Matlab Exercises

Lecture 4 – Bayesian tracking

1) **Motion Model**

Write a function that generates a 1D random motion of the following type:

- Brownian motion
- WNA
- Constant acceleration (a=9.81) + perturbation

Where w = Gauss(0,σ=1), Δt=0.1 for all the situations.

For each case, plot a corresponding trajectory in time: p(t), with t=(0, Δt, 2Δt,…, NΔt), N=1000.

Afterwards, compute and plot the corresponding probabilistic motion models $P(\mathbf{s}_t|\mathbf{s}_{t-1})$ (in 1 or 2 dimensions, depending on the case)

2) **Measurement model**

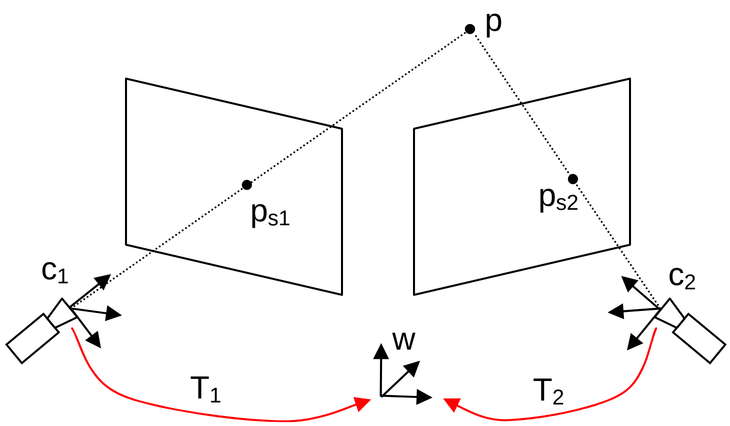
Write the measurement model $\mathbf{z} = h(\mathbf{s},\mathbf{v})$ for the following case:

Suppose a point in space **p** with camera coordinates (x,y,z) is being projected on the screen (x_s,y_s), and the measurement instrument is the camera (intrinsic parameters (f,r_x,r_y)) + an image processing algorithm that identifies the screen coordinates with a Gaussian uncertainty **v** = Gauss([0,0],Λ=I)

Hint: First identify the variables **s** and **z** (which and how many), then write down the function $\mathbf{z} = h(\mathbf{s},\mathbf{v})$.

3) **Measurement model (stereo)**

Consider the following case:



Suppose to have two cameras in a stereo configuration, and again one point in space **p**. The two camera frames c_1, c_2 have fixed poses T_1, T_2 with respect to a world coordinate system w, so that the point **p** transform in space as:
$^{c1}\mathbf{p} = T_1\, {}^{w}\mathbf{p}$, and $^{c2}\mathbf{p} = T_2\, {}^{w}\mathbf{p}$ (homogeneous coordinates).

The point in space with coordinates $^{w}\mathbf{p} = (x,y,z)$ is being projected on the two screens (x_{s1},y_{s1}), (x_{s2},y_{s2}), and the cameras have the same intrinsic parameters (f,r_x,r_y).
The measurement instrument is: the 2 cameras + an image processing algorithm that identifies the screen coordinates with independent Gaussian errors $\mathbf{v}_1$, $\mathbf{v}_2$ Gauss($[0,0],\Lambda=I$).

Write down the measurement function $\mathbf{z} = h(\mathbf{s},\mathbf{v})$.
(Hint: the same as before: first identify the variables, then write the function)

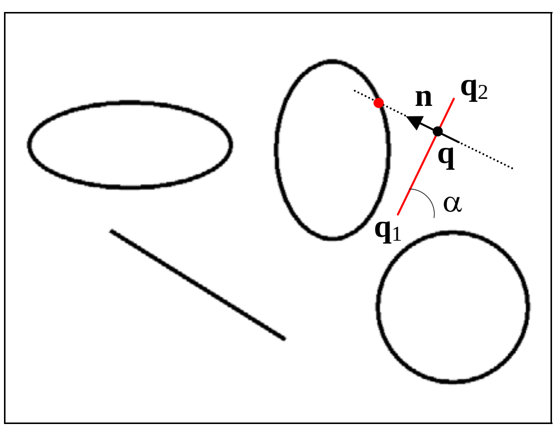
4) **Likelihood function (edges)**

Consider now a segment model: the state s is (x_1,y_1,α) the position and orientation of the segment (the length is 10).

$\mathbf{q} = k\mathbf{q}_1(s) + (1-k)\mathbf{q}_2(s)$; k runs from 0 to 1

where $\mathbf{q}_1 = (x_1,y_1)$, $\mathbf{q}_2 = (x_1+10\cos(\alpha), y_1+10\sin(\alpha))$

The measurement is the image z = I.
The likelihood model is an “expected image” I_{exp} given the segment hypothesis **s** + an uncertainty model.
The expected image is an image that contains a segment exactly in the position **s**.
The uncertainty is measured by the distance along the normal direction of the nearest edge, where every point along the segment **q** contribute independently to this error.
So, we can model the uncertainty P(error) as a product of single uncertainties $P(err_i)$, for every point, each one given by a Gaussian centered in the segment point. In the ideal case ($I=I_{exp}$) we have the maximum Likelihood $P(I|s)$ (the nearest edges are exactly on the segment hypothesis).



With this model, given an image (example above) write a Matlab function that computes the Likelihood P(z|s):

- take a set of 11 equidistant points in the segment (k=0,0.1,0.2,…,1)
- take the normal vector to the segment **n** = (-sin(α), cos(α))
- From a point **q**(k_i), search along the normal directions **q**(k_i)+j**n** where j= 0,-1,1,-2,2,…,-L,L (up to a length L=5 in the two directions)
- Stop the search if a black pixel is found (pixel coordinates → the points **q**+j**n** need to be rounded to integers), or if the maximum distance L is reached
- The result is the distance of the nearest edge, l_{ok}, or L
- Now weight the distance l_{ok} with a Gaussian: P(l_{ok}) = Gauss(0,1)
- Finally, multiply all the 11 Gaussians, to get the Likelihood P(**z**|**s**)

5) **Measurement model : 3D projection**

Suppose we have

- a set of 3D points ${}^{B}p_1, \ldots, {}^{B}p_N$ (body frame referred),
- the 6 pose parameters **s** = ($\alpha,\beta,\gamma,t_x,t_y,t_z$) in Euler angles + translation vector.
- A set of measured features **z** = ($q_1,\ldots,q_N$) on the image

We model the uncertainty as N independent Gaussians: Gauss(err_i,1) where the error is the re-projection error of feature i (distance between expected and measured point).

→ P(**z**|**s**) is the product of N Gaussians.

Write a Matlab function that computes the Likelihood function P(**z**|**s**) for this case, with the 3 inputs specified above.