CSE 573: Homework Assignment 5

Name: Sokhey, Harshdeep , UB#50247213

July 23, 2018

1 Solution 1

1.1 Part 1

The video assumes a simplistic model where the input to the car is the velocity.
Thus $x_k = \{position\}$, $C = 1$.

The car dynamics can be written as :

$$\begin{aligned} x_k =& Ax_{k-1} + Bu_k + w_k \\ y_k =& Cx_k + v_k \end{aligned}$$

In the above equation 1.1, the v_k is the noise component of the y_k which can be caused due to the noisy GPS readings. . The w_k is the noise component of the x_k which can be caused due to changes in wind or changes in car velocity, etc. Both v_k and w_k noise is random in nature and obeys a Gaussian distribution, represented by, $v \sim \mathcal{N}(0, R)$, $w \sim \mathcal{N}(0, Q)$.

The car's position can be predicted by multiplying the prediction probability estimate, $\hat{x}_k$ and measurement probability,$\hat{y}_k$. The resultant is also a Gaussian function. The mean of the resultant gaussian function gives the optimal estimate of the car's position with minimal variance.

1.2 Part 2

The Kalman Filter Algorithm gets the optimal estimate in 2 steps:

1. **Prediction**, given by the equations :

$$\begin{aligned} \hat{x_k}^- =& A\hat{x}_{k-1} + Bu_k \\ P_k{}^- =& AP_{k-1}A^T + Q \end{aligned}$$

   The $\hat{x_k}^-$ is the predicted state estimate and $P_k{}^-$ is the error covariance, calculated from the process noise and $\hat{x_{k-1}}$. During the initial state, values for $\hat{x_{k-1}}$, $P_k{}^-$ come from the initial estimates.

2. **Update**, given by the equations:

$$\begin{aligned} K_k =& \frac{P_k^- C^T}{CP_k^- C^T + R} \\ \hat{x_k} =& \hat{x_k}^- + K_k(y_k - C\hat{x_k}^-) \\ P_k =& (I - K_kC)P_k{}^- \end{aligned}$$

The update step uses the *Aprior* estimates from the prediction step and updates them to calculate posterior estimates of the states and error co-variance. The *Kalman Gain* , K_k , minimizes the aposterior error covariance,P_k. The Kalman gain helps determine the contribution of the aprior estimate,$\hat{x}^-$, and the measurement term, y_k, to the calculation of the optimal state estimate $\hat{x_k}$. Mathematically, it can be written as:

- Case 1: Measurement Error is **low**

$$\lim_{R\to 0} K_k = \lim_{R\to 0} \frac{P_k^- C^T}{CP_k^- C^T + R} = \frac{\cancel{P_k^- C^T}}{C\cancel{P_k^- C^T} + 0} = \frac{1}{C} \tag{1}$$

  Applying the value of K_k from Case 1 in $\hat{x_k}$, we get,

$$\begin{aligned}\hat{x_k} =& \hat{x_k}^- + K_k(y_k - C\hat{x_k}^-)\\ =& \cancel{\hat{x_k}} + \frac{y_k}{C} - \cancel{\hat{x_k}}\\ \hat{x_k} =& y_k\end{aligned}$$

- Case 2: Aprior Error is **low**

$$\lim_{P_k^-\to 0} K_k = \lim_{\lim_{P_k^-\to 0}\to 0} \frac{P_k^- C^T}{CP_k^- C^T + R} = 0 \tag{2}$$

  Applying the value of K_k from Case 2 in $\hat{x_k}$, we get,

$$\begin{aligned}\hat{x_k} =& \hat{x_k}^- + 0(y_k - C\hat{x_k}^-)\\ \hat{x_k} =& \hat{x_k}^-\end{aligned}$$

The estimates($\hat{x_k}, P_k$) calculated from the update step become the inputs for the prediction step in the next epoch.

2 Solution 2

2.1 Part 1

There are three essential steps for estimating **Circular Hough Transforms**:

- The foreground pixels of high gradient, called candidate pixels, are allowed to cast *votes* in the accumulator array. The candidate pixels vote in pattern around them that forms a full circle of a fixed radius.
- The votes of candidate pixels corresponding to the image circle tend to accumulate at the accumulator array bin corresponding to the center. The circle centers are estimated by detecting the peaks in the accumulator array.
- The radii can be estimated using the information from the circle centers and the accumalator array.

2.2 Part 2

A **Deformable Part Models (DPM)** have the ability to identify difficult objects as a set of parts constrained in the spatial arrangement they can take.

A DPM can be trained by representing an image using component objects. Since the positions of the parts are not fixed, the model is a deformable one and penalizes the parts that differ spatially. We assume the positions of the parts are fixed for the frame and the part detector is trained using SVM. The trained SVM classifier is used to find the optimal part position in each training example.

2.3 Part 3

Eigenfaces is a set of eigenvectors used for face recognition. The set can be generated by performing a principal component analysis on a large set of different face images.

3 Solution 3

CSE	573LEC	L	Comp Vision & Image Proc	LEC	07/17/18	07/29/18	07/13/18	Yes	Retake