Human Pose Tracking I: Basics

David Fleet
University of Toronto

CIFAR Summer School, 2009
Looking at People
Challenges: Complex pose / motion

People have many degrees of freedom, comprising an articulated skeleton overlaid with soft tissue and deformable clothing.
Challenges: Complex movements

People move in complex ways, often communicating with subtle gestures.
Challenges: Appearance, size and shape

People come in all shapes and sizes, with highly variable appearance.
Challenges: Appearance variability

Image appearance changes dramatically over time due to non-rigidity of body and clothing and lighting.
Challenges: Appearance variability

Image appearance changes dramatically over time due to non-rigidity of body and clothing and lighting.
Challenges: Context dependence

Perceived scene context influences object recognition.

[Courtesy of Antonio Torralba]
Challenges: Noisy and missing measurements

Ambiguities in pose are commonplace, due to
- background clutter
- apparent similarity of parts
- occlusions
- loose clothing
- …
Challenges: Depth and reflection ambiguities

Multiple 3D poses may be consistent with a given image.

[courtesy of Cristian Sminchisescu]
Model-based pose tracking

Video input

3D articulated model
Outline

- Introduction
- Bayesian Filtering
- Kinematic Motion Models
- Discriminative Pose Estimation
- Physics-Based Motion Models
- Concluding remarks
Bayesian Filtering

State: n-vector comprising variables to be inferred: $s_t$
- continuous variables [eg., position, velocity, shape, size, ...]
- discrete state variables [eg., # objects, gender, activity, ... ]
- state history: $s_{1:t} = (s_1, ..., s_t)$

Observations: data from which we estimate state: $z_t = f(s_t)$
- observation history: $z_{1:t} = (z_1, ..., z_t)$
Bayesian Filtering

Posterior distribution over states conditioned on observations

\[ p(s_{1:t} \mid z_{1:t}) \]

Bayes’ rule:

\[ p(s_{1:t} \mid z_{1:t}) = \frac{p(z_{1:t} \mid s_{1:t}) p(s_{1:t})}{p(z_{1:t})} \]

Filtering distribution: marginal posterior at current time

\[ p(s_t \mid z_{1:t}) = \int_{s_1} \int_{s_{t-1}} p(s_{1:t} \mid z_{1:t}) \]

likelyhood

prior

independent

of state
1\textsuperscript{st}-order Markov model for state dynamics:

\[ p(s_t \mid s_{1:t-1}) = p(s_t \mid s_{t-1}) \]

so

\[ p(s_{1:t}) = \left( \prod_{j=2}^{t} p(s_j \mid s_{j-1}) \right) p(s_1) \]

Conditional independence of observations

\[ p(z_{1:t} \mid s_{1:t}) = p(z_t \mid s_t) p(z_{1:t-1} \mid s_{1:t-1}) \]

\[ = \prod_{\tau=1}^{t} p(z_\tau \mid s_\tau) \] likelihood at time \( \tau \)
Recursive form of filtering/posterior distribution

Filtering distribution:

\[ p(s_t \mid z_{1:t}) = \int_{s_1} \cdots \int_{s_{t-1}} p(s_{1:t} \mid z_{1:t}) \]

\[ = c \; p(z_t \mid s_t) \; \underbrace{p(s_t \mid z_{1:t-1})}_{	ext{likelihood}} \]

Prediction distribution (temporal prior):

\[ p(s_t \mid z_{1:t-1}) = \int_{s_{t-1}} p(s_t \mid s_{t-1}) \; p(s_{t-1} \mid z_{1:t-1}) \]
Bayesian smoothing

Inverting the dynamics permits inference backwards in time:

\[ p(s_\tau \mid z_{\tau:t}) = c \, p(z_\tau \mid s_\tau) \int_{s_{\tau+1}} p(s_\tau \mid s_{\tau+1}) \, p(s_{\tau+1} \mid z_{\tau+1:t}) \]

\[ = c \, p(z_\tau \mid s_\tau) \, p(s_\tau \mid z_{\tau+1:t}) \]

Smoothing distribution (forward-backward belief propagation):

\[ p(s_\tau \mid z_{1:t}) = \frac{c}{p(s_\tau)} \, p(z_\tau \mid s_\tau) \, p(s_\tau \mid z_{1:\tau-1}) \, p(s_\tau \mid z_{\tau+1:t}) \]

Batch Algorithms (smoothing): Estimation of state sequences using the entire observation sequence (i.e., all past, present & future data):
- optimal and efficient but not always applicable

Online Algorithms (filtering): Casual estimation of \( x_t \) occurs as observations become available, using present and past data only.


