Tracking and Kalman Filtering

CMPE 264: Image Analysis and Computer Vision
Spring’02, Hai Tao
5/8/02
Object tracking

Object tracking is the problem of estimating the positions and other relevant information of moving objects in image sequences.

- Two-frame tracking can be accomplished using correlation-based matching methods, optical flow techniques, or change-based moving object detection methods.
- The main difficulties in reliable tracking of moving objects include:
  - Rapid appearance changes caused by image noise, illumination changes, nonrigid motion, and varying poses.
  - Occlusion.
  - Cluttered Background.
  - Interaction between multiple objects.
- In a long image sequence, if the dynamics of the moving object is known, prediction can be made about the positions of the objects in the current image. This information can be combined with the actual image observation to achieve more robust results.
Correlation-based tracking

For a given region in one frame, find the corresponding region in the next frame by finding the maximum correlation score in a search region
Change-based tracking

Algorithm

- Align the background using a parametric motion model, e.g., a homography
- Image subtraction to detect motion blobs
  - Compute the difference image between two frames
  - Thresholding to find the blobs
  - Locate the blob center as the position of the object
Change-based tracking

Example
The best linear unbiased estimator (BLUE)

An optimal estimation algorithm – produce optimal estimates of the state of a dynamic system, on the basis of noisy measurements and an uncertain model of the system’s dynamics.

To derive the Kalman filter, we will first introduce the best linear unbiased estimator (BLUE)

\[
y = Hx + n
\]

where \( y \): observation, \( x \): state, \( H \): measurement matrix, \( n \): noise \( N(0,R) \).

Given \( y, R, H \), estimate \( x \)

- Linear filter: \( \hat{x} = Ly \)
- Unbiased: \( E(\hat{x} - x) = 0 \)
- Best: Covariance \( E[(\hat{x} - x)^T (\hat{x} - x)] \) is minimum
- Lemma: The linear estimator is unbiased iff \( I = LH \)

Proof:

\[
E[x - \hat{x}] = E[x - Ly] = E[x - L(Hx + n)]
\]
\[
= E[(I - LH)x - Ln] = E[(I - LH)x]
E[(I - LH)x] = 0 \text{ iff } LH = I
\]
The best linear unbiased estimator (BLUE)

Theorem: The best linear unbiased estimator \( \hat{x} = L\mathbf{v} \) for the measurement model \( \mathbf{y} = H\mathbf{x} + \mathbf{n} \) is \( L = (H^T R^{-1} H)^{-1} H^T R^{-1} \) and the covariance matrix of the estimate is \( P = (H^T R^{-1} H)^{-1} \).

Proof: This is equivalent to

\[
\min_L \| E[(L\mathbf{y} - \mathbf{x})(L\mathbf{y} - \mathbf{x})^T] \|
\]
subject to \( LH = I \)

Observe that \( E[(L\mathbf{y} - \mathbf{x})(L\mathbf{y} - \mathbf{x})^T] = E[L\mathbf{n}\mathbf{n}^T L^T] = LRL^T \)

Suppose a solution can be written as \( L = L_0 + (L - L_0) \)
then it is obvious that \( (L - L_0)H = LH - L_0H = I - I = 0 \).

The covariance can be written as

\[
P = LRL^T = (L_0 + (L - L_0))R(L_0 + (L - L_0))^T
\]
\[
= L_0R_L^T + (L - L_0)RL_0^T + L_0R(L - L_0)^T + (L - L_0)R(L - L_0)^T
\]

Since \( RL_0^T = R[(H^T R^{-1} H)^{-1} H^T R^{-1}]^T = H (H^T R^{-1} H)^{-1} \)
therefore \( (L - L_0)RL_0^T = (L - L_0)H (H^T R^{-1} H)^{-1} = 0 \), similarly \( L_0R(L - L_0)^T = [(L - L_0)RL_0^T]^T = 0 \)
As the result \( P = L_0RL_0^T + (L - L_0)R(L - L_0)^T \)
Since both terms are positive definite or semidefinite matrices, it is minimum when \( L = L_0 \).
Kalman filter – problem statement

- **Dynamic system and the state**

  \[ x_k = \Phi_{k-1} x_{k-1} + \xi_{k-1} \]
  \[ z_k = H_k x_k + \mu_k \]

  where \( x_k \) is the a state vector at time instant \( k \), e.g. \( x_k = [x_k, y_k, v_{x_k}, v_{y_k}] \),

  \( z_k \) is the a observation vector at time instant \( k \), e.g. \( z_k = [z_x, z_y] \),

  \( \Phi_{k-1} \) is the state transition matrix, e.g. \( \Phi_{k-1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \)

  \( H_k \) is the measurement matrix, e.g. \( H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \)

  \( \xi_{k-1} \) is a random vector modelling the uncertainty of the model, assumed to be \( N(0, Q_{k-1}) \)

  \( \mu_k \) is a random vector modelling the additive noise in the observation, assumed to be \( N(0, R_k) \)
Kalman filter – derivation

Suppose the prediction is $\hat{x}^-$, with covariance matrix $P_k^-$. They are based on observations before $k$. As the result, if the true state is $x_k$

$$\hat{x}^-_k = x_k + e_k : N(0, P^-_k)$$

Another piece of information is

$$z_k = H_k x_k + \mu_k : N(0, R_k)$$

This can be written as

$$\begin{bmatrix} \hat{x}^-_k \\ z_k \end{bmatrix} = \begin{bmatrix} I \\ H_k \end{bmatrix} x_k + n : N\left(0, \begin{bmatrix} P^-_k & 0 \\ 0 & R_k \end{bmatrix}\right)$$

