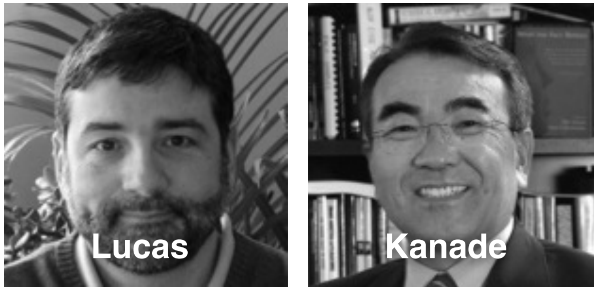
ENEE 731: KLT tracking

Ching-Hui Chen (ching@umd.edu)

Credits: Materials are collected from slides in
http://www.cs.cmu.edu/~16385/lectures/Lecture23.pdf
http://crcv.ucf.edu/courses/CAP5415/Fall2013/Lecture-10-KLT.pdf

Kanade-Lucas-Tomasi (KLT) Tracker



An Iterative Image Registration Technique with an Application to Stereo Vision. (1981)

Detection and Tracking of Feature Points. (1991)

The original KLT algorithm

Good Features to Track. (1994)

Slide from: http://www.cs.cmu.edu/~16385/lectures/Lecture23.pdf

KLT papers

Lucas – Kanade: Method for aligning an image patch (track from frame to frame)

Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision (1981).

Tomasi – Kanade: Method for choosing the best feature (image patch) for tracking (The original KLT)

Carlo Tomasi and Takeo Kanade. Detection and Tracking of Point Features (1991).

Affine variation of KLT:

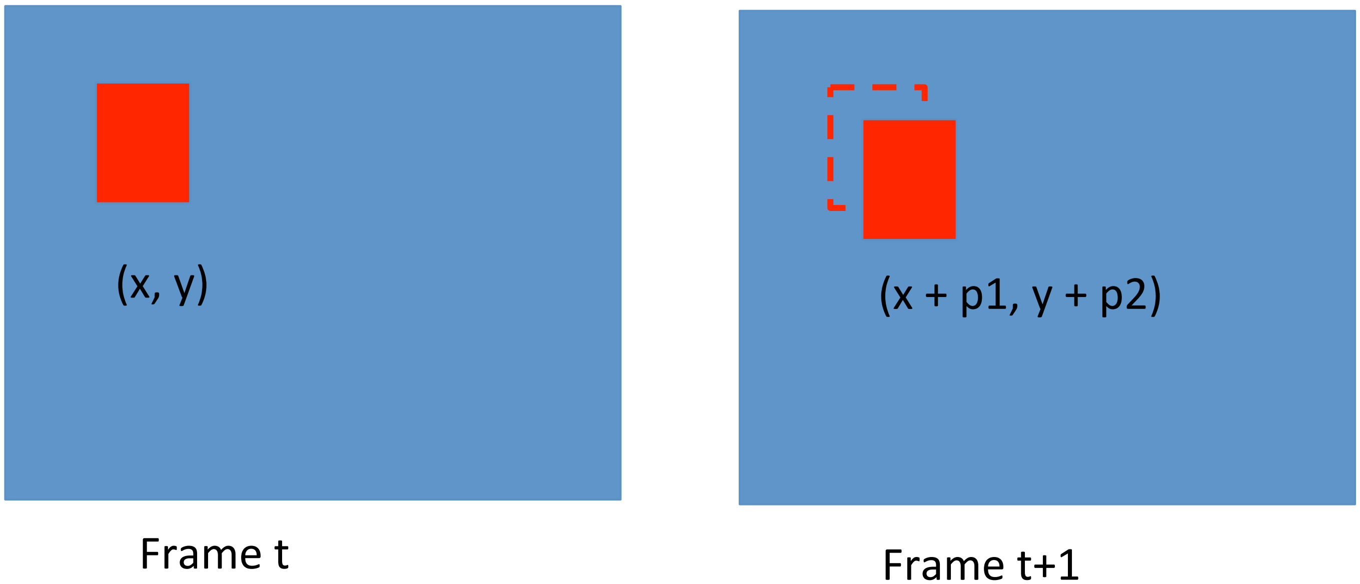
Jianbo Shi and Carlo Tomasi. Good Features to Track (1994).

