ASSIGNMENT 5

1)

a) An outlier is a point that is not part of the data any might be harmful to the solution

All the points in the data that are close to each other nicely fit the model. They do not affect the error much. But when we have one point which ~~which~~ is different, the error is very large and because of this the ~~point~~ point becomes very influencial and we do not want the wrong points to be the most influencial.

b) The objective function used for robust estimation is

$$E(\theta) = \sum \rho_\sigma(d(x_i;\theta)) \quad \text{— (1)}$$

Standard least squares objective function

$$E(\theta) = \sum d^2(x_i;\theta) \quad \text{— (2)}$$

The differance is that (1) uses $\rho_\sigma(x)$ whereas (2) uses x^2.

If $\rho_\sigma(x) = x^2$ then the equations are exactly the same.

c) Geman McClure function for robust estimation:

$$\rho_\sigma(x) = \frac{x^2}{x^2+\sigma^2}$$

The Geman McClure function has an upper bound. The square function takes the error and squares them and it has no upper bound.

In Geman McClure function no matter how high the error is the upper bound is stagnant.

If σ is too small, we ~~are~~ are narrowing the selection to a small neighbourhood which will not fit the line correctly since there will be very few relevant points.

If σ is too large, we include all the points and in turn we ~~don~~ do not acheive any benefit of identifying outliers.

To get the correct σ we start with higher value of σ and as we fit the model we make the value of σ smaller at each step as we converge.

Steps to follow:

→ Draw a large set of points uniformly at random
→ Select initial value of σ
→ Fit model → $\Theta^{(i)}$
→ Compute $\sigma^{(i)}$ using median distance of points
→ Continue until objective is decreasing.

→ Steps to be repeated.

d) The principle of RANSAC algorithm is:-
- Perform multiple experiments.
- Choose best results
- Use small sets in hope that at least one set will not have outliers.

The number of points drawn at each attempt should be small because it is easier to come up with a set without any outliers as opposed to larger sets.

e) Parameters of the RANSAC algorithm:

n = number of points drawn at each evaluation

d = minimum number of points needed to estimate model

K = number of trials

t = distance threshold to identify inliers.

Number of trials, $k = \frac{\log(1-p)}{\log(1-w^n)}$

f) Objective of image segmentation:

→ separate objects from background

→ find contours of objects

→ label each pixel in the image with class label.

In agglomerative approach, we start with each pixel in a separate cluster ~~whereas in the divisive approach we start with~~ and merge clusters with small distance whereas in divisive approach we start with all pixels in one cluster and split clusters to produce large distance between them.

g) K means is used for partitioning n observations into K clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.
The Gaussian mixture model is similar to k-means but with different measure distance.

h) The mean-shift algorithm is similar to k-means, where it basically ~~Instead of ...~~, adds weights to each distance and the closer the data point is to the center, the higher weight it has.

2)
a) Forward projection is when P_i and M are given and image point needs to be computed.
Calibration is when image and world points are given and M needs to be computed.
Reconstruction is when image point and M are given and world point needs to be computed.
Forward projection is the easiest
Reconstruction is the hardest.

2) b) The necessary input for camera calibration is two files in which each row will have a world point and a corresponding image point in meters and pixels respectively.

c) The steps in non coplanar calibration algorithm are:

(i) Finding the projection matrix M

Using $p_i' = M \underline{P}_i$ where $p_i = \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$, $\underline{P}_i = \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$

~~we get~~

$$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} - m_1^T - \\ - m_2^T - \\ - m_3^T - \end{bmatrix} \underline{P}_i$$

$$\left.\begin{aligned} x_i' &= m_1^T \underline{P}_i \\ y_i' &= m_2^T \underline{P}_i \\ w_i' &= m_3^T \underline{P}_i \end{aligned}\right\} \quad \begin{aligned} x_i &= \frac{x_i'}{w_i'} = \frac{m_1^T \underline{P}_i}{m_3^T \underline{P}_i} \Rightarrow m_1^T \underline{P}_i - x_i m_3^T \underline{P}_i = 0 \\ y_i &= \frac{y_i'}{w_i'} = \frac{m_2^T \underline{P}_i}{m_3^T \underline{P}_i} \Rightarrow m_2^T \underline{P}_i - y_i m_3^T \underline{P}_i = 0 \end{aligned}$$

For each point in M we have 2 equations with 12 unknowns.

