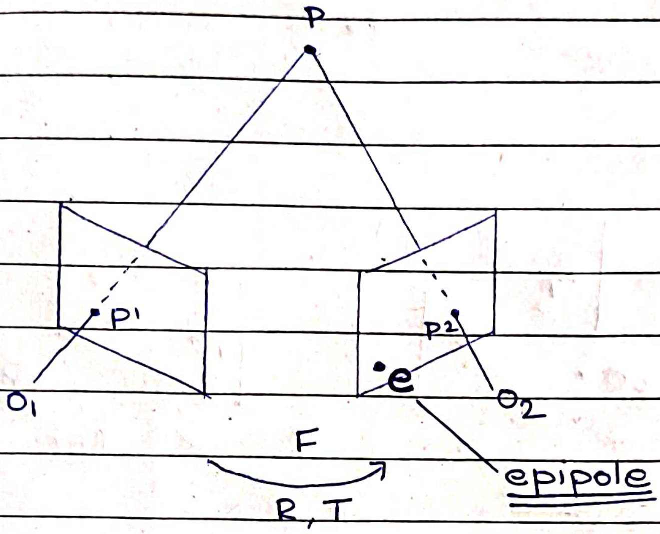
Roll no.
2019111019

Date: ______
Page: ______

Mobile Robotics

Q1 a)



If a point corresponds to P_1 in image 1 and P_2 in image 2
Then we know $P_1^T F P_2 = 0$ and $P_2^T F^T P_1 = 0$.

For any point P, $F^T P_1$ will represent the epipolar line in the second image. This line will always pass through the epipole e.

$$(F^T P_1)^T e = 0$$
$$P_1^T F e = 0$$

Since this will be true for all P_1 $\Rightarrow$ $Fe = 0$.

b) If fundamental matrix between image 1 and image 2 is F
$\Rightarrow P_1^T F P_2 = 0$

then between image 2 and image 1, the fundamental matrix is F^T.

$\Rightarrow P_1^T F P_2 = 0 \Rightarrow (P_1^T F P_2)^T = 0 \Rightarrow P_2^T F^T P_1 = 0$

Date:
Page:

Fundamental matrix is dependent on the camera intrinsics and the transformation between 2 cameras.

$$F_{12} = K^{-T} [t_{12}]_\times R_{12} K^{-1}$$

and since the transformation from camera 1 to 2 and 2 to 1 will be different (as $t_A^B = -R_B^{A\,T} \cdot t_B^A$), the fundamental matrices will be different too.

c) For a point P, if its corresponding non-normalized images are p_1 and p_2 and the camera intrinsics are K.

Then the essential matrix relates the normalized images $= K^{-1}p_1$ and $K^{-1}p_2$

$$\boxed{(K^{-1}p_1)^T E (K^{-1}p_2) = 0} \quad — ①$$

$$p_1^T K^{-T} E K^{-1} p_2 = 0 \quad — ② \quad \text{(on simplifying)}$$

The fundamental matrix relates non-normalized images $= p_1$ and p_2

$$\boxed{p_1^T F p_2 = 0} \quad — ③$$

on comparing ② and ③, we can see:

$$F = K^{-T} E K^{-1}$$

Date:
Page:

Q2 a) Let there be a world point X^w with coordinates X_1^c and X_2^c in the 2 camera frames.

We know camera intrinsics K and rotation R from C_1 to C_2

subscript h represents homogenous coordinates.

$$\text{Image point } 1 = x_1 = K[I\ 0]\, X_{1h}^c$$

$$\text{" " } 2 = x_2 = K[I\ 0]\, X_{2h}^c$$

$$X_1^c = R X_2^c$$

$$\text{or } X_{1h}^c = \begin{bmatrix} R X_2^c \\ 1 \end{bmatrix}$$

$$\text{So } x_1 = K[I\ 0]\begin{bmatrix} R X_2^c \\ 1 \end{bmatrix}$$

$$= K R X_2^c$$

$$x_2 = K X_2^c$$

$$\Rightarrow X_2^c = \lambda K^{-1} x_2$$

$$x_1 = K R X_2^c$$
$$= KR(\lambda K^{-1} x_2)$$
$$= KRK^{-1}(\lambda x_2)$$
$$= KRK^{-1} x_2 \qquad (\text{as } x_1, x_2 \text{ are homogenous})$$

$$x_1 = H x_2$$

This homography relation relates pixel x_1 in image1 to x_2 in image 2 when there is pure rotation.

