Exercise 07 for MA-INF 2201 Computer Vision WS18/19
25.11.2018
Submission on 1.12.2018

1. **Theory Question**
Prove the following property of k dimensional Gaussian distributions $\text{Norm}_x[\mu, \Sigma]$:
$$\int \text{Norm}_x[a, A]\text{Norm}_x[b, B]dx = \text{Norm}_a[b, A+B] \int \text{Norm}_x[\Sigma_*(A^{-1}a + B^{-1}b), \Sigma_*]dx$$
where $\Sigma_* = (A^{-1} + B^{-1})^{-1}$.
(5 points)

2. **Kalman Filtering**
You need to implement the basic Kalman Filtering algorithm. You observe a set of 2D noisy observations (x_i, y_i) which are the coordinates of the 2D space. They are observed coordinated of a rotating object as shown in Figure 1.

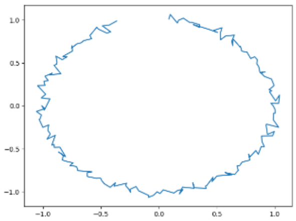

Figure 1: Observations from location of a clockwise rotating object.

State: The state of the object should be the 4D vector (x, y, v_x, v_y) which denote the location and the velocity in each axis.
Initial State: You should consider the initial state of $(0, 1, 0, 0)$.
Time Evolution Equation: What should be the time evolution equation?
Measurement Equation: What should be the measurement equation?
Code for reading observations is provided. You should write code for performing the filtering. You may use numpy for matrix operations. At the end visualize the filtered output. Use the template `kalman.py`.
(5 points)

3. **Tracking Using Particle Filter** The goal is to track the fish in the sequence 1:32.png (in the *images* folder). The initial fish bounding box in the first frame is given. Use the template `track_fish.py`.
The template for the tracker is the color histogram of the initial bounding box.
State model: Using your own implementation of a Particle Filter, track the fish in 2D space assuming the size of the bounding box is constant and the hidden state is two-dimensional position (x, y).
Measurement model: The measure of fitness between any particle and the template is the Bhattacharya distance d_{BH} between the template histogram and the histogram of the particle.
Likelihood: The likelihood can be modelled by a Gaussian proportional to $exp(-\frac{d_{BH}}{2\sigma_2^2})$.
Motion model: The motion model for iteration $n+1$ is normally distributed with mean μ_n and variance σ_1^2. While σ_1 remains constant, μ is updated according to
$$\mu_{n+1} = d_\mu * \mu_n + (1 - d_\mu)(m_{n+1} - m_n)$$

$$\mu_0 = 0$$

where m_n denotes the mean position of the fitness weighted samples in iteration n.
Find a configuration for d_μ and σ_1 that works well.
OpenCV hints: `calcHist`, `compareHist`
(10 points)