AE 640A: Autonomous Navigation
Assignment #2

Due on Monday, March 6, 2018 (11:59 pm)

Problem 0: Warm up

Familiarize yourself with the Point Cloud Library. As C++ will be the language of choice throughout this course, please stick to those tutorials where needed.

Problem 1: Plane Segmentation in 3-D Point Cloud

In this problem, you need to perform the task of segmentation of the given point cloud data available here. *(50 points)*

First create a new package named **cloudpub_<roll_number>**. Implement the following nodes for this package:

- A node named `cb_publisher`. The node shall advertise an image topic named `cb_cloud`, containing a generated point cloud sampled from a multivariate Gaussian distribution. Your node should have the following ROS parameters:
  - N - the number of points to be present in the advertised cloud
  - *mean* - mean of the point cloud distribution
  - *variance* - variance of the point cloud distribution
  - *frequency* - the frequency with which to publish the point cloud

  You need to specify default values to these parameters. Mean and variance are defined for the (x, y, z) coordinates so would be a vector of dimension 3 and a matrix of dimension 3×3 respectively. Assume the coordinates are uncorrelated.
- A node named `file_publisher`, that loads the given point cloud file from disk and advertises it under the topic `cloud_raw`. You are allowed to use PCL for loading the cloud. *Hint: Check the tutorial on reading point cloud file.* Your node should have the following parameters:
  - *file* - the path to the cloud file to load
  - *frequency* - frequency used for publishing the image (use a default parameter here)

  To visualize your results you can use the `rviz` node in the `rviz` package.

NOTE: You are free to refer to the point cloud library (PCL) tutorials for this. Kindly ensure that the published message is of type PointCloud2 and the Header is defined properly with the `frame_id = "camera_link"` *for both the parts.*

You now need to create a new package named **cloud_segmenter_<roll_number>** for segmenting a plane out of the point cloud using the RANSAC algorithm. Implement the following nodes for this package:

- A node named `cloud_segmenter_algo`. The new node should have the following ROS parameters:
  - n - minimum number of data points required to fit the model

- w - probability of choosing an inlier each time a single point is selected
- p - probability that the algorithm selects only inliers from the input data set after some iterations while choosing the n points to estimate model parameters
- t - threshold value to determine when a data point fits a model
- d - number of close data points required to assert that a model fits well to data

You need to specify default values to these parameters. Assume the outliers constitute 10% of the given data and the probability of choosing an inlier set is 0.95.

After subscribing to a point cloud topic `cloud_raw`, you need to perform the RANSAC operation over it. The algorithm is described below:

1. select n data items at random
2. estimate parameters of the model
3. find how many data items (of M) fit the model within a user given tolerance
4. if number of inlier are big enough, accept fit and exit with success
5. repeat steps 1-4 N times
6. fail if you get here

where there are M data items in total and the parameters can be estimated from n data items. The number of iterations N can be calculated using the formula: $N = \frac{log(1-p)}{log(1-w^n)}$.

- A node named `cloud_segmenter_pcl`. The new node is similar to the `cloud_segmenter_algo` node, however instead of writing your own RANSAC implementation, you need to use the off-the shelf functions available in the point cloud library (PCL). More details are available on the tutorial here.

  To visualize your results you can use the `rviz` node in the `rviz` package.

NOTE: You need to publish the segmented point cloud to a topic named `segmented_output`. *The inlier in the point cloud should have the color green while the outliers should have the color red. Kindly ensure that the published message is of type PointCloud2 and the Header is defined properly with the* `frame_id = "camera_link"` *for both the parts.*

Problem 2: Probability Theory

1. Prove $P(A|B,C) = \frac{P(B|A,C)P(A|C)}{P(B|C)}$. *(5 points)*
2. Prove $P(x_t|z_t, x_{t-1}, u_{t-1}) = \frac{P(z_t|x_t,x_{t-1},u_{t-1})P(x_t|x_{t-1},u_{t-1})}{P(z_t|x_{t-1},u_{t-1})}$. *(5 points)*
3. Prove $\frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{+\infty} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = 1$. *(5 points)*
4. Find mean and variance of x_{t+1}, given that $x_{t+1} = Fx_t + Gu_t$ where $x_t \sim \mathcal{N}(\hat{x}_t, \Sigma_t)$, $u_t = u + \eta_t$, $\eta_t \sim \mathcal{N}(0, Q_t)$. *(10 points)*

Problem 3: Least Square Estimation

There are 3 buildings A, B, and C. Suppose you have an instrument that measures the relative height of the building from the point where you are standing. Since the instrument is erroneous, you take multiple measurements. Suppose the height of each of these buildings measured from ground is 24.64 mt, 38.80 mt and 48.30 mt. The height of building B measured from the top of A is 14.22 mt.The height of C measured from A is 23.55 mt, and the height of C measured from B is 9.5 mt. Find the least squares estimate of the height of buildings A, B, C. *(15 points)*

Problem 4: Kalman Filter

Consider the following linear Gaussian system: *(25 points)*

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t, \\ y_t &= Hx_t + \eta_t, \end{aligned}$$

where $x_t \in \mathbb{R}^{n_x}$ is the state vector, $u_t \in \mathbb{R}^{n_u}$ is the input vector, $w_t \in \mathbb{R}^{n_w}$ is a disturbance vector acting on the system, $y_t \in \mathbb{R}^{n_y}$ is the output vector, and $\eta_t \in \mathbb{R}^{n_\eta}$ is a noise vector acting on the output. The initial state is assumed to be Gaussian $x_0 \sim \mathcal{N}(\hat{x}_0, \Sigma_{x_0})$. The disturbance w_t and η_t has the known probability distribution $w_t \sim \mathcal{N}(0, \Sigma_{w_t})$ and $\eta_t \sim \mathcal{N}(0, \Sigma_{\eta_t})$. Show that the Kalman filter is optimal using the covariance matrix $P_{t|t} = \mathbf{E}[(x_t - x_{t|t})(x_t - x_{t|t})^T]$.

Problem 5: Joint Probability Distribution

Let X and Y denote two random variables that are jointly Gaussian: *(20 points)*

$$P(X,Y) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp\left(-\frac{1}{2} \begin{bmatrix} X - \mu_x \\ Y - \mu_y \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} X - \mu_x \\ Y - \mu_y \end{bmatrix} \right)$$

where μ_x and μ_y are mean and $\Sigma = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{bmatrix}$ is the covariance matrix of random variable X and Y. Show that

$$P(X|Y = y) = \frac{1}{\sqrt{2\pi\sigma_*^2}} \exp\left(-\frac{(X - \mu_*)^2}{2\sigma_*^2} \right)$$

where $\mu_* = \mu_x + \frac{\sigma_{xy}}{\sigma_y^2}(y - \mu_y)$ and $\sigma_* = \sigma_x^2 - \frac{\sigma_{xy}^2}{\sigma_y^2}$.

Problem 6: Random Variable Transformation

Let X and Y be i.i.d random variables having uniform distribution $U[0, 1]$. Find the probability density functions of following random variables: *(15 points)*

1. $Z_1 := log(X)$
2. $Z_2 := exp(X)$
3. $Z_3 := X + Y$