**Kalman filter**

Assume linearity and Gaussianity for the observation and dynamical models:

\[
\begin{align*}
    s_t &= A s_{t-1} + \eta_d & \eta_d &\sim \mathcal{N}(0, C_d) \\
    z_t &= M s_t + \eta_m & \eta_m &\sim \mathcal{N}(0, C_m)
\end{align*}
\]

**Key Result:** Prediction and filtering distributions are Gaussian, so they may be represented by sufficient statistics:

\[
\begin{align*}
    p(s_t \mid z_{1:t-1}) &= \int_{s_{t-1}} p(s_t \mid s_{t-1}) p(s_{t-1} \mid z_{1:t-1}) \sim \mathcal{N}(s_t^-, C_t^-) \\
    p(s_t \mid z_{1:t}) &= c p(z_t \mid s_t) p(s_t \mid z_{1:t-1}) \sim \mathcal{N}(s_t^+, C_t^+)
\end{align*}
\]
Depiction of filtering

\[ p(s_{t-1} | z_{1:t-1}) \]  
**deterministic drift**

\[ p(s_t | z_{1:t}) \]  
**incorporate data**

\[ p(s_t | z_{1:t-1}) \]  
**stochastic diffusion**
Kalman filter

First well-known uses in computer vision:

- Road following by tracking lane markers
  

- Rigid structure from feature tracks under perspective projection
  
Measurement clutter and occlusion often cause multimodal likelihoods.
Object motion and interactions between objects often produce complex nonlinear dynamics (so Gaussianity is not preserved)
Approximate inference

Coping with multimodal, non-Gaussian distributions

- Optimization (to find MAP solution)
  - e.g., WSL tracker
- Monte Carlo approximations
WSL tracker

**Goal:** Tracking with precise alignment over long times

**Problem:** Changing appearance and unmodeled deformations

**Key:** Use ‘stable’ properties of appearance for tracking
WSL tracker
Monte Carlo inference (Particle filters)

Approximate the filtering distribution using point samples:

- By drawing a set of random samples from the filtering distribution, we could use samples statistics to approximate expectations

Let $S = \{s^{(j)}\}$ be a set of $N$ fair samples from distribution $\mathcal{P}(s)$, then for functions $f(s)$

$$
E_S[f(s)] \equiv \frac{1}{N} \sum_{j=1}^{N} f(s^{(j)}) \xrightarrow{N \to \infty} E_{\mathcal{P}}[f(s)]
$$

**Problem:** we don’t know how to draw samples from $p(s_t \mid z_{1:t})$
Importance sampling

weighted samples
Importance sampling

Weighted sample set \( S = \{ s^{(j)}, w^{(j)} \} \)

- draw samples \( s^{(j)} \) from a *proposal distribution* \( Q(s) \), with weights \( w^{(j)} = w(s^{(j)}) \), then

\[
E_S [f(s)] \equiv \sum_{j=1}^{N} w^{(j)} f(s^{(j)}) \xrightarrow{N \to \infty} E_Q [w(s) f(s)]
\]

- If \( w(s) = \mathcal{P}(s)/Q(s) \) then weighted sample statistics approximate expectations under \( \mathcal{P}(s) \), i.e.,

\[
E_Q [w(s) f(s)] = \int w(s) f(s) Q(s) \, ds \\
= \int f(s) \mathcal{P}(s) \, ds \\
= E_{\mathcal{P}} [f(s)]
\]
Simple particle filter approximates the filtering distribution by drawing samples from the prediction distribution:

$$p(s_t | z_{1:t}) = c \ p(z_t | s_t) \ p(s_t | z_{1:t-1})$$

$$w(s) = \frac{\mathcal{P}(s)}{\mathcal{Q}(s)}$$
Particle filter

Simple particle filter approximates the filtering distribution by drawing samples from the prediction distribution:

\[ p(s_t \mid z_{1:t}) = c \ p(z_t \mid s_t) \ p(s_t \mid z_{1:t-1}) \]

With resampling at each time step:

\[ p(s_{t-1} \mid z_{1:t-1}) \rightarrow p(s_t \mid s_{t-1}) \rightarrow p(z_t \mid s_t) \rightarrow p(s_t \mid z_{1:t}) \]

posterior → temporal dynamics → likelihood → posterior

[Gordon et al '93; Isard & Blake '98; Liu & Chen '98, ...]
Particle filter

Given a weighted sample set $S = \{s^{(j)}_{t-1}, w^{(j)}_{t-1}\}$, the prediction distribution is a mixture model

$$p(s_t | z_{1:t-1}) = \sum_{j=1}^{N} w^{(j)} p(s_t | s^{(j)}_{t-1})$$

To draw samples from it:
- sample a component of the mixture by the treating weights as mixing probabilities
- then sample from the associated dynamics pdf $p(s_t | s^{(i)}_{t-1})$
Particle filter

