The KLT
(Kanade-Lucas-Tomasi)
tracker

KLT Introduction

Given:

- Input image I and the template image T
- Region of interest (ROI) in T
- Assumed transformation W from ROI in T to (part of) I :

$$W(\mathbf{x};\mathbf{p}) : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \quad (\mathbf{p} = \text{parameter})$$

Find:

- $\mathbf{p}^*$ such that $T(\mathbf{x})$ is "close" to $I(W(\mathbf{x};\mathbf{p}^*))$ for $\mathbf{x} \in \text{ROI}$

Template image T

$I(W(\mathbf{x};\mathbf{p}^*))$

Input image I

KLT, Transformation functions

Examples of transformation $W(\mathbf{x};\mathbf{p}): \mathbb{R}^2 \rightarrow \mathbb{R}^2$ ($\mathbf{p}$ = parameter)

Template image $T(\mathbf{x})$, $\mathbf{x} = (x_1, x_2)^T$,

Input image $I(\mathbf{y})$, $\mathbf{y} = (y_1, y_2)^T$.

- Translation by $\mathbf{p} = (p_1, p_2)^T$, $\mathbf{p} \in \mathbb{R}^2$:

$$W(\mathbf{x};\mathbf{p}) = \mathbf{x} + \mathbf{p} = \begin{pmatrix} x_1 + p_1 \\ x_2 + p_2 \end{pmatrix}$$

- Affine transformation, $\mathbf{p} \in \mathbb{R}^{2\times 3}$:

$$W(\mathbf{x};\mathbf{p}) = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

- Rotation by p around origin, $\mathbf{p} = p \in [0, 2\pi)$:

$$W(\mathbf{x};\mathbf{p}) = \begin{bmatrix} \cos p & \sin p \\ -\sin p & \cos p \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

KLT, Measuring Fit

Measure of "closeness" of $T(\mathbf{x})$ and $I(W(\mathbf{x};\mathbf{p}))$, $\mathbf{x} \in \mathrm{ROI}$

SSD (used most often):

$$SSD(\mathbf{p}) = \sum_{\mathbf{x} \in \mathrm{ROI}} \left[I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x}) \right]^2$$

Optimization task: Find $\mathbf{p*}$ such that

$$\mathbf{p}^* = \operatorname*{argmin}_{\mathbf{p}} SSD(\mathbf{p})$$

Note: In the subsequent text, for better clarity we use the following notation for 'function at a point' interchangeably:

$$f(x) \equiv f|_x, \text{ e.g. } I(W(\mathbf{x};\mathbf{p})) \equiv I|_{W(\mathbf{x};\mathbf{p})}$$

KLT, Finding Best Transformation $\mathbf{p}^*$

$$SSD(\mathbf{p}) = \sum_{\mathbf{x} \in \text{ROI}} [I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2 \qquad \mathbf{p}^* = \underset{\mathbf{p}}{\operatorname{argmin}}\, SSD(\mathbf{p})$$

How to find $\mathbf{p}^*$ ***?***

It would be possible to quantize the search space of $\mathbf{p}$, enumerate all possibilities and take the one minimizing SSD. But this computationally expensive and would be infeasible in practice.

KLT approach:

1. Make the linearization (first-order Taylor approximation) inside brackets w.r.t. the parameter $\mathbf{p}$
2. By this, the problem is converted to least-squares problem which is well understood and has a closed-form solution.
3. Because of approximation in step 1, iterate.

Approximating the warp

$$SSD(\mathbf{p}) = \sum_{\mathbf{x} \in \text{ROI}} [I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2 \qquad \mathbf{p}^* = \underset{\mathbf{p}}{\operatorname{argmin}}\, SSD(\mathbf{p})$$