Date:
Page:

In case of translation:

$$x_1^c = R x_2^c + t$$

$$\text{or } (x_1^c)_h = \begin{bmatrix} R x_2^c + t \\ 1 \end{bmatrix}$$

$$x_1 = K [I \; 0] \begin{bmatrix} R x_2^c + t \\ 1 \end{bmatrix}$$

$$= K (R x_2^c + t)$$

$$= K R x_2^c + K t$$

$$x_1 = K R K^{-1} x_2 + K t$$

Hence the relation b/w x_1 and x_2 changes as a term related to translation is introduced.

b) → Homography - $x_1 = K R K^{-1} x_2$

Here ~~cam~~ same camera (same K) is used in both images and K and R are known. Both images are viewed from same plane but different angle.

or if we know H then - $x_1 = H x_2$

Here both images are from same camera and are viewed from same plane, diff. angle.

→ Fundamental Matrix - $x_1^T F . x_2 = 0$

Here x_1, x_2 are non-normalized, camera is uncaliberated.

To get F we use 8 point algorithm so ~~corr~~ atleast 8 correspondences are known.

→ Essential matrix $x_1^T E x_2 = 0$

x_1 and x_2 are normalized (camera is caliberated)
To get E we use 8 point algo so min. 8 correspondes are known

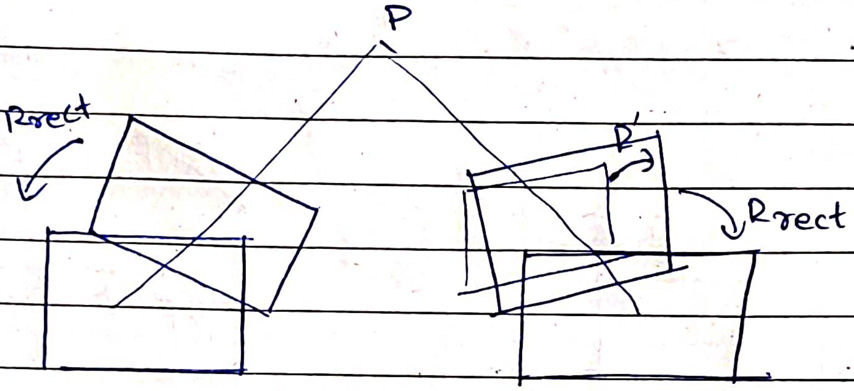
c) In stereo rectification, the two type of homographies involved are Rotation Homography (only rotation transformation b/w the cameras) and Homography on a plane (same plane but rotation + translation both are there)

→ For Rotation Homography we get
$x_1 = H x_2$ where $H = KRK^{-1}$ (derived in (a) part)
using this we get point to point correspondences between 2 images.

During stereo rectification, we rotate the left camera by Rrect and right camera by R' Rrect.

So for all point x_1 in image 1
$x_1' = R_{rect}\, x_1$ — ①
Similarly $x_2' = R' R_{rect}\, x_2$ — ②
Homographies ① and ② are used in stereo rectification

In ②, R' is used to rotate the right camera and then a rectification rotation is performed on the right camera to adjust the epipole to infinity.
Similarly in ①, a rectification is performed and left camera is rotated to adjust the epipole to infinity.



Here the homographies take the epipole to infinity.
Both rotational and standard homographies try provide ~~transformation~~ operation trying to map the two images, which is ~~which~~ what homography ~~tries~~ aims to achieve.

Date:
Page:

Q3 Points in world coordinate are mapped to pixel coordinates as

$$x = PX$$

image coordinate — projection matrix — world coordinate

Projection matrix is dependent on camera intrinsics and extrinsics.

$$P = KR[I_3 \mid -X_0]$$

Given a set of x and X with known correspondences we try to estimate P. (11 intrinsic and extrinsic parameters) using the direct linear transformation.

$$x_{3\times1} = P_{3\times4} X_{4\times1}$$

Since each point gives us 2 equations (x, y coordinate), we need atleast 6 points as DLT as 11 degrees of freedom.

