Tracking

* determine motion of object in image

E.g., given a video (sequence of images)
determine position of object ~~at~~ in each image frame

* See Eddie's video (bead motion)

object is characterized by its <u>state</u>

* position
* heading
* other

} at each step (or time)

RV: $\bar{X}_i^{⊕}$ e.g., $\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_i^{⊕}$

instance $\bar{x}_i$

Is also good to have a <u>model</u> of how object state changes with time

* don't have this in bead motion

We also may have <u>observations</u> (or measurements) of state or related variables at each step

RV $\bar{Y}_i$ e.g., $\begin{bmatrix} y_x \\ y_y \end{bmatrix}_i^{⊕}$

instance $\bar{y}_i$

11/2

Simple Tracking Strategies

* tracking by detection : appearance in image
* tracking by matching : motion model

Detection

* Find stable properties (color, size)
* Build probabilistic models

$$p\left(\text{object} \mid \text{image features}\right) = \frac{p(\text{image features} \mid \text{object})\, p(\text{object})}{p(\text{image features})}$$

e.g., in red bead:

red channel is > 100
red channel is > 1.2 * green channel
red channel is > 1.2 * blue channel
→ find statistics ?

* Background subtraction

Take all images (or subset) and average them. Then look at each image for largest difference.

Tracking Linear Dynamical Models with Kalman Filters

$\bar{X}_i$ state

$\bar{Y}_i$ observations

need to determine

$$P(\bar{X}_k \mid \bar{Y}_0, \bar{Y}_1, \ldots, \bar{Y}_k)$$

assume

* $P(\bar{Y}_k \mid \bar{X}_0, \bar{X}_1, \ldots \bar{X}_N, \bar{Y}_0, \ldots \bar{Y}_n) = P(\bar{Y}_k \mid \bar{X}_k)$

* $P(\bar{X}_k \mid \bar{X}_0, \ldots \bar{X}_{k-1}) = P(\bar{X}_k \mid \bar{X}_{k-1})$

recursive formulation for tracking:

p. 346

Prediction (process model)

$$\bar{x}_i^- = D_i \bar{x}_{i-1}^+$$

D_i is matrix
i.e., process is linear

$$\Sigma_i^- = \Sigma_{d_i} + D_i \Sigma_{i-1}^+ D_i$$

Correction (observation model)

$$K_i = \Sigma_i^- M_i^T \left[M_i \Sigma_i^- M_i^T + \Sigma_{m_i} \right]^{-1}$$

$$\bar{x}_i^+ = \bar{x}_i^- + K_i [\bar{y}_i - M_i \bar{x}_i^-]$$

$$\Sigma_i^+ = [Id - K_i M_i] \Sigma_i^-$$

usually : process given by

$$\bar{x}_t = A_t \bar{x}_{t-1} + B_t \bar{u}_t + \varepsilon_t$$

$\uparrow$ next state $\quad$ $\uparrow$ previous state $\quad$ $\uparrow$ control variable $\quad$ $\uparrow$ noise

$$p(\bar{x}_t \mid \bar{u}_t, \bar{x}_{t-1}) = \det(2\pi R_t)^{-\frac{1}{2}}$$
$$\exp\left\{-\tfrac{1}{2}(\bar{x}_t - A_t\bar{x}_{t-1} - B_t\bar{u}_t)^T R_t^{-1}(\bar{x}_t - A_t\bar{x}_{t-1} - B_t\bar{u}_t)\right\}$$

R_t is covariance on process

observation :

$$\bar{z}_t = C_t \bar{x}_t + \delta_t$$

Q_t is covariance of observation

$$p(\bar{z}_t \mid \bar{x}_t) = \det(2\pi Q_t)^{-1/2} \exp\left\{-\tfrac{1}{2}(\bar{z}_t - C_t\bar{x}_t)^T Q_t^{-1}(\bar{z}_t - C_t\bar{x}_t)\right\}$$