Taylor approximation of $I(W(\mathbf{x};\mathbf{p}))$:

$$I(W(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p})) \approx \underbrace{I\big|_{W(\mathbf{x};\mathbf{p}_0)}}_{\text{im. for current } \mathbf{p}_0} + \underbrace{\nabla I^T\big|_{W(\mathbf{x},\mathbf{p}_0)}}_{\text{image gradient}} \underbrace{\frac{\partial W}{\partial \mathbf{p}}\bigg|_{\mathbf{x},\mathbf{p}_0}}_{\text{Jacobian of } W} \Delta\mathbf{p}$$

im. for current $\mathbf{p}_0$

image gradient

Jacobian of W

$$\frac{\partial W}{\partial \mathbf{p}} = \begin{bmatrix} \frac{\partial W_1}{\partial p_1} & \frac{\partial W_1}{\partial p_2} & \cdots & \frac{\partial W_1}{\partial p_n} \\ \frac{\partial W_2}{\partial p_1} & \frac{\partial W_2}{\partial p_2} & \cdots & \frac{\partial W_1}{\partial p_n} \end{bmatrix}$$

Example, pure translation

Transformation: $$W(\mathbf{x};\mathbf{p}) = \begin{pmatrix} x_1 + p_1 \\ x_2 + p_2 \end{pmatrix}$$

Its Jacobian: $$\frac{\partial W}{\partial \mathbf{p}}(\mathbf{x};\mathbf{p}) = \begin{bmatrix} \frac{\partial W_1}{\partial x_1} & \frac{\partial W_1}{\partial x_2} \\ \frac{\partial W_2}{\partial x_1} & \frac{\partial W_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$I(W(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p})) \approx I|_{W(\mathbf{x};\mathbf{p}_0)} + \underbrace{\nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \frac{\partial W}{\partial \mathbf{p}}}\Big|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p}$$

$$\nabla I^T \frac{\partial W}{\partial \mathbf{p}} = \nabla I^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \nabla I^T = \left(\frac{\partial I}{\partial x_1}, \frac{\partial I}{\partial x_2}\right)$$

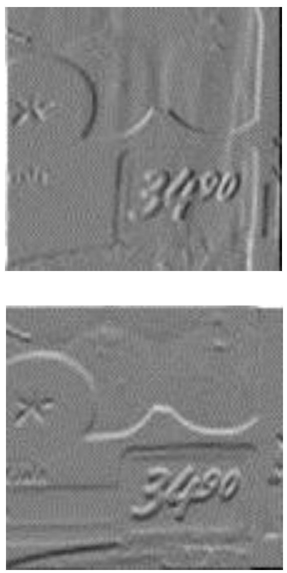
$$I(W(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p})) \approx I|_{W(\mathbf{x};\mathbf{p}_0)} + \Delta p_1 \frac{\partial I}{\partial x_1}\Big|_{W(\mathbf{x},\mathbf{p}_0)} + \Delta p_2 \frac{\partial I}{\partial x_2}\Big|_{W(\mathbf{x},\mathbf{p}_0)}$$

Example, pure translation

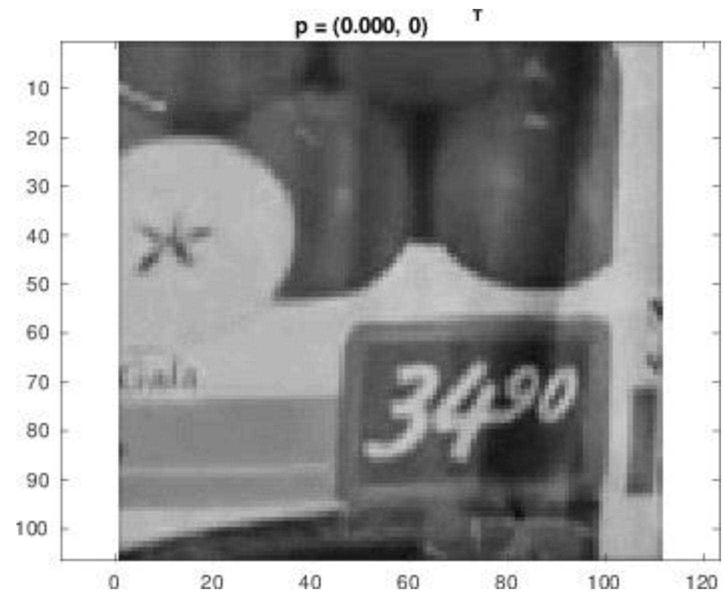
$$I(W(\mathbf{x};\mathbf{p}_0+\Delta\mathbf{p})) \approx I|_{W(\mathbf{x};\mathbf{p}_0)} + \Delta p_1 \left.\frac{\partial I}{\partial x_1}\right|_{W(\mathbf{x},\mathbf{p}_0)} + \Delta p_2 \left.\frac{\partial I}{\partial x_2}\right|_{W(\mathbf{x},\mathbf{p}_0)}$$

image

gradient



approximation



LSQ problem to find $\Delta \mathbf{p}^*$

$$SSD(\mathbf{p}) = \sum_{\mathbf{x} \in ROI} [I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2 \qquad \mathbf{p}^* = \underset{\mathbf{p}}{\operatorname{argmin}}\, SSD(\mathbf{p})$$