We need to have at least 6 point pairs for calibration

~~Using the~~ Converting the obtained equations into matrix form we get

For ~~a~~ single point:

$$\begin{bmatrix} \underline{P}_i^T & 0 & -x_i \underline{P}_i^T \\ 0 & \underline{P}_i^T & -y_i \underline{P}_i^T \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

For 'm' points:

$$\begin{bmatrix} \underline{P}_1^T & 0 & -x_1 \underline{P}_1^T \\ 0 & \underline{P}_1^T & -y_1 \underline{P}_1^T \\ \vdots & \vdots & \vdots \\ \underline{P}_m & 0 & -x_m \underline{P}_m^T \\ 0 & \underline{P}_m^T & -y_m \underline{P}_m^T \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

which is $Ax = 0$

This is solved using SVD

$A = UDV^T$

Solution is the column of V belonging to zero singular value.

$$\hat{x} = \begin{bmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_3 \end{bmatrix} \Rightarrow \hat{M} = \begin{bmatrix} - M_1^T - \\ - M_2^T - \\ - M_3^T - \end{bmatrix}$$

The solution is not unique

$$A\hat{x} = 0$$

we also have $A(\rho\hat{x}) = 0$

$\Rightarrow \rho\hat{x}$ is a solution

where ρ is the scaling factor

(ii) Finding the parameters using M

After estimating M we need to find ρ such that

$$M = K^* [R^* | T^*] = \rho\hat{M}$$

and we need to find the parameters K^*, R^* and T^*

$$K^* = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R^* = \left[r_1^T \quad r_2^T \quad r_3^T \right]^T$$

$$T^* = \epsilon |\rho| (K^*)^{-1} b$$

d) $$M = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 0 & 3 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \qquad P_i = (1, 2, 3)$$

$$P_i' = M \underline{P_i}$$

$$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 0 & 3 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1+4+9+4 \\ 1+0+9+4 \\ 1+2+3+1 \end{bmatrix}$$

$$\begin{bmatrix} x_i' \\ y_i' \\ w_i' \end{bmatrix} = \begin{bmatrix} 18 \\ 14 \\ 7 \end{bmatrix}$$

$$x_i = \frac{18}{7} = 2.57 \qquad y_i = \frac{14}{7} = 2$$

Therefore $(x_i, y_i) = (2.57, 2)$

e) ~~We know~~

Given world point $(X, Y, Z) = (1, 2, 3)$
and image point $(x_i, y_i) = (100, 200)$

The matrix needed to solve for M for a single point is given as

$$\begin{bmatrix} \underline{P}_i^T & 0 & -x_i P_i^T \\ 0 & \underline{P}_i^T & -y_i P_i^T \end{bmatrix} \quad \text{where } \underline{P}_i^T = [X_i \; Y_i \; Z_i \; 1]$$

Substituting the values we ~~get~~ get

$$\begin{bmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -x_iX_i & -x_iY_i & -x_iZ_i & -x_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -y_iX_i & -y_iY_i & -y_iZ_i & -y_i \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2 & 3 & 1 & 0 & 0 & 0 & 0 & -100 & -200 & -300 & -100 \\ 0 & 0 & 0 & 0 & 1 & 2 & 3 & 1 & -200 & -400 & -600 & -200 \end{bmatrix}$$

f) The minimum number of points necessary to be able to find a unique solution of M is 6.

We have the equations

$$m_1^T \underline{P}_i - X_i m_3^T \underline{P}_i = 0$$
$$m_2^T \underline{P}_i - y_i m_3^T \underline{P}_i = 0$$

Using the above equation for a single point we have

$$\begin{bmatrix} \underline{P}_i^T & 0 & -x_i \underline{P}_i^T \\ 0 & \underline{P}_i^T & -y_i \underline{P}_i^T \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

For 'm' points

$$\begin{bmatrix} \underline{P}_1^T & 0 & -x_1 \underline{P}_1^T \\ 0 & \underline{P}_1^T & -y_1 \underline{P}_1^T \\ \vdots & \vdots & \vdots \\ \underline{P}_m^T & 0 & -x_m \underline{P}_m^T \\ 0 & \underline{P}_m^T & -y_m \underline{P}_m^T \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

which is of the form

$$A\underline{x} = 0$$

To solve this use SVD

$$A = UDV^T$$

The solution would be the column of V belonging to zero singular value.