11/5

Simple Example: Heat Flow (1D)

$$\frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} \cancel{+ \frac{\partial^2 T}{\partial y^2}}$$ heat flow equation

convert to finite difference

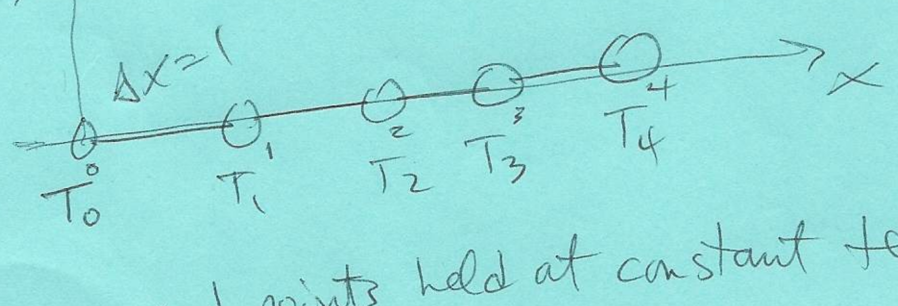
$$\frac{\partial T}{\partial t} = \frac{T_i^{k+1} - T_i^k}{\Delta t}$$ ← time step

T_i^k ← step, ← location

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^k - T_i^k + T_{i-1}^k}{\Delta x^2}$$ Δx = spacing

$$\Rightarrow \frac{T_i^{k+1} - T_i^k}{\Delta t} = \frac{k\left(T_{i+1}^k - T_i^k + T_{i-1}^k\right)}{\Delta x^2}$$

$$\Rightarrow T_i^{k+1} = T_i^k + \frac{k \Delta t}{\Delta x^2}\left(T_{i+1}^k - T_i^k + T_{i-1}^k\right)$$

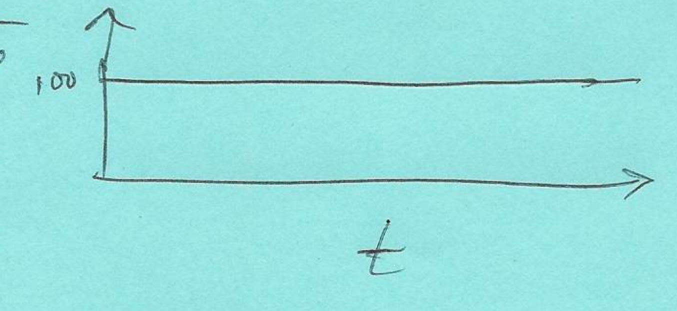


Assume end points held at constant temperature

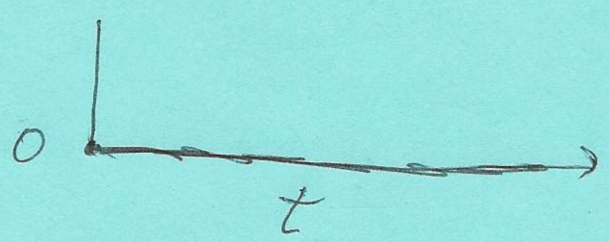
11) 6

Plot temperature at T_0

$100 = T_0$



$0 = T_4$



Assume $T_1^0 = T_2^0 = T_3^0 = 0$

<u>Determine</u>: $T_1^k \quad k = 1, 2, \ldots$

$T_2^k \quad \ldots$

$T_3^k \quad \ldots$

E.g., assume $k = 1$ & $\Delta t = 0.1$

$$T_1^1 = T_1^0 + \frac{k\,\Delta t}{\Delta x^2}\left(T_2^0 - T_1^0 + T_0^0\right)$$

$$= 0 + \frac{1 * 0.1}{1}(0 - 0 + 100) =$$

$$= 10$$

11\7

Can use this to compute temperatures.