The BLUE estimator of this system is

$$\hat{x}_k = P_k [I, H_k^T] \begin{bmatrix} (P^-_k)^{-1} & 0 \\ 0 & R_k^{-1} \end{bmatrix} \begin{bmatrix} \hat{x}^-_k \\ z_k \end{bmatrix} = P_k [(P^-_k)^{-1} \hat{x}^-_k + H_k^T R_k^{-1} z_k]$$

$$P_k^{-1} = \begin{bmatrix} [I, H_k^T] \begin{bmatrix} (P^-_k)^{-1} & 0 \\ 0 & R_k^{-1} \end{bmatrix} \begin{bmatrix} I \\ H_k \end{bmatrix} \end{bmatrix} = (P^-_k)^{-1} + H_k^T R_k^{-1} H_k$$

Therefore

$$\hat{x}_k = P_k [(P^-_k)^{-1} \hat{x}^-_k + H_k^T R_k^{-1} z_k] = P_k [(P_k^{-1} - H_k^T R_k^{-1} H_k) \hat{x}^-_k + H_k^T R_k^{-1} z_k]$$

$$=\hat{x}^-_k + P_k H_k^T R_k^{-1} (z_k - H_k \hat{x}^-_k)$$

This is the update stage of the Kalman filter
Kalman filtering

Propagation stage:
\[ \hat{x}_{k+1}^- = \Phi_k \hat{x}_k \]
\[ P_{k+1}^- = E[(\hat{x}_{k+1}^- - x_{k+1})(\hat{x}_{k+1}^- - x_{k+1})^T] \]
\[ = E[(\Phi_k \hat{x}_k - \Phi_k x_k - \xi_k)(\Phi_k \hat{x}_k - \Phi_k x_k - \xi_k)^T] \]
\[ = \Phi_k P_k \Phi_k^T + Q_k \]

Update equations

Propagation equations

\[ \hat{x}_{k+1}^- = \Phi_k \hat{x}_k \]
\[ P_{k+1}^- = \Phi_k P_k \Phi_k^T + Q_k \]
Kalman filtering - algorithm

Kalman_filtering_algorithm

- Initialize $Q_0, R_0, \Phi_k = \Phi, H_k = H, x_0 = z_0,$ and $P_0$ as a large covariance matrix
- For each time instant $k=1,…$
  - Predict $P_k, x_k$ using the propagation equations
  - Compute the gain $K_k$ and update $P_k, \hat{x}_k$ using the update equations
  - Output the estimate $\hat{x}_k$ and optionally the covariance matrix $P_k$
Hidden Markov model

- Hidden Markov model (HMM)
  - $x_k$ is called the hidden state and $z_k$ is called the observation
  - Markov chain
    \[
    p(x_k \mid x_1, \ldots, x_{k-1}) = p(x_k \mid x_{k-1})
    \]
    \[
    p(z_k \mid x_1, \ldots, x_k) = p(z_k \mid x_k)
    \]
  - As the result
    \[
    p(x_1, \ldots, x_k, z_1, \ldots, z_k) = p(x_1)p(z_1 \mid x_1) \prod_{i=2}^{k} [p(x_i \mid x_{i-1})p(z_i \mid x_i)]
    \]
Forward algorithm

To compute the \textit{a posterior} probability \( p(x_k \mid z_1, \ldots, z_k) \), we can use the forward algorithm

\[
p(x_k \mid z_k, \ldots, z_1) \propto p(z_k \mid x_k, z_{k-1}, \ldots, z_1) p(x_k \mid z_{k-1}, \ldots, z_1)
\]

\[
= p(z_k \mid x_k) \int p(x_k, x_{k-1} \mid z_{k-1}, \ldots, z_1) dx_{k-1}
\]

\[
= p(z_k \mid x_k) \int p(x_k \mid x_{k-1}, z_{k-1}, \ldots, z_1) p(x_{k-1} \mid z_{k-1}, \ldots, z_1) dx_{k-1}
\]

\[
= p(z_k \mid x_k) \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid z_{k-1}, \ldots, z_1) dx_{k-1}
\]

Notice that Bayes rules is used for computing the \textit{a posterior} probability

\[
p(x \mid z) = \frac{p(z \mid x)p(x)}{p(z)}
\]
Probabilistic interpretation of the Kalman filter

\[ N(\hat{x}_{k-1} : x_{k-1}, P_{k-1}) \quad p(x_k | x_{k-1}) = N(x_k^- : \Phi_{k-1} \hat{x}_{k-1}, \Phi_{k-1} P_{k-1} \Phi_{k-1}^T + Q_{k-1}) \]

\[ \begin{array}{c}
x_{k-1} \\
\Phi_{k-1} \\
\end{array} \xrightarrow{} \begin{array}{c}
x_k \\
H_k \\
\end{array} \]

\[ p(z_k | x_k) = N(H_k x_k : z_k, R_k) \]

\[ p(x_k | z_k, z_{k-1}, ..., z_1) \propto p(z_k | x_k, z_{k-1}, ..., z_1) p(x_k | z_{k-1}, ..., z_1) \]

\[ = p(z_k | x_k) p(x_k | z_{k-1}, ..., z_1) = p(z_k | x_k) \int p(x_k | x_{k-1}) p(x_{k-1} | z_{k-1}, ..., z_1) dx_{k-1} \]

\[ = N(H_k x_k : z_k, R_k) \int_{x_{k-1}} N(x_k : \Phi_{k-1} x_{k-1}, \Phi_{k-1} P_{k-1} \Phi_{k-1}^T + Q_{k-1}) dx_{k-1} \]
Homework

- In the Kalman filtering equations, does $P_k$ depend on the observation data? Does $K_k$ depend on observation data?
- Optional: Using the probabilistic interpretation to derive the Kalman equations