Reality
- Action Recognition
- Security, traffic monitoring
- Video Compression
- Video Summarization
- Medical Screening
Things that make visual tracking difficult

- Small, few visual features
- Erratic movements, Moving very quickly
- Occlusions, leaving and coming back
- Surrounding similar-looking objects
Strategies for tracking

• Tracking by repeated detection
  – Works well if object is easily detectable (e.g., face or colored glove) and there is only one
  – Need some way to link up detections
  – Best you can do, if you can’t predict motion
Tracking with dynamics

• Key idea: Based on a model of expected motion, predict where objects will occur in next frame, before even seeing the image
  – Restrict search for the object
  – Measurement noise is reduced by trajectory smoothness
  – Robustness to missing or weak observations
Strategies for tracking

• Tracking with motion prediction
  – Predict the object’s state in the next frame
  – **Kalman filtering**: next state can be linearly predicted from current state (Gaussian)
  – **Particle filtering**: sample multiple possible states of the object (non-parametric, good for clutter)
General model for tracking

- **state** $X$: The actual state of the moving object that we want to estimate
  - State could be any combination of position, pose, viewpoint, velocity, acceleration, etc.

- **observations** $Y$: Our actual measurement or observation of state $X$. Observation can be very noisy

- At each time $t$, the state changes to $X_t$ and we get a new observation $Y_t$
Steps of tracking

- **Prediction:** What is the next state of the object given past measurements?

\[ P(X_t | Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}) \]
Steps of tracking

- **Prediction**: What is the next state of the object given past measurements?

\[ P(X_t | Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}) \]

- **Correction**: Compute an updated estimate of the state from prediction and measurements

\[ P(X_t | Y_0 = y_0, \ldots, Y_{t-1} = y_{t-1}, Y_t = y_t) \]
Simplifying assumptions

- Only the immediate past matters

\[ P(X_t | X_0, \ldots, X_{t-1}) = P(X_t | X_{t-1}) \]

dynamics model
Simplifying assumptions

• Only the immediate past matters

\[ P(X_t | X_0, \ldots, X_{t-1}) = P(X_t | X_{t-1}) \]

dynamics model

• Measurements depend only on the current state

\[ P(Y_t | X_0, Y_0 \ldots, X_{t-1}, Y_{t-1}, X_t) = P(Y_t | X_t) \]

observation model
Simplifying assumptions

- Only the immediate past matters
  \[
  P(X_t|X_0,\ldots, X_{t-1}) = P(X_t|X_{t-1})
  \]
  dynamics model

- Measurements depend only on the current state
  \[
  P(Y_t|X_0, Y_0 \ldots, X_{t-1}, Y_{t-1}, X_t) = P(Y_t|X_t)
  \]
  observation model
Problem statement

• We have models for
  
  Likelihood of next state given current state: \( P(X_t|X_{t-1}) \)
  
  Likelihood of observation given the state: \( P(Y_t|X_t) \)

• We want to recover, for each t: \( P(X_t|y_0,\ldots,y_t) \)
Tracking as induction

• Base case:
  – Start with initial prior that predicts state in absence of any evidence: \( P(X_0) \)
  – For the first frame, *correct* this given the first measurement: \( Y_0 = y_0 \)
Tracking as induction

- **Base case:**
  - Start with initial prior that predicts state in absence of any evidence: $P(X_0)$
  - For the first frame, *correct* this given the first measurement: $Y_0 = y_0$

$$P(X_0 \mid Y_0 = y_0) = \frac{P(y_0 \mid X_0)P(X_0)}{P(y_0)} \propto P(y_0 \mid X_0)P(X_0)$$
Tracking as induction

• Base case:
  – Start with initial prior that predicts state in absence of any evidence: $P(X_0)$
  – For the first frame, correct this given the first measurement: $Y_0 = y_0$

• Given corrected estimate for frame $t-1$:
  – Predict for frame $t \Rightarrow P(X_t | y_0, \ldots, y_{t-1})$
  – Observe $y_t$; Correct for frame $t \Rightarrow P(X_t | y_0, \ldots, y_{t-1}, y_t)$
Prediction