Taylor approximation of $I(W(\mathbf{x};\mathbf{p}))$:

$$I(W(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p})) \approx I|_{W(\mathbf{x};\mathbf{p}_0)} + \nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \left.\frac{\partial W}{\partial \mathbf{p}}\right|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p}$$

Solve for $\Delta \mathbf{p}^*$ **:**

$$\Delta\mathbf{p}^* = \underset{\Delta\mathbf{p}}{\operatorname{argmin}} \sum_{\mathbf{x} \in ROI} \left[I|_{W(\mathbf{x};\mathbf{p}_0)} + \nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \left.\frac{\partial W}{\partial \mathbf{p}}\right|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p} - T(\mathbf{x}) \right]^2$$

KLT, Finding Best Transformation $\mathbf{p}^*$

This:

$$\Delta\mathbf{p}^* = \underset{\Delta\mathbf{p}}{\operatorname{argmin}} \sum_{\mathbf{x}\in\mathrm{ROI}} \left[I|_{W(\mathbf{x};\mathbf{p}_0)} + \nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \frac{\partial W}{\partial \mathbf{p}}\bigg|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p} - T(\mathbf{x}) \right]^2$$

has a form of a classical least-squares problem:

$$\underset{\mathbf{z}}{\operatorname{argmin}} \|\mathbf{A}\mathbf{z} - \mathbf{b}\|^2$$

where

- rows of matrix $\mathbf{A}$ are equal to $\nabla I^T(\partial W/\partial \mathbf{p})$ (for a given $\mathbf{x}$ and $\mathbf{p}_0$),
- $\mathbf{z} = \Delta\mathbf{p}$, and
- elements of $\mathbf{b}$ equal to $T(\mathbf{x}) - I(W(\mathbf{x};\mathbf{p}_0))$, that is, current residuals for given $\mathbf{x}$.

Such problem has a well-known solution, $\mathbf{z} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$.

Example, pure translation

Transformation: $W(\mathbf{x};\mathbf{p}) = \begin{pmatrix} x_1 + p_1 \\ x_2 + p_2 \end{pmatrix}$

Its Jacobian:

$$\frac{\partial W}{\partial \mathbf{p}}(\mathbf{x};\mathbf{p}) = \begin{bmatrix} \frac{\partial W_1}{\partial x_1} & \frac{\partial W_1}{\partial x_2} \\ \frac{\partial W_2}{\partial x_1} & \frac{\partial W_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Solution for update (from previous slide):

$$\mathbf{z} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$$

Rows of **A**: $\nabla I^T \frac{\partial W}{\partial \mathbf{p}} = \nabla I^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \nabla I^T = \left(\frac{\partial I}{\partial x_1}, \frac{\partial I}{\partial x_2}\right)$

$$\mathbf{A}^T\mathbf{A} = \sum_{\mathbf{x}\in \mathrm{ROI}} \begin{bmatrix} \left(\frac{\partial I}{\partial x_1}\right)^2, & \frac{\partial I}{\partial x_1}\frac{\partial I}{\partial x_2} \\ \frac{\partial I}{\partial x_1}\frac{\partial I}{\partial x_2} & \left(\frac{\partial I}{\partial x_2}\right)^2, \end{bmatrix}$$

Stability of computation will rely on rank of this matrix

$$\mathbf{A}^T\mathbf{b} = \sum_{\mathbf{x}\in \mathrm{ROI}} [T(\mathbf{x}) - I(W(\mathbf{x};\mathbf{p}_0))] \left(\frac{\partial I}{\partial x_1}, \frac{\partial I}{\partial x_2}\right)^T$$

