let $(\omega(x,y);p) = [\omega_x, \omega_y]$

(3) Lucas-Kanade Algorithm for motion tracking when motion is affine

$u(x,y) = a_1 * x + b_1 * y + c_1$ $\quad$ $v(x,y) = a_2 * x + b_2 * y + c_2$

$= a_1 x + b_1 y + c_1$ $\quad$ $= a_2 x + b_2 y + c_2$

Lucas-Kanade algorithm is basically used to compute the optic flow
u,v are location of the template in current frame

Assuming there is constant flow for all the pixels in the image is not reasonable considering the long periods of time.
We can use the approach for 2D motion models like affine by using "warp" function (ω)

Initially,

$$E(u,v) = \Sigma[I(x+u, y+v) - T(x,y)]^2$$

$\downarrow$ Generalize for affine motion using warp

$$\Sigma [I(\omega[x,y];P) - T(x,y)]^2$$

Procedure:-

Step 1: Obtain warp-I : $I(\omega(x,y);P)$
Jacobian of affine warp
let $(\omega(x,y);p) = [\omega_x, \omega_y]$

$$\frac{\partial \omega}{\partial p} = \begin{bmatrix} \frac{\partial \omega_x}{\partial p_1} & \frac{\partial \omega_x}{\partial p_2} & \frac{\partial \omega_x}{\partial p_3} & \cdots & \frac{\partial \omega_x}{\partial p_n} \\ \frac{\partial \omega_y}{\partial p_1} & \frac{\partial \omega_y}{\partial p_2} & \frac{\partial \omega_y}{\partial p_3} & \cdots & \frac{\partial \omega_y}{\partial p_n} \end{bmatrix}$$

Functions has 6 parameters $(P_1, P_2, P_3, P_4, P_5, P_6)$

$$w([x,y]^T; p) = \begin{pmatrix} 1+P_1 & P_3 & P_5 \\ P_2 & 1+P_4 & P_6 \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} ; \quad \frac{\partial \begin{bmatrix} x + xP_1 + yP_3 + P_5 \\ xP_2 + y + yP_4 + P_6 \end{bmatrix}}{\partial p}$$

$$= \begin{bmatrix} x & 0 & y & 0 & 1 & 0 \\ 0 & x & 0 & y & 0 & 1 \end{bmatrix}$$

step 2: Compute the error image

$$T(x) - I(w[x,y]; p)$$

Step 3: Warp the gradient ∇I with $w([x,y]; p)$

step 4: Calculate the Jacobian as said in step 1

evaluate $\frac{\partial w}{\partial p}$ at $([x,y]; p)$

steps: Compute the steepest descent Images

$$\nabla I \cdot \frac{\partial w}{\partial p}$$

step 6: Compute the Hessain matrix

$$\sum \left(\nabla I \frac{\partial w}{\partial p}\right)^T \left(\nabla I \frac{\partial w}{\partial p}\right)$$

step 7: Calculate

$$\sum \left(\nabla I \frac{\partial w}{\partial p}\right)^T \left(T(x,y) - I(w[x,y]; p)\right)$$

Step 8: Compute the magnitude of P (ΔP)

step 9: Update the p until Δp magnitude is negligible

$$\underset{\text{update}}{p} \leftarrow p + \Delta p$$

The above is the procedure/algorithm to peform for motion tracking when the motion is affine

Derivation: Using the Taylor series.

$$I(W([x,y]; P+\Delta P) \approx I(W[x,y]; P) + \nabla I \frac{\partial W}{\partial P} \Delta P$$

Using the chain Rule,

$$z = f(x,y) \quad x = g(t) \quad y = h(t)$$
$$= f(g(t), h(t))$$

Then
$$\frac{\partial z}{\partial t} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial T} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial T}$$

$$z(T + \Delta T) = z(T) + \frac{\partial z}{\partial T}\Big|_{\tilde{T}} \left(\Delta T + \text{higher order term}\right)$$

$$I(x; y; \tilde{P} + \Delta P) \approx I(x, y; \tilde{P}) + \frac{\partial I}{\partial P}\Big)_{\tilde{P}} \Delta P.$$

$$= I(x, y; \tilde{P}) + \underbrace{\left[\frac{\partial I_x}{\partial x}\frac{\partial W_x P}{\partial P} + \frac{\partial I}{\partial y}\frac{\partial W_y(P)}{\partial P}\right]}_{\text{Chain Rule}} \Delta P$$

Now Expression can be written as.

$$I(x, y; \tilde{P}_1 \cdots \tilde{P}_n) + \left[\frac{\partial I}{\partial x} \; \frac{\partial I}{\partial y}\right] \begin{pmatrix} \frac{\partial W_x}{\partial P_1} \\ \frac{\partial W_y}{\partial P_1} \end{pmatrix}_{\tilde{P}_1} \Delta P_1 +$$

$$\left[\frac{\partial I}{\partial x} \; \frac{\partial I}{\partial y}\right] \begin{pmatrix} \frac{\partial W_x}{\partial P_2} \\ \frac{\partial W_y}{\partial P_2} \end{pmatrix} \Delta P_2 + \cdots$$

3 <u>Plane and orthographic</u> :-

$$u = c_1 + a_1 x + b_1 y$$
$$v = c_2 + a_2 x + b_2 y$$

consider a planar Z

$$Z = aX + bX + cY$$

$$u = V_1 + \Omega_2 Z - \Omega_3 y$$
$$v = V_2 + \Omega_2 Z - \Omega_1 y$$

$u = Ax + b$ where $b_1 = V_1 + a\Omega_2$

$a_1 = b\Omega_2$

$a_2 = -c\Omega_2 - \Omega_3$

and

$b_2 = V_2 - a\Omega_1$

$a_3 = \Omega_3 - b\Omega_1$

$a_4 = -c\Omega_1$

The optical flow is affine for x and y in the displacement model $\bar{x}$ and $\bar{y}$ is affine of x and y.

Scanned with CamScanner