Motion Estimation
—— Optical Flow (I)

Dr. Xiqun Lu
College of Computer Science
Zhejiang University

Outline

- Optical Flow (Pixel-level)
  - What is optical flow?
  - Lucas-Kanade algorithm (LK) [2]
  - Horn-Schunck algorithm (HS) [3]
- BMA (Block-level)
  - The principle of BMA
  - Full search scheme
  - Three step search [4]
  - New three step search [5]
  - Four step search [6]
  - Diamond search scheme [7]

Outline

- Optical Flow (Pixel-level)
  - What is optical flow?
  - Lucas-Kanade algorithm (LK) [2]
  - Horn-Schunck algorithm (HS) [3]

- BMA (Block-level)
  - The principle of BMA
  - Full search scheme
  - Three step search [4]
  - New three step search [5]
  - Four step search [6]
  - Diamond search scheme [7]

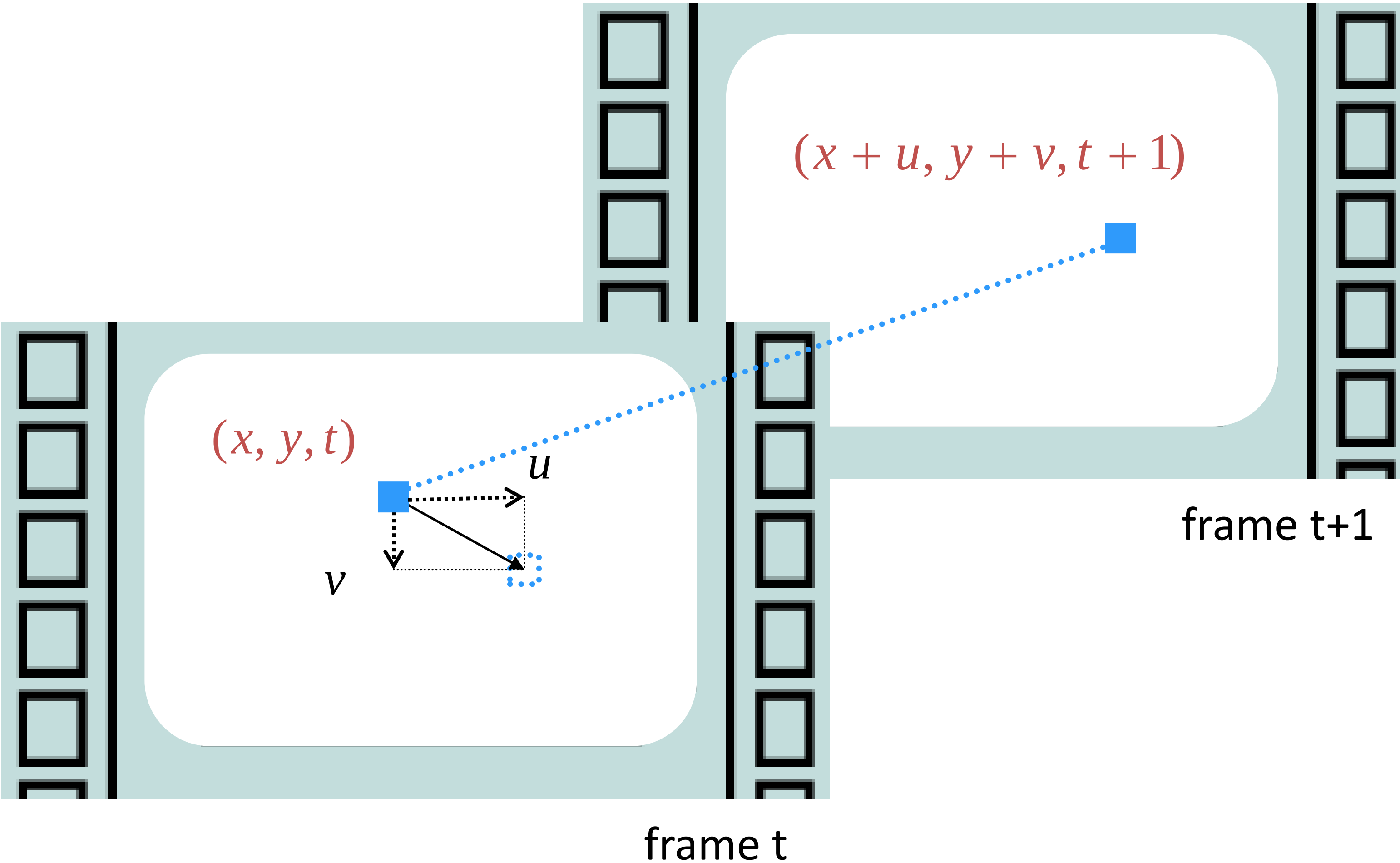
Key Assumptions of Lucas-Kanade [2]