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} = P \begin{bmatrix} U \\ V \\ W \\ T \end{bmatrix} \Rightarrow \begin{bmatrix} U/W \\ V/W \\ 1 \end{bmatrix} = P \begin{bmatrix} U/T \\ V/T \\ W/T \\ 1 \end{bmatrix}$$

So for a correspondence $\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$ we have :

2 equations
$$\begin{cases} x = (P_{11}X + P_{12}Y + P_{13}Z + P_{14}) / (P_{31}X + P_{32}Y + P_{33}Z + P_{34}) \\ y = (P_{21}X + P_{22}Y + P_{23}Z + P_{24}) / (P_{31}X + P_{32}Y + P_{33}Z + P_{34}) \end{cases}$$

Date:
Page:

So we compute P for an uncaliberated camera using DLT if we know $\geq$ 6 points (with known correspondences)

$$x_i = PX_i = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix} X_i$$

Rewrite as

$$x_i = \begin{bmatrix} A^T \\ B^T \\ C^T \end{bmatrix} X_i = \begin{bmatrix} A^T X_i \\ B^T X_i \\ C^T X_i \end{bmatrix}$$

$$x_i = \frac{A^T X_i}{C^T X_i} \qquad y_i = \frac{B^T x_i}{C^T x_i}$$

$\Rightarrow$ $x_i C^T X_i - A^T X_i = 0$ and $y_i C^T x_i - B^T x_i = 0$

$\Rightarrow$ $-X_i^T A + x_i X_i^T C = 0$ and $-X_i^T B + y_i X_i^T C = 0$

We make a new ~~p~~ vector $p = \begin{bmatrix} A \\ B \\ C \end{bmatrix}_{12\times 1}$ and ~~we~~ rewrite the above 2 equations as

$$a_{x_i}^T p = 0$$
$$a_{y_i}^T p = 0$$

where $a_{x_i}^T = \left[-X_i^T, 0^T, x_i X_i^T \right]_{1\times 12}$

$a_{y_i}^T = \left[0^T, -X_i^T, y_i X_i^T \right]_{1\times 12}$

So we stack all points together

$$\begin{bmatrix} a_{x_1}^T \\ a_{y_1}^T \\ a_{x_2}^T \\ a_{y_2}^T \\ \vdots \\ a_{x_I}^T \\ a_{y_I}^T \end{bmatrix} p = M_{2I\times 12}\, p_{12\times 1} = 0 \qquad \text{for } I \text{ points.}$$

Now solving $Mp = 0$ is equivalent to finding null space of M. Thus we apply SVD and choose p as the singular vector belonging to singular value of 0.

However due to noise, ~~with~~ we will have contradictions.

That is $Mp \neq 0$, $Mp = \omega$ (close to 0)

So we try to find a p to minimize ω ($\omega^T\omega$)

$$\begin{aligned} \hat{p} &= \arg\min_p (\omega^T\omega) \\ &= \arg\min_p ((Mp)^T (Mp)) \\ &= \arg\min_p (p^T M^T M p) \end{aligned}$$

with $\|p\|_2 = 1$

We apply SVD on M

$$M_{2I\times 12} = U_{2I\times 12}\, S_{12\times 12}\, V^T_{12\times 12} = \sum_{i=1}^{12} s_i u_i v_i^T$$

Choosing $p = v_{12}$ (singular vector corresponding to smallest singular value s_{12}) minimizes $\omega^T\omega$.

Date:
Page:

We know $U^T U = I$, $V^T V = I$ and $s_1 \geq s_2 \geq \dots \geq s_{12}$

$$
\begin{aligned}
\omega^T \omega &= p^T M^T M p \\
&= p^T (V S U^T)(U S V^T) p \\
&= p^T V S^2 V^T p \\
&= p^T \left(\sum_{i=1}^{12} s_i^2 v_i v_i^T \right) p
\end{aligned}
$$

Due to orthogonality of V, $v_i v_i^T = 1$.
If we choose $p = v_i$

$$\omega^T \omega = v_i^T (s_i^2 v_i v_i^T) v_i = s_i^2$$

So by choosing p as a vector corresponding to smallest singular value, $\omega^T \omega$ simplifies to square of smallest singular value.