A good summary of KLT:

Simon Baker and Iain Mattews. Lucas-Kanade 20 Years On: A Unifying Framework (2004).

Ref: http://www.cs.cmu.edu/~16385/lectures/Lecture23.pdf

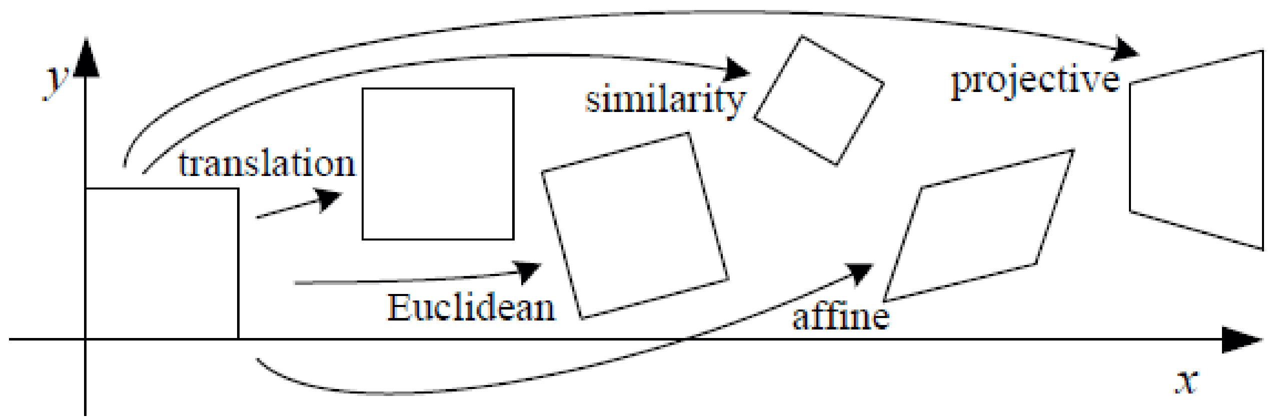
Warping Function of Translation



Translation:

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

Basic Set of 2-D Transformation



Richard Szeliski, "[Computer Vision: Algorithms and Application]".

Summary of Displacement Models (2-D Transformations)

Translation

$$x' = x + b_1$$
$$y' = y + b_2$$

Rigid

$$x' = x\cos\theta - y\sin\theta + b_1$$
$$y' = x\sin\theta + y\cos\theta + b_2$$

Affine

$$x' = a_1 x + a_2 y + b_1$$
$$y' = a_3 x + a_4 y + b_2$$

Projective

$$x' = \frac{a_1 x + a_2 y + b_1}{c_1 x + c_2 y + 1}$$
$$y' = \frac{a_3 x + a_4 y + b_2}{c_1 x + c_2 y + 1}$$

Bi-quadratic

$$x' = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 y^2 + a_6 xy$$
$$y' = a_7 + a_8 x + a_9 y + a_{10} x^2 + a_{11} y^2 a_{12} xy$$

Bi-Linear

$$x' = a_1 + a_2 x + a_3 y + a_4 xy$$
$$y' = a_5 + a_6 x + a_7 y + a_8 xy$$

Pseudo-Perspective

$$x' = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy$$
$$y' = a_6 + a_7 x + a_8 y + a_4 xy + a_5 y^2$$

Warp function W(x; p)

$$W(x;p) = \begin{bmatrix} x + b_1 \\ y + b_2 \end{bmatrix}$$

$$W(x;p) = \begin{bmatrix} 1 & 0 & b_1 \\ 0 & 1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$W(x;p) = [R \mid t]_{2X3} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$W(x;p) = [sR \mid t]_{2X3} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$W(x;p) = [A]_{2X3} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$W(x;p) = [H]_{3X3} \begin{bmatrix} x \\ y \end{bmatrix}$$

