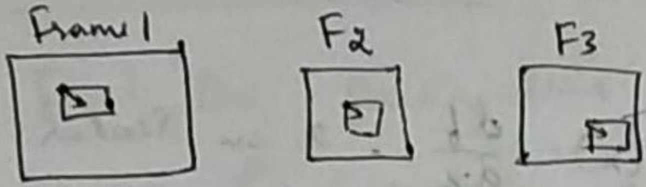
KLT tracking (Kanade, Lucas, Tomasi)



→ Optical flow

Simple KTL Algorithm.

1. Detect Harris corners in 1st frame.
2. For each Harris corner point, compute motion (translation or affine) between consecutive frames.
   local — global
3. Link motion vectors in successive frames to get a track of each Harris point.
4. Introduce new Harris points by applying Harris detector at every m (10-15) frames.
5. Track new & old points using steps 1-3.

Alignment

→ Given two (local) patches between 2 frames, need to find transformation between 2 patches.

→ patch can be around Harris point or around object.

2D Affine has 6 DOF. $\begin{bmatrix} a_1 & a_2 & T_1 \\ a_3 & a_4 & T_2 \\ 0 & 0 & 1 \end{bmatrix}$

Displacement of model parameters. (need for alignment)

Translation: $x' = x + b_1$, $y' = y + b_2$ — This transf is represented as function (warping function)

$$W(X; P) = (x + b_1,\ y + b_2)$$

X, P: Two vectors. This function gives two values.

Affine: $$W(X; P) = (a_1 x + a_2 y + b_1,\ a_3 x + a_4 y + b_2)$$

→ This function is required to estimate transformation.

→ **Derivative & Gradient.**

$f(x)$, derivative $f'(x) = \frac{df}{dx}$, x is scalar.

Function: $f(x_1, x_2 \dots x_n)$

Gradient: $\nabla f(x_1 \dots x_n) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2} \dots \frac{\partial f}{\partial x_n}\right)$

→ **Jacobian.**

→ How to you find derivative of vector valued function, $W(X; P)$.

eg:- $F(x_1, x_2 \dots x_n) = (f_1(x_1 \dots x_n), f_2(x_1 \dots x_n) \dots f_m(x_1 \dots x_n))$

If we find derivative for the above, it will be matrix.

$$J(F) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \end{bmatrix}$$

→ **Displacement Model Jacobians.**

$$W(X; P) = (x + b_1, y + b_2)$$

$$\frac{\partial W}{\partial P} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Rigid $W(X; P) = (x\cos\theta - y\sin\theta + b_1, x\sin\theta + y\cos\theta + b_2)$

$$\frac{\partial W}{\partial P} = \begin{bmatrix} 1 & 0 & -x\sin\theta - y\cos\theta \\ 0 & 1 & x\cos\theta - y\sin\theta \end{bmatrix}$$

Why we can find Affine.

Finding Alignment

→ Find the parameters

$$\sum_x \left[I(W(x;P)) - T(x)\right]^2$$

we need to transform I, to that we need to know W, which inturn means we need to know parameter P.

→ Assume initial estimate of P is known, ΔP.

here, ΔP is change in P.

$$\sum_x \left[\underbrace{I(W(x;P+\Delta P)}_{} - T(x)\right]^2$$

if it is zero, its best alignment

Sum of squared difference

→ This is the function need to be minimized.

Find a Taylor series (to find approximation to this function)

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots$$

$$= \sum_x \left[I(W(x;P) + \nabla I \frac{\partial W}{\partial P}(P+\Delta P - P) - T(x)\right]^2$$

here $f(a) = I(W(x;P))$

$f(x) = I(W(x;P+\Delta P))$.

$$= \sum_x \left[I(W(x;P) + \nabla I \frac{\partial W}{\partial P}\Delta P - T(x)\right]^2$$

W is a vector valued function

general for any transformation (Jacobian)

It can be Translation/Affine/Perspective/Rigid

$$\nabla I = \begin{bmatrix} I_x \\ I_y \end{bmatrix}$$

gradient

→ Now, we need to find out ΔP such that this function is minimized.

→ To do this we differentiate wrt ΔP & equate to zero

$$2\sum_{x}\left[\nabla I\frac{\partial W}{\partial p}\right]^T\left[I(W(x;p)+\nabla I\frac{\partial W}{\partial p}\Delta p-T(x)\right]=0$$

$\frac{d}{dx}(x)^2$

$2x$.

$ax \to a$

find Δp.

$$\boxed{\Delta p = H^{-1}\,2\sum_{x}\left[\nabla I\frac{\partial W}{\partial p}\right]^T\left[T(x)-I(W(x;p))\right]}$$

$$H = \underbrace{\sum_{x}\left[\nabla I\frac{\partial W}{\partial p}\right]^T\left[\nabla I\frac{\partial W}{\partial p}\right]}_{\text{whole thing is } 2\times 2 \text{ matrix Hessian}}$$

∇I → matrix, $\frac{\partial W}{\partial p}$ → Jacobian matrix

If we consider Translation, then Jacobian matrix is $\frac{\partial W}{\partial p}=\begin{bmatrix}1&0\\0&1\end{bmatrix}$

$$\therefore H = \underbrace{\sum_{x}\begin{bmatrix}I_x^2 & I_xI_y\\ I_xI_y & I_y^2\end{bmatrix}}_{\text{Harris Corner}} \Rightarrow \begin{bmatrix}\left(\frac{\partial I}{\partial x}\right)^2 & \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\\ \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} & \left(\frac{\partial I}{\partial y}\right)^2\end{bmatrix}$$

→ In algorithm to find alignment b/w two frames, then change only Jacobian, then it can be Trasl, Rigid, affine etc..

→ To do this alignment we need 9 steps,

1. Warp I with $W(x;p)$
2. Subtract I from T. $[T(x)-I(W(x;p))]$
3. Compute gradient ∇I

(4) Evaluate the Jacobian. $\frac{\partial W}{\partial p}$.

(5) compute steepest descent $\nabla I \frac{\partial W}{\partial p}$.

(6) Compute Inverse Hessian H^{-1}.

(7) Multiply steepest descent with error $\sum_{x} \left[\nabla I \frac{\partial W}{\partial p}\right]^T [T(x) - W(x;$

(8) Compute Δp.

(9) Update parameter $p \to p + \Delta p$.

(Baker et-al), 2004

has same steps but slightly changed it.

$$\Delta p = H^{-1} \sum_{x} \left[\nabla T \frac{\partial W}{\partial p}\right]^T \left[I(W(x;p)) - T(x)\right]$$

→ Now operation on T will be performed only once. instead of every time (iteration)