- Prediction involves representing $P(X_t | y_0, \ldots, y_{t-1})$ given $P(X_{t-1} | y_0, \ldots, y_{t-1})$

\[
P(X_t | y_0, \ldots, y_{t-1}) = \int P(X_t, X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}
\]

Law of total probability
Prediction

• Prediction involves representing $P(X_t | y_0, \ldots, y_{t-1})$ given $P(X_{t-1} | y_0, \ldots, y_{t-1})$

$$P(X_t | y_0, \ldots, y_{t-1})$$

$$= \int P(X_t, X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}$$

$$= \int P(X_t | X_{t-1}, y_0, \ldots, y_{t-1}) P(X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}$$

Conditioning on $X_{t-1}$
Prediction

• Prediction involves representing $P(X_t | y_0, \ldots, y_{t-1})$ given $P(X_{t-1} | y_0, \ldots, y_{t-1})$

\[
P(X_t | y_0, \ldots, y_{t-1}) = \int P(X_t, X_{t-1} | y_0, \ldots, y_{t-1})dX_{t-1}
\]

\[
= \int P(X_t | X_{t-1}, y_0, \ldots, y_{t-1})P(X_{t-1} | y_0, \ldots, y_{t-1})dX_{t-1}
\]

\[
= \int P(X_t | X_{t-1})P(X_{t-1} | y_0, \ldots, y_{t-1})dX_{t-1}
\]

Independence assumption
Prediction

- Prediction involves representing $P(X_t | y_0, \ldots, y_{t-1})$ given $P(X_{t-1} | y_0, \ldots, y_{t-1})$

\[
P(X_t | y_0, \ldots, y_{t-1}) = \int P(X_t, X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}
\]

\[
= \int P(X_t | X_{t-1}, y_0, \ldots, y_{t-1}) P(X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}
\]

\[
= \int P(X_t | X_{t-1}) P(X_{t-1} | y_0, \ldots, y_{t-1}) dX_{t-1}
\]

dynamics model \hspace{1cm} corrected estimate from previous step
Correction

- Correction involves computing \( P(X_t | y_0, \ldots, y_t) \) given predicted value \( P(X_t | y_0, \ldots, y_{t-1}) \)
Correction

- Correction involves computing \( P(X_t | y_0, \ldots, y_t) \) given predicted value \( P(X_t | y_0, \ldots, y_{t-1}) \)

\[
P(X_t | y_0, \ldots, y_t) = \frac{P(y_t | X_t, y_0, \ldots, y_{t-1})}{P(y_t | y_0, \ldots, y_{t-1})} P(X_t | y_0, \ldots, y_{t-1})
\]

Bayes’ Rule
Correction

• Correction involves computing $P(X_t|y_0,\ldots,y_t)$
given predicted value $P(X_t|y_0,\ldots,y_{t-1})$

\[
P(X_t|y_0,\ldots,y_t) = \frac{P(y_t | X_t, y_0,\ldots,y_{t-1})}{P(y_t | y_0,\ldots,y_{t-1})} P(X_t | y_0,\ldots,y_{t-1})
\]

\[
= \frac{P(y_t | X_t)P(X_t | y_0,\ldots,y_{t-1})}{P(y_t | y_0,\ldots,y_{t-1})}
\]

Independence assumption
(observation $y_t$ directly depends only on state $X_t$)
Correction

• Correction involves computing $P(X_t | y_0, \ldots, y_t)$ given predicted value $P(X_t | y_0, \ldots, y_{t-1})$

$$P(X_t | y_0, \ldots, y_t) = \frac{P(y_t | X_t, y_0, \ldots, y_{t-1}) P(X_t | y_0, \ldots, y_{t-1})}{P(y_t | y_0, \ldots, y_{t-1})}$$

$$= \frac{P(y_t | X_t) P(X_t | y_0, \ldots, y_{t-1})}{P(y_t | y_0, \ldots, y_{t-1})}$$

$$= \frac{P(y_t | X_t) P(X_t | y_0, \ldots, y_{t-1})}{\int P(y_t | X_t) P(X_t | y_0, \ldots, y_{t-1}) dX_t}$$

