Projet : Visual odométry

→ formalisat° mathématiq   accumulative drift?

2 equat° $\begin{cases} \text{motion: } x_k = f(x_{k-1}, u_k, w_k) \\ \text{observat°: } z_{k,j} = h(y_j, x_k, v_{k,j}) \end{cases}$

x_k posit° at k
u_k command at k
w_k noise at k
z_k observat° at k
y_j j^{th} landmark
$v_{k,j}$ visual noise

→ Bundle adjustem^t : where we are, where are the points on camera

FOR 1 CAMERA & 1 point

$\tilde{m} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}$ coord de l'objet 3D projetés en 2D tq mesuré

$P = K[R\ T]$
$= KR[I\ -C]$

P roject°
K allbrat°
R otat°
T ranslat°

Camera center in observer

→ Applicat° of least squares method to estimate R and C

→ Then use them to project $\tilde{m}$ into 3D world as X

$X = \begin{pmatrix} x_x \\ y_x \\ z_x \end{pmatrix}$ and use the values of X and project them on the camera, crating $m = \begin{pmatrix} x \\ y \end{pmatrix}$ due to noise

→ Least Square Method has an error: this creates the reproject° error

$e = \tilde{m} - m$ → erreur dans les coord de l'image

→ challenge of minimizing the error

$$e = \tilde{m} - m = \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} - \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} - \begin{pmatrix} u/w \\ v/w \end{pmatrix}$$ avec u, v, w tq

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = P\begin{pmatrix} X \\ 1 \end{pmatrix} = KR(I - C)\begin{pmatrix} X \\ 1 \end{pmatrix} = KR(X - C)$$

$$\underset{\substack{R, C, X \\ \text{/quaternion}}}{\text{minimize}} \left\| \underbrace{\begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix}}_{b} - \underbrace{\begin{pmatrix} u(R, C, X) / w(R, C, X) \\ v(R, C, X) / w(R, C, X) \end{pmatrix}}_{f(R, C, X)} \right\|^2$$

(with R → $R(q)$)

$$= \underset{q, C, X}{\text{minimize}} \underbrace{\| b - f(R(q), C, X) \|^2}_{\text{cost function}}$$

→ Calculat° of Jacobian matrix (necessary for minimizat°)

$$J \Rightarrow \begin{cases} n \text{ rows} \Rightarrow n \text{ constraints} \\ m \text{ columns} \Rightarrow m \text{ variables} \end{cases}$$

coordinates on the img point

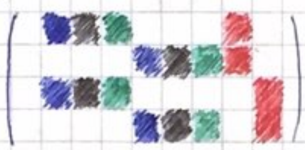
$$J = \left(\underbrace{\frac{\partial f(R(q),C,X)}{\partial q}}_{\text{4 quaternions}} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial C}}_{\text{3 angles of rotat°}} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial X}}_{\text{3 3D space points}} \right) \Big\} \, 2 \; x,y$$

$$J = \left(\underbrace{\frac{\partial f(R(q),C,X)}{\partial R}}_{2\times 9} \underbrace{\frac{\partial R}{\partial q}}_{9\times 4} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial C}}_{2\times 3} \quad \underbrace{\frac{\partial f(R(q),C,X)}{\partial X}}_{2\times 3} \right)$$

$\rightarrow 2\times 4$

J will give us the ΔX, ΔC and ΔR we need to adjust to minimize the cost fctn

FOR Many CAMERAS & MANY POINTS

$J =$ 

On a donc : $J^T J \, \Delta X = J^T (b - f(x))$

$\iff \Delta X = \underbrace{(J^T J)^{-1}}_{*} J^T (b - f(x))$ } fctn à résoudre

* voir décomposit° de la matrice dans Robotics -4.4.14- Multi view geometry sur YT