So $p = \begin{bmatrix} A \\ B \\ C \end{bmatrix}_{12 \times 1} = v_{12}$.

$$P = \begin{bmatrix} A^T \\ B^T \\ C^T \end{bmatrix}_{3 \times 4}$$

Now to get a solution for p, it is important for M to have a rank 11. However this condition fails if all points lie on a plane. For example, let all points lie of the plane $Z = K$. Then $Z_i = K$ for all points.

$$A = \begin{bmatrix} -x_i & -y_i & -K & -1 & 0 & 0 & 0 & 0 & x_i X_i & x_i Y_i & x_i K & x_i \\ 0 & 0 & 0 & 0 & -x_i & -y_i & -c & -1 & y_i X_i & y_i Y_i & y_i K & y_i \end{bmatrix}$$

Date:
Page:

$C_3 \leftarrow C_3 - C_4(c)$

$C_7 \leftarrow C_7 - C_4(c)$

By this C_7 becomes 0.

Rank (M) < 11 (rank deficiency).

$\therefore$ No sol^n will exist.

Once we have P, we can get K, R, X_0 as

$P = KR[I|-X_0]$

$= [H_\infty | h]$

where $H_\infty = KR$, $h = -KRX_0$

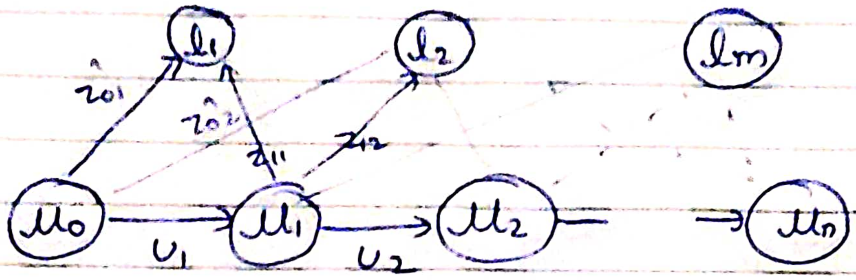
$X_0 = -H_\infty^{-1} h$

and QR decomposition of H_∞^{-1} yeilds R and K

$H_\infty^{-1} = (KR)^{-1} = R^{-1}K^{-1}$

and $K = K/K_{33}$. (due to homogenity)

Q4 a)



$\mu_0, \mu_1 \ldots \mu_n \rightarrow$ state of the robot
where $\mu_i = [\mu_{x_i}\ \mu_{y_i}\ \mu_{\theta_i}]^T$

$u_1, u_2 \ldots u_n \rightarrow$ controls

$l_1, l_2 \ldots l_m \rightarrow$ landmarks with states typically
$l_i = [\mu_{m_ix} ; \mu_{m_iy}]^T$

So $\hat{\mu}_{i+1} = f(\hat{\mu}_i, u_{i+1})$

$\hat{z}_{ik} = h(\hat{\mu}_i, \hat{\mu}_{mk})$

z_{ik} is observation.

$= h(\hat{\mu}_i, \hat{l}_k)$ is the measurment model.

SAM then optmizes the cost function

$$\sum_{i=0}^{n-1} \|f(\hat{\mu}_i, u_{i+1}) - \mu_{i+1}\|_2^2$$
$$+ \sum_{i=1}^{n}\sum_{k=1}^{m} \|\hat{z}_{ik} - z_{ik}\|_2^2$$

Linearize the odometry term as $f(\mu_i^0, u_{i+1}) + F\delta\hat{\mu}_i$ and then the motion model or odometry takes the form

$$\sum_{i=0}^{n} \|f(\mu_i, u_{i+1}) + F\delta\hat{\mu}_i - \mu_{i+1}\|_2^2$$

or $$\sum_{i=0}^{n-1} \|F_i\delta\hat{\mu}_i - a_i\|_2^2$$

where $a_i = f(\mu_i, v_{i+1}) - \mu_{i+1}$ is the odometry error term.

F_i is the 3×3 motion Jacobian

$$\hat{z}_{ik} = h(\mu_i, \mu_{mk})$$

$$\hat{z}_{ik} = h(\mu_i^0, \hat{\mu}_{mk}^0) + \frac{\partial h}{\partial \mu_i} \delta\mu_i \Big|_{\mu_i} + \frac{\partial h}{\partial \mu_{mk}} \delta\mu_{mk} \Big|_{\mu_{mk}^0} - z_{ik}$$

$$\hat{z}_{ik} - z_{ik} = h(\mu_i^0, \mu_{mk}^0) + H_{ik}\, \delta\hat{\mu}_i + J_{ik}\, \delta\hat{\mu}_{mk} - z_{ik}$$

$$= H_{ik}\, \delta\hat{\mu}_i + J_{ik}\, \delta\hat{\mu}_{mk} - c_{ik}$$

$c_{ik} = h(\mu_i^0, \mu_{mk}^0) - z_{ik}$ is the measurement error.