We get

$$\hat{x} = \begin{bmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_3 \end{bmatrix} \Rightarrow \hat{M} = \begin{bmatrix} - \hat{m}_1^T - \\ - \hat{m}_2^T - \\ - \hat{m}_3^T - \end{bmatrix}$$

12×1

where $\hat{M}$ is the estimated projection matrix

The solution is not unique if given $A\hat{x} = 0$

we also have $A(\rho\hat{x}) = 0 \Rightarrow \rho\hat{x}$ is a solution

where ρ is the scaling factor

There can be multiple solutions but that does not matter because after homogenizing they will all give the same results.

g) We use the orthoganality principle of r_1, r_2, r_3 which means

$$r_1 \cdot r_2 = 0 \quad r_2 \cdot r_3 = 0 \quad r_1 \cdot r_3 = 0$$

$$r_1 \times r_2 = r_3 \quad r_2 \times r_3 = r_1 \quad r_1 \times r_3 = r_2$$

Using this ~~pm~~ principle we get the following results.

$$|\rho| = \frac{1}{|a_3|}$$

$$u_0 = |\rho|^2 a_1 \cdot a_3 \quad \left(\text{using } \rho a_1^T \cdot \rho a_3^T\right)$$

$$v_0 = |\rho|^2 a_2 \cdot a_3 \quad \left(\text{using } \rho a_2^T \cdot \rho a_3^T\right)$$

$$\alpha_v = \sqrt{\rho^2 a_2 \cdot a_2 - v_0^2} \quad \left(\text{using } \rho a_2 \cdot \rho a_2\right)$$

$$s = \frac{1}{\alpha_v} \rho^4 (a_1 \times a_3) \cdot (a_2 \times a_3) \quad \left(\text{using } \rho a_1 \times \rho a_3\right)$$

To find sign of ρ

$\rho = \epsilon|\rho|$ where ϵ is the sign ($+$ or $-$)

$K^* T^* = \rho b = \epsilon|\rho| b$

Assuming object is in front of camera

$[K^* T^*]_z = \epsilon|\rho| b_z$

$\Rightarrow \epsilon = \text{sign}(b_z)$

To find T^*, build K^* from recovered intrinsic parameters

$K^* T^* = \epsilon|\rho| b$

$T^* = (K^*)^{-1} \epsilon|\rho| b$

h) To compute the quality of the projection matrix M estimate we use the formula:

$$E = \frac{1}{m} \sum_i \left(\left\| x_i - \frac{m_1^T \underline{P}_i}{m_3^T \underline{P}_i} \right\|^2 + \left\| y_i - \frac{m_2^T \underline{P}_i}{m_3^T \underline{P}_i} \right\|^2 \right)$$

where $\frac{m_1^T \underline{P}_i}{m_3^T \underline{P}_i}$ and $\frac{m_2^T \underline{P}_i}{m_3^T \underline{P}_i}$ are the predicted x ~~cop~~ and Y coordinate after projecting the point P_i and x_i and y_i are ~~to tea~~ x and Y coordinate that we know we shold ~~get~~ get and E is the average error. ~~...~~

If E is small we can say that the matrix M is of ~~good~~ good quality and ~~if~~ if E is large then the matrix M is of bad quality.

i) For planor camera calibration:

- → Estimate 2D homography between calibration plane and image (for several images)
- → Estimate intrinsic parameters.
- → Compute extrinsic parameters for view of interest.

The planar calibration uses a 2D homography (projective map) which is a 3x3 matrix whereas non coplanar calibration uses a 3x4 projection matrix.

Planar calibration uses multiple images rather ~~tha~~ than a single image as in the case of non coplanar calibration

j) A 2D projective map is a 3x3 matrix which projects a 2DH point to a 2DH point whereas a projection matrix M is a 3x4 matrix ~~[illegible]~~ which ~~map~~ projects a 3DH to a 2DH point.

The assumption we make is that the 'z' ~~[illegible]~~ coordinate of all the world points on the calibration target is 0.