Example, pure translation

$$\mathbf{A}^T\mathbf{A} = \sum_{\mathbf{x}\in ROI} \begin{bmatrix} \left(\frac{\partial I}{\partial x_1}\right)^2, & \frac{\partial I}{\partial x_1}\frac{\partial I}{\partial x_2} \\ \frac{\partial I}{\partial x_1}\frac{\partial I}{\partial x_2} & \left(\frac{\partial I}{\partial x_2}\right)^2, \end{bmatrix}$$

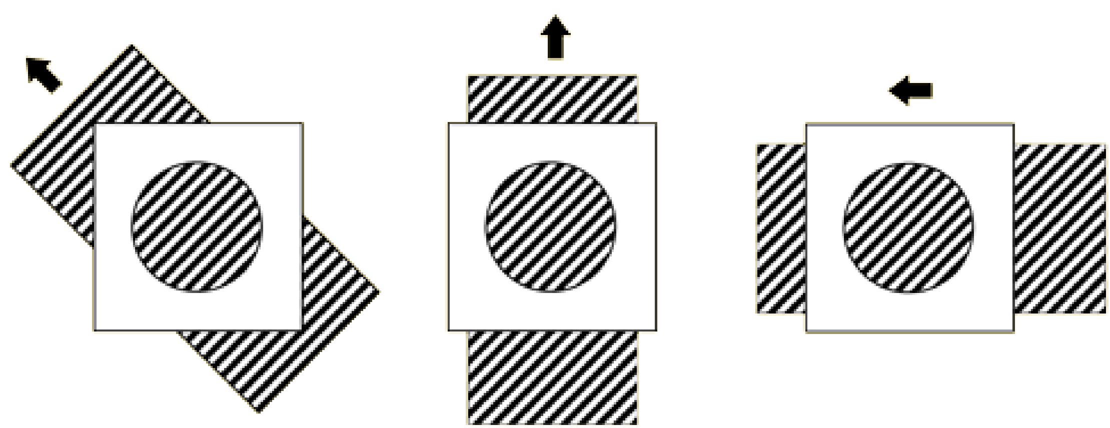
The stability of computation will depend on invertibility of this matrix.

We have already encountered this matrix in this course.

It is directly related to Harris corner detector.

For obvious reasons, the KLT tracking for translation is well posed when the ROI contains a corner, as opposed to a flat region or an edge.

Aperture problem:



(image courtesy of Wolfe et al. Sensation & Perception)

KLT, algorithm summary

$$SSD(\mathbf{p}) = \sum_{\mathbf{x} \in \mathrm{ROI}} [I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2 \qquad \mathbf{p}^* = \operatorname*{argmin}_{\mathbf{p}} SSD(\mathbf{p})$$

Taylor approximation of $I(W(\mathbf{x};\mathbf{p}))$:

$$I(W(\mathbf{x};\mathbf{p}_0 + \Delta\mathbf{p})) \approx I|_{W(\mathbf{x};\mathbf{p}_0)} + \nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \left.\frac{\partial W}{\partial \mathbf{p}}\right|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p}$$

1. Input: current estimate $\mathbf{p}_0$. Solve for $\Delta\mathbf{p}^*$:

$$\Delta\mathbf{p}^* = \operatorname*{argmin}_{\Delta\mathbf{p}} \sum_{\mathbf{x} \in \mathrm{ROI}} \left[I|_{W(\mathbf{x};\mathbf{p}_0)} + \nabla I^T|_{W(\mathbf{x},\mathbf{p}_0)} \left.\frac{\partial W}{\partial \mathbf{p}}\right|_{\mathbf{x},\mathbf{p}_0} \Delta\mathbf{p} - T(\mathbf{x}) \right]^2$$

2. Update, $\mathbf{p}_0 \leftarrow \mathbf{p}_0 + \Delta\mathbf{p}^*$
3. If not converged, go to 1.
4. Best transformation $\mathbf{p}^* = \mathbf{p}_0$.

Multi-resolution Lucas-Kanade

KLT is based on 1st order Taylor approximation and cannot find the solution if the displacement is too high compared to frequencies in the image.

Possible solution is to blur the image on the scale of assumed displacement and get a rough estimate.

Subsequently, continue on less blurred image and make an update to that estimate. Repeat.

This is efficiently implemented using image pyramids.

slide credit:
Patrick Perez