Finding Alignment

Find $\mathbf{p}$ s.t. following is minimized

$$\sum_{\mathbf{x}} [I(W(\mathbf{x};\mathbf{p})) - T(\mathbf{x})]^2$$

Assume initial estimate of $\mathbf{p}$ is known, find $\Delta\mathbf{p}$

$$\sum_{\mathbf{x}} [\boxed{I(W(\mathbf{x};\mathbf{p}+\Delta\mathbf{p}))} - T(\mathbf{x})]^2$$

Find Taylor Series

$$\sum_{\mathbf{x}} [I(W(\mathbf{x};\mathbf{p})) + \nabla I \frac{\partial W}{\partial \mathbf{p}} \Delta\mathbf{p} - T(\mathbf{x})]^2 \qquad \nabla I = \begin{bmatrix} I_x & I_y \end{bmatrix}$$

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

Differentiate wrt $\Delta\mathbf{p}$ and equate it to zero

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

And equate it to zero to find

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

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

$$H = \sum_{\mathbf{x}} \left[\nabla I \frac{\partial W}{\partial p}\right]^T \left[\nabla I \frac{\partial W}{\partial p}\right]$$

Lucas-Kanade Method

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

1. Warp I with $W(\mathbf{x};\mathbf{p})$
2. Subtract I from T $[T(\mathbf{x}) - I(W(\mathbf{x};\mathbf{p}))]$
3. Compute gradient ∇I
4. Evaluate the Jacobian $\frac{\partial W}{\partial p}$
5. Compute steepest descent $\nabla I \frac{\partial W}{\partial \mathbf{p}}$ $W(\mathbf{x};\mathbf{p})$
6. Compute Inverse Hessian H^{-1}
7. Multiply steepest descend with error $\sum_{\mathbf{x}} \left[\nabla I \frac{\partial W}{\partial \mathbf{p}} \right]^{T} [T(\mathbf{x}) - I(W(\mathbf{x};\mathbf{p}))]$
8. Compute $\Delta \mathbf{p}$
9. Update parameters $\mathbf{p} \rightarrow \mathbf{p} + \Delta \mathbf{p}$

Stability of gradient decent iterations depends on …

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

Inverting the Hessian

$$H = \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^{\top} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]$$

When does the inversion fail?

H is singular. But what does that mean?

Above the noise level

$$\lambda_1 \gg 0$$

$$\lambda_2 \gg 0$$

both Eigenvalues are large

Well-conditioned

both Eigenvalues have similar magnitude

Concrete example: Consider translation model

$$\mathbf{W}(\boldsymbol{x};\boldsymbol{p}) = \begin{bmatrix} x + p_1 \\ y + p_2 \end{bmatrix} \qquad \frac{\mathbf{W}}{\partial \boldsymbol{p}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Hessian