$\mu_i^0, \mu_{mk}^0 \rightarrow$ current estimate of μ_i, μ_{mk} and if it is correct $c_{ik} \rightarrow 0$

Then the measurement term

$\sum_{i=1}^{m}\sum_{k=1}^{n} \|\hat{z}_{ik} - z_{ik}\|_2^2$ takes the form

$$\sum_{i=1}^{m}\sum_{k=1}^{n} \|H_{ik}\, \delta\hat{\mu}_i + J_k\, \delta\hat{\mu}_{mk} - c_k\|_2^2$$

Jacobian :

$$
\begin{bmatrix}
F_1 & 0 & \cdots & & & & 0 \\
H_{11} & 0 & \cdots & 0 & J_{11} & \cdots & 0 \\
H_{12} & 0 & \cdots & 0 & 0 & J_{12} \cdots & 0 \\
\vdots & \vdots & & & & & \\
H_{1M} & 0 & \cdots & 0 & 0 & \cdots & J_{1M} \\
0 & F_2 & \cdots & & & & 0 \\
0 & H_{21} & \cdots & 0 & J_{21} & \cdots & 0 \\
0 & H_{2M} & \cdots & 0 & 0 & \cdots & J_{2M} \\
\vdots & & & & & & \\
0 & & & F_N & 0 & \cdots & 0 \\
\vdots & & & H_{M1} & J_{M1} & \cdots & 0 \\
0 & & & H_{NM} & 0 & \cdots & J_{NM}
\end{bmatrix}
$$

optimization eq^n -

$$
\sum \| f(\hat{\mu}_i , v_{i+1}) - \mu_{i+1} \|^2 + \sum\sum \| H_{ik} \, \delta \hat{\mu}_i + J_k \, \delta \hat{\mu}_{mk} + C_{ik} \|_2^2
$$

b) We are given a series of RGB images across a trajectory and are required to estimate the relative pose between images and map the environment.

This can be done using structure from motion.

We can start by using an algorithm (such as SIFT) to find features that are invariant across many images. After detecting these features we try to map corresponding features across the images. Feature descriptors can be matched to do this (compute nearest neighbours of each descriptor in one image with descriptor in others).

Now if we have α images ~~and~~ ~~if~~, we extract n feature points in ~~them~~ each image.

Now since we only have the images we don't know the projection matrices. Each image will have a different Projection matrix. Let P_j be projection matrix for j^{th} image. For i^{th} world point, the image of this point in j^{th} image will be

$$x_{ij} = P_j X_i$$

Now we have to get $P_1, P_2 \dots P_\alpha$, $X_1, X_2 \dots X_n$ such that

$$\underset{P_i,\, X_j}{\operatorname{argmin}} \sum_{i=1}^{\alpha} \sum_{j=1}^{n} \left(x_{ij} - P_j X_i \right)^2$$

(we can separately add points that are not visible in all, but visible in multiple images separately. to the above summation)

Now break x_{ij} into (x_{ij}, y_{ij}) – 2 image coordinates.

$\cancel{X_i = \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix}}$ to adjust for homogenous, we ~~divide by 3rd row (as done in Q3)~~

So we get

$$\underset{P_i, X_j}{\operatorname{argmin}} \sum_i \sum_j \left(\left[\frac{P_{i11} X_j + P_{i12} Y_j + P_{i13} Z_j + P_{i14}}{P_{i31} X_j + P_{i32} Y_j + P_{i33} Z_j + P_{i34}} - x_{ij} \right]^2 + \left[\text{same for } y_{ij} \right]^2 \right)$$

This is reprojection error as we are estimating error between point in image and the projection.

We can solve this ~~by~~ (min the error) by method of least squares. we will get the Jacobian J by differentiating this cost w.r.t P_i and X_js and after that we can update ~~our~~ estimates using update rule

$$\text{update} = -\underbrace{(J^T J)^{-1} J^T r}_{}$$

r – residual.