Now, notice that each state variable element is a linear combination of other elements:

$$\begin{bmatrix} T_0 \\ T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}^{k} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \alpha & 1-\alpha & \alpha & 0 & 0 \\ 0 & \alpha & 1-\alpha & \alpha & 0 \\ 0 & 0 & \alpha & 1-\alpha & \alpha \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_0 \\ T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}^{k-1} + \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix}$$

$$\bar{X}^{k} = D\bar{X}^{k-1} + \epsilon$$

Be able to write a program to produce next state from process model + current state

---

Need to include noise:

$$\bar{X}^{k} = D\bar{X}^{k-1} + \varepsilon$$

how do we get ε?

for each state element, sample from σ_i randn

Observation of Temperature

$$\bar{Y}^k = M\bar{X}^k + \delta$$

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix}^k = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}^{k-1} + \begin{bmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix}$$

We can now run a simulation

<u>function</u> T_next = CS5320_heat_flow_step(D, T, R, Q, k, del_t, del_x)

<u>function</u> T_trace = CS5320_heat_flow_sim(D, T, R, Q, k, del_t, del_x, max_step)

Constant Velocity Model

$$\bar{P}_i = \bar{P}_{i-1} + \Delta t\, \bar{V}_{i-1}$$

$$\bar{V}_i = \bar{V}_{i-1}$$

$$\bar{x} = \begin{bmatrix} \bar{P} \\ \bar{V} \end{bmatrix} = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}$$

$$D_i = \begin{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & \Delta t \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{bmatrix} \equiv \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Process

$$\bar{P}_i = \bar{P}_{i-1} + \Delta t\, \bar{V}_{i-1} + \bar{\varepsilon}$$

$$\bar{V}_i = \bar{V}_{i-1} + \bar{\varepsilon}$$

$$\bar{x}_i = D_i \bar{x}_{i-1} + \bar{\varepsilon}$$

---

tr = cs5320.const_vel(0,0,1,0,0.1,R)

R = 0.0001 * eye(4,4);

R(3:4,3:4) = 0;

Constant Acceleration

$$\bar{p}_i = \bar{p}_{i-1} + \Delta t\, \bar{v}_{i-1}$$

$$\bar{v}_i = \bar{v}_{i-1} + \Delta t\, \bar{a}_{i-1}$$

$$\bar{a}_i = \bar{a}_{i-1}$$

$$\bar{x} = \begin{bmatrix} \bar{p} \\ \bar{v} \\ \bar{a} \end{bmatrix}$$

$$D_i = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Process

$$\bar{x}_i = D_i \bar{x}_{i-1} + \bar{\varepsilon}$$

```
tr = CS5320_const_acc(0,10,0,0,0,-9.8,0.1,3,R);
    R = eye(6,6);
    R(5:6,5:6) = 0;
    R = 0.0001*R;
```

11\11

Observation

constant velocity

$$\begin{bmatrix} y_x \\ y_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + \delta \;;$$

δ ↑ Q : covariance

```
function y = CS5320_odometry(x, M, Q)
%
y = M*x + mvnrnd(zeros(size(x)), Q)';
y = [y; x(3:4)];
```

Kalman Filter

each step: (to estimate state by combining process model and observation):

Prediction from model

state $\bar{x}_i^- = D_i \bar{x}_{i-1}^+$

covariance $\Sigma_i^- = \Sigma_{d_i} + D_i \Sigma_{i-1}^+ D_i^T$

process covariance R

observation covariance Q

Correction from observation

Kalman gain $K_i = \Sigma_i^- M_i^T [M_i \Sigma_i^- M_i^T + \Sigma_{m_i}]^{-1}$

state $\bar{x}_i^+ = \bar{x}_i^- + K_i [\bar{y}_i - M_i \bar{x}_i^-]$

covariance $\Sigma_i^+ = [Id - K_i M_i] \Sigma_i^-$

11\13

Run simulation + estimation

see CS5320_const_acc_Kalman

use for tracking

change sensor to look at images.
Estimate variance in x + y
synthetic image vs. real image sequences