Name : Arpit Hasmukhbhai Patel
ID : A20424085
Course : CS-512 (computer vision)
Semester : Fall-18

[1]

a) outliners are points that are different from all other points
- outliners do not fit the right model that is why they give wrong solution.

(b) objective function used for robust estimation is:

$$E(\theta) = \sum_{i=1}^{n} \rho_{\sigma}(d(x_i;\theta))$$

→ The standard least error equation is more sensitive to outliners

(c) Germam-McClure function for robust estimation:

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

- Its advantage is that it lower the influence of outliners.
- If σ is bigger the open range will bigger. if σ is smaller. the model become more selective of outlinear.

(d) The main principle of RANSAC algorithm is to ~~find~~ use minimum number of points to fit the model and try this process several times and choose the best model after many trials. Number of points drawn at each ~~attempt~~ should be small because there are less chance of getting outliner and get a better model.

(e) The parameters for RANSAC algorithm

n = number of points drawn at each evalution

d = minimum number of points needed to estimate model

K = number of trials

t = distance to determine inliners

=) Formula for estimating the number of trials K:

$$K = \frac{\log(1-P)}{\log(1-w^n)}$$

P = Probability that atleast one of the trials will succeed

w = probability that a point is inliner

n = number of points drawn at each trial.

(f) Objective of image segmentation is to separate foreground from the background.

Merge approch: start with each pixel in separate cluster iteratively merge cluster.

split approch: Start with all pixels in one cluster iteratively split cluster.

(g) K means -

- select K
- select initial guess of K-means : $m_1 \ldots$
- Assign $l_i = \underset{j \in [1,K]}{\operatorname{argmin}} \|f_i - m_j\|^2$ for each pixels & assign correspondent cluster.
- re compute mean : $m_j = \frac{\sum_{i \in s_j} f_i}{\# s_j}$
- stop when m_j doesn't change.

Mixture of Gaussian segmentation :

- It is like k-means with additional parameters.
- Replacing $d = ||f_i - m_j||^2$ with $d = (f_i - m_j)^T$
- Replacing $d = ||f_i - m_j||^2$ with $d = (f_i - m_j)^T$. $\Sigma_i^{-1} (f_i - m_j)$

$$m_j = \frac{\sum_{i \in j} f_i}{\# S_j} \qquad \Sigma_j = \frac{\Sigma (f_i - m_j)(f_i - m_j)^T}{\# S_j}$$

$$d = (f_i - m_j)^T \Sigma_{ij}^{-1} (f_i - m_j)$$

distance

mean of cluster

covariance of cluster

(h) mean shift is similar to K-means.

- Instead of m_j in K-means use weighted from sample to the mean.

$$m_j = \frac{\sum_{i \in j} w(f_i - m_j) f_i}{\sum_{j \in j} w(f_i - m_j)}$$

- The closest a sample is to the mean, more weighted or effect to the mean.

(2) Camera Calibration :-

(a) Forward Projection :- given a 3D world point P project into the image using the projection matrix M

* Camera Calibration : given the image coordinate and the world coordinate of the object. find the camera parameters (internal & external) used in the projection.

* Reconstruction : given image point p, compute world point P.

→ Forward projection is easiest and Reconstruction is most difficult.

(b) For camera calibration we need a set of 3D world points and it's 2D corresponding points.

(c) Step 1 : estimate projection matrix M
step 2 : find parameter (K^*, R^*, T^*) given matrix M.

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

$$p_i = MP_i = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 0 & 3 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 18 \\ 14 \\ 7 \end{bmatrix}}_{2DH}$$

In 2D $p_i = \begin{bmatrix} 18/7 \\ 2 \end{bmatrix}$

(f) we need at least 6 points to get matrix M because we need to solve 11 unknown in order to solve it we need 6 points to get 12 equations.

(e) World point = (1, 2, 3)
image point = (100, 200)

so, for matrix:

$$\Rightarrow \begin{bmatrix} P_i^T & 0^T & -x_i P_i^T \\ 0^T & P_i^T & -y_i P_i^T \\ & & \end{bmatrix} \begin{bmatrix} [M_1] \\ [M_2] \\ [M_3] \end{bmatrix} = \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix}$$

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

(g) The principle that is used to extract the unknown parameters from the projection matrix M is:

$$m = R^* [K^* \mid T^*]$$

- We use the fact that the rotation matrix has the orthogonal vectors along the rows.
- we take ~~a~~ dot product and cross product of the rows in 'M' and this will cancel out some unknown parameters.

Scanned by CamScanner

(h) we need to compute the error

$$E(K^*, R^*, T^*) = \sum_{i=1}^{n} \left(x_i - \frac{M_1^T P_i}{M_3^T P_i} \right) + \left(y_i - \frac{M_2^T P_i}{M_3^T P_i} \right)$$

(i) Planar Calibration steps:

(1) Estimate 2D homography between calibration plane and image

(2) Estimate intrinsic parameters.

(3) computer extrinsic parameters for view of interest.

- Planer solve 2DH points. Non-planer solve 3DH points.

(j) $\overset{2DH}{P_i} = K^* [R^* | T^*] \overset{3DH}{P_i}$

$$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = \underbrace{K^* [r_1 \; r_2 \; T^*]}_{M^* \text{ Homography}} \overset{2DH:}{\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}}$$

$$P_i = \begin{bmatrix} x_i \\ y_i \\ 0 \end{bmatrix}$$ assume z coordinate is zero

Scanned by CamScanner