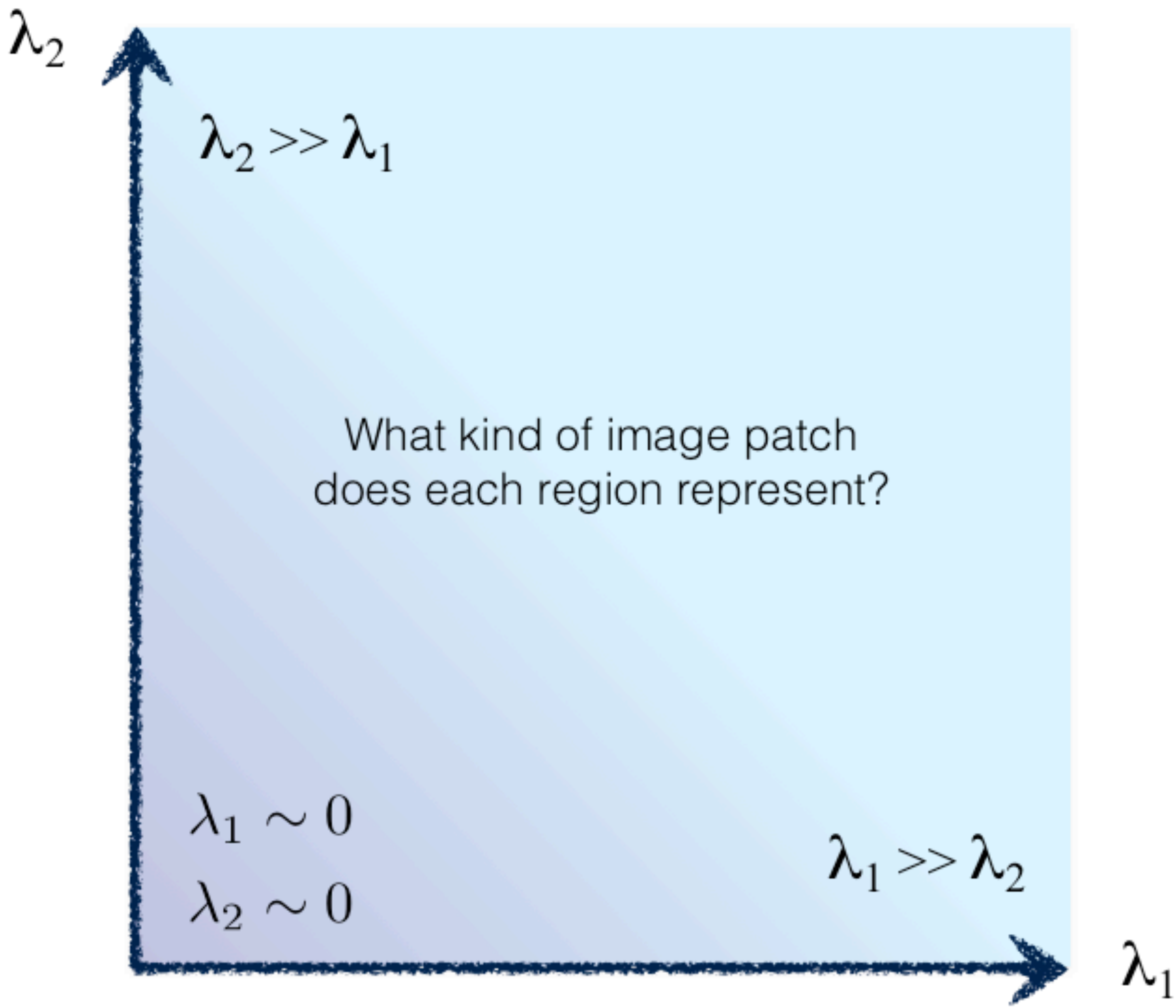
$$H = \sum_{\boldsymbol{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]^\top \left[\nabla I \frac{\partial \mathbf{W}}{\partial \boldsymbol{p}} \right]$$

$$= \sum_{\boldsymbol{x}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_x \\ I_y \end{bmatrix} \begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