[Isard and Blake, IJCV '98]
Lessons learned: Sampling efficiency

3D Kinematic Model
(28D state: 22 joint angles, 6 global DOFs)

[Choo & Fleet, ICCV ‘01]
Likelihood and dynamics

Given the state, $s$, and the articulated model, the 3D marker positions $X_j$ onto the 2D image plane:

$$d_j(s) = T_j(X_j; s)$$

Observation model:

$$\hat{d}_j = d_j + \eta_j, \quad \eta_j \sim \mathcal{N}(0; \sigma^2_m I_2)$$

Likelihood of observed 2D locations, $D = \{\hat{d}_j\}$:

$$p(D | s) \propto \exp \left( -\frac{1}{2\sigma^2_m} \sum_j || \hat{d}_j - d_j(s) ||^2 \right)$$

Smooth dynamics:

$$s_t = s_{t-1} + \epsilon_t$$

where $\epsilon_t$ is isotropic Gaussian for translational & angular variables
Performance

Estimator Variance:
- multiple runs with independent noise & sampling
- variance measured as MSE from ground truth (from MCMC)
Problem: Sampling efficiency

Number of samples needed depends on the effective volumes (entropies) of the prediction and posterior distributions.

- With random sampling from the prediction density, the number of particles grow exponentially for samples to fall on states with high posterior. E.g., for $D$-dim spheres, with radii $R$ and $r$,

  \[ N > \left( \frac{R}{r} \right)^D \]

- effective number of ‘independent’ samples: $N_e = 1 / \sum_j (w^{(j)})^2$
Hybrid Monte Carlo filter

Improved sampling through MCMC:
- initialize a set of particles from a particle filter
- select a subset from which to initiate MCMC with stochastic gradient search (hybrid Monte Carlo)

[Choo & Fleet, ICCV '01]
Mean estimates on independent trials

Particle Filter

Hybrid MC Filter

Black: Ground truth (at frame 10)
Red: Mean state from 6 random trials
Lessons learned: Effective proposals

If proposal and target distributions differ significantly, then:

- most particle weights are near zero, and some modes get no samples, so the normalization constant $c$ can be wildly wrong.

Prediction distributions $Q = p(s_t | z_{1:t-1})$ make poor proposals: dynamics are often uncertain, and likelihoods are often peaked.

Use the current observation to improve proposals.

- Let $\mathcal{D}(z_t)$ be a continuous distribution obtained from some detector that yields target locations (e.g., a Gaussian mixture).
- Then, just modify the proposal density and importance weights:

$$Q = \mathcal{D}(z_t) p(s_t | z_{1:t-1}) \quad \text{with} \quad w(s_t) = \frac{c p(z_t | s_t)}{\mathcal{D}(z_t)}$$
Lessons learned: Proper likelihoods

Do not compare states using different sets of observations. Explain the entire image or use likelihood ratios.

E.g., let pixels intensities, conditioned on state be independent where $D_f$ and $D_b$ are disjoint sets of foreground and background pixels, and $p_f$ and $p_b$ are the respective likelihood functions.

Divide $p(I \mid s)$ by the background likelihood of all pixels (i.e., as if no target is present):

$$p(I \mid s) \propto \frac{\prod_{y \in D_f} p_f(I(y) \mid s) \prod_{y \in D_b} p_b(I(y))}{\prod_{y} p_b(I(y))}$$

$$= \frac{\prod_{y \in D_f} p_f(I(y) \mid s) \prod_{y \in D_b} p_b(I(y))}{\prod_{y \in D_f} p_b(I(y) \mid s) \prod_{y \in D_b} p_b(I(y))}$$

$$= \prod_{y \in D_f} \frac{p_f(I(y) \mid s)}{p_b(I(y))}$$
Lessons Learned: Use the right state space

Despite the potential to approximate multimodal posteriors, tracking multi targets with a single target state space is unadvised.
Finding occlusion boundaries

Motion boundaries yield information about position and orientation of surface boundaries, and about relative surface depths
Finding occlusion boundaries

Estimation of smooth motion and occlusion boundaries on hybrid random fields with non-parametric Bayesian inference.

[Nestares and Fleet, CVPR ‘01]
Finding lips and lip reading

Probabilistic detection, tracking, and recognition of motion events in video, with learned models of image motion.

[Fleet, Black, Yacoob and Jepson, IJCV 2000]
Human pose tracking

Estimate the three-dimensional structure of people from video, with constraints on their shape, size, & motion.

[Sidenbladh, Black and Fleet, ECCV 2000]
Selected references


Jepson A. et al, Robust online appearance models for visual tracking. *IEEE Trans PAMI* 25(10), 2003