Conditioning on $X_t$
Correction

- Correction involves computing \( P(X_t | y_0, \ldots, y_t) \) given predicted value \( P(X_t | y_0, \ldots, y_{t-1}) \)

\[
P(X_t | y_0, \ldots, y_t) = \frac{P(y_t | X_t, y_0, \ldots, y_{t-1})}{P(y_t | y_0, \ldots, y_{t-1})} \cdot P(X_t | y_0, \ldots, y_{t-1})
\]

\[
= \frac{P(y_t | X_t)P(X_t | y_0, \ldots, y_{t-1})}{P(y_t | y_0, \ldots, y_{t-1})} \cdot \int P(y_t | X_t)P(X_t | y_0, \ldots, y_{t-1}) dX_t
\]

observation model

predicted estimate

normalization factor
Summary: Prediction and correction

Prediction:

\[ P(X_t \mid y_0, \ldots, y_{t-1}) = \int P(X_t \mid X_{t-1})P(X_{t-1} \mid y_0, \ldots, y_{t-1})dX_{t-1} \]

Correction:

\[ P(X_t \mid y_0, \ldots, y_t) = \frac{P(y_t \mid X_t)P(X_t \mid y_0, \ldots, y_{t-1})}{\int P(y_t \mid X_t)P(X_t \mid y_0, \ldots, y_{t-1})dX_t} \]
The Kalman filter

- Linear dynamics model: state undergoes linear transformation plus Gaussian noise

- Observation model: measurement is linearly transformed state plus Gaussian noise

- The predicted/corrected state distributions are Gaussian
  - You only need to maintain the mean and covariance
  - The calculations are easy (all the integrals can be done in closed form)
Example: Kalman Filter

Observation

Prediction

Ground Truth

Correction

Next Frame

Update Location, Velocity, etc.
Comparison

Ground Truth  Observation  Correction
Decent model if there is just one object, but localization is imprecise
Particle filtering

Represent the state distribution non-parametrically

- Prediction: Sample possible values $X_{t-1}$ for the previous state
- Correction: Compute likelihood of $X_t$ based on weighted samples and $P(y_t|X_t)$

M. Isard and A. Blake, CONDENSATION -- conditional density propagation for visual tracking, IJCV 29(1):5-28, 1998
Particle filtering

Start with weighted samples from previous time step

Sample and shift according to dynamics model

Spread due to randomness; this is predicted density $P(X_t|Y_{t-1})$

Weight the samples according to observation density

Arrive at corrected density estimate $P(X_t|Y_t)$

M. Isard and A. Blake, CONDENSATION -- conditional density propagation for visual tracking, IJCV 29(1):5-28, 1998
Propagation of non-parametric densities

Current state

deterministic drift

Expected change

p(x)

Good if there are multiple, confusable objects (or clutter) in the scene

p(x)

Observation and Correction

stochastic diffusion

reactive effect of measurement

Uncertainty

x

x
Particle filtering results

People: http://www.youtube.com/watch?v=wCMk-pHzScE

Hand: http://www.youtube.com/watch?v=tljufInUqZM
Tracking issues

• Initialization
  – Manual
  – Background subtraction
  – Detection
Tracking issues

• Initialization

• Getting observation and dynamics models
  – Observation model: match a template or use a trained detector
  – Dynamics model: usually specify using domain knowledge
Tracking issues

• Initialization

• Obtaining observation and dynamics model

• Uncertainty of prediction vs. correction
  – If the dynamics model is too strong, will end up ignoring the data
  – If the observation model is too strong, tracking is reduced to repeated detection
Tracking issues

• Initialization
• Getting observation and dynamics models
• Prediction vs. correction
• Data association
  – When tracking multiple objects, need to assign right objects to right tracks (particle filters good for this)
Tracking issues

- Initialization
- Getting observation and dynamics models
- Prediction vs. correction
- Data association
- Drift
  - Errors can accumulate over time
Drift

Things to remember

• Tracking objects = detection + prediction

• Probabilistic framework
  – Predict next state
  – Update current state based on observation

• Two simple but effective methods
  – Kalman filters: Gaussian distribution
  – Particle filters: multimodal distribution
Next class: action recognition

• Action recognition
  – What is an “action”?  
  – How can we represent movement?  
  – How do we incorporate motion, pose, and nearby objects?