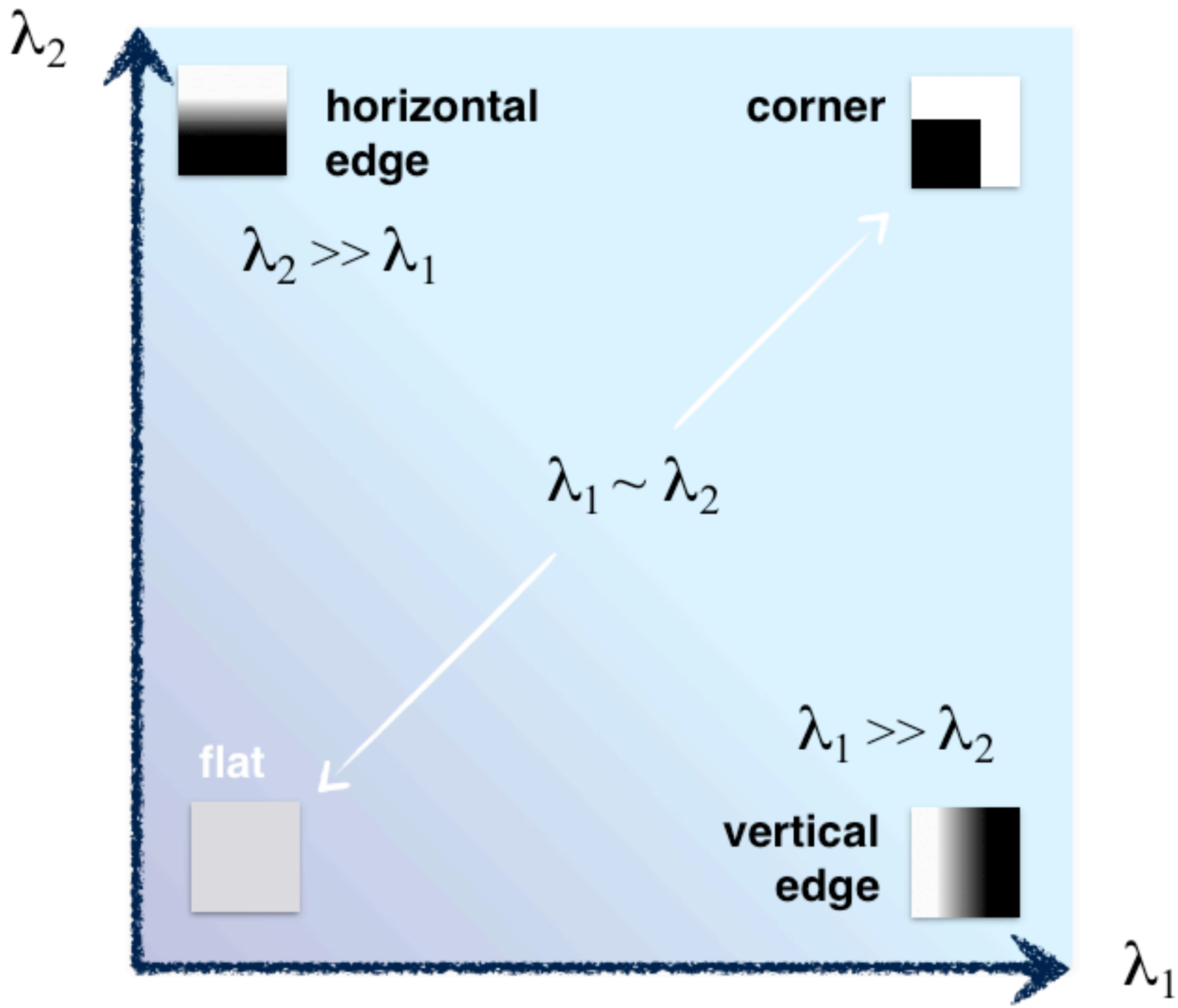
$$= \begin{bmatrix} \sum_{\boldsymbol{x}} I_x I_x & \sum_{\boldsymbol{x}} I_y I_x \\ \sum_{\boldsymbol{x}} I_x I_y & \sum_{\boldsymbol{x}} I_y I_y \end{bmatrix}$$

How are the eigenvalues related to image content?

interpreting eigenvalues



interpreting eigenvalues



KLT algorithm

1. Find corners satisfying $\min(\lambda_1, \lambda_2) > \lambda$

2. For each corner compute displacement to next frame using the Lucas-Kanade method

3. Store displacement of each corner, update corner position

4. (optional) Add more corner points every M frames using 1

5. Repeat 2 to 3 (4)

6. Returns long trajectories for each corner point

Demo

Codes

- KLT: An Implementation of the Kanade-Lucas-Tomasi Feature Tracker http://cecas.clemson.edu/~stb/klt/
- OpenCV – KLT tracker http://docs.opencv.org/trunk/d7/d8b/tutorial_py_lucas_kanade.html

References

- Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. International Joint Conferenceon Artificial Intelligence, pages 674–679, 1981.
- Carlo Tomasi and Takeo Kanade. Detection and Tracking of Point Features. Carnegie Mellon University Technical Report CMU-CS-91-132, April 1991.
- Jianbo Shi and Carlo Tomasi. Good Features to Track. IEEE Conference on Computer Vision and Pattern Recognition, pages 593–600, 1994.
- Lucas-Kanade 20 Years On: A Unifying Framework. International Journal of Computer Vision February 2004, Volume 56, Issue 3, pp 221–255