- **Brightness constancy:** projection of the same point looks the same in every frame
- **Small motion:** points do not move very fast
- **Spatial coherence:** points move like their neighbors

[2] B. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

Brightness Constancy

$$\boxed{I(x+u, y+v, t+1) = I(x, y, t)}$$



Brightness Constant

- The color or intensity values of image objects in subsequent frames do no change over time.

$$\boxed{I(x+\delta_x, y+\delta_y, t+\delta_t) = I(x,y,t)}$$

  - Assume that δ_t is enough small, the above eq. can be linearized by a first order Taylor series expansion (we omit the high-order terms here)

$$I(x+\delta_x, y+\delta_y, t+\delta_t) = I(x,y,t) + \frac{\partial I(x,y,t)}{\partial x}\delta_x + \frac{\partial I(x,y,t)}{\partial y}\delta_y + \frac{\partial I(x,y,t)}{\partial t}\delta_t$$

$$\frac{\partial I(x,y,t)}{\partial x}\delta_x + \frac{\partial I(x,y,t)}{\partial y}\delta_y + \frac{\partial I(x,y,t)}{\partial t}\delta_t = 0$$

  - The **optical flow constraint** (for gray images) is

$$\boxed{I_x u + I_y v + I_t = 0}$$

where $I_x = \dfrac{\partial I(x,y,t)}{\partial x}, I_y = \dfrac{\partial I(x,y,t)}{\partial y}, I_t = \dfrac{\partial I(x,y,t)}{\partial t}, u = \dfrac{\delta_x}{\delta_t}, v = \dfrac{\delta_y}{\delta_t}$

How to get more equations for a pixel?

- Spatial Coherence Constraint
  - Assume the pixel's neighbors have the same (*u*, *v*)
  - For example, if we use a 5×5 window, that gives us 25 equations (each pixel in the window has one equation)

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix}$$

Lucas-Kanade Algorithm (LK)[1]

- Overdetermined linear system

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix}$$

$$\underset{(25\times 2)}{\mathbf{A}}\,\underset{(2\times 1)}{\mathbf{d}} = \underset{(25\times 1)}{\mathbf{b}}$$

- Least-squares solution of $\mathbf{d}$ — $\boxed{\min_{\mathbf{d}} \left\| \mathbf{Ad} - \mathbf{b} \right\|_2^2}$

Lucas-Kanade Algorithm (LK)[1]

- Least-square solution of **d**

$$\boxed{\min_{\mathbf{d}} \|\mathbf{A}\mathbf{d}-\mathbf{b}\|_2^2 \rightarrow (\mathbf{A}\mathbf{d}-\mathbf{b})^T(\mathbf{A}\mathbf{d}-\mathbf{b})}$$

$$(\mathbf{A}^T\mathbf{A})\mathbf{d} = \mathbf{A}^T\mathbf{b}$$

$$\boxed{\underbrace{\begin{pmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_y I_x & \sum I_y I_y \end{pmatrix}}_{\mathbf{A}^T\mathbf{A}} \underbrace{\begin{pmatrix} u \\ v \end{pmatrix}}_{\mathbf{d}} = -\underbrace{\begin{pmatrix} \sum I_x I_t \\ \sum I_y I_t \end{pmatrix}}_{\mathbf{A}^T\mathbf{b}}}$$

The summations are over all pixels in the local window.

Conditions for solvability

$$\boxed{\underbrace{\begin{pmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_y I_x & \sum I_y I_y \end{pmatrix}}_{\mathbf{A}^T\mathbf{A}} \underbrace{\begin{pmatrix} u \\ v \end{pmatrix}}_{\mathbf{d}} = -\underbrace{\begin{pmatrix} \sum I_x I_t \\ \sum I_y I_t \end{pmatrix}}_{\mathbf{A}^T\mathbf{b}}}$$

- When is this solvable?
  - $\mathbf{A}^T\mathbf{A}$ should be invertible
  - $\mathbf{A}^T\mathbf{A}$ should not be too small due to noise
    - eigenvalues λ_1 and λ_2 of $\mathbf{A}^T\mathbf{A}$ should not be too small
  - $\mathbf{A}^T\mathbf{A}$ should be well-conditioned
    - λ_1 / λ_2 should not be too large (λ_1 = the larger eigenvalue)

Flat region

$\mathbf{A}^T\mathbf{A}$

- gradients have small magnitude
- small λ_1, small λ_2

Edge

$$\mathbf{A}^T\mathbf{A}$$

– gradients very large or very small

– large λ_1, small λ_2

High-texture region

$\mathbf{A}^T\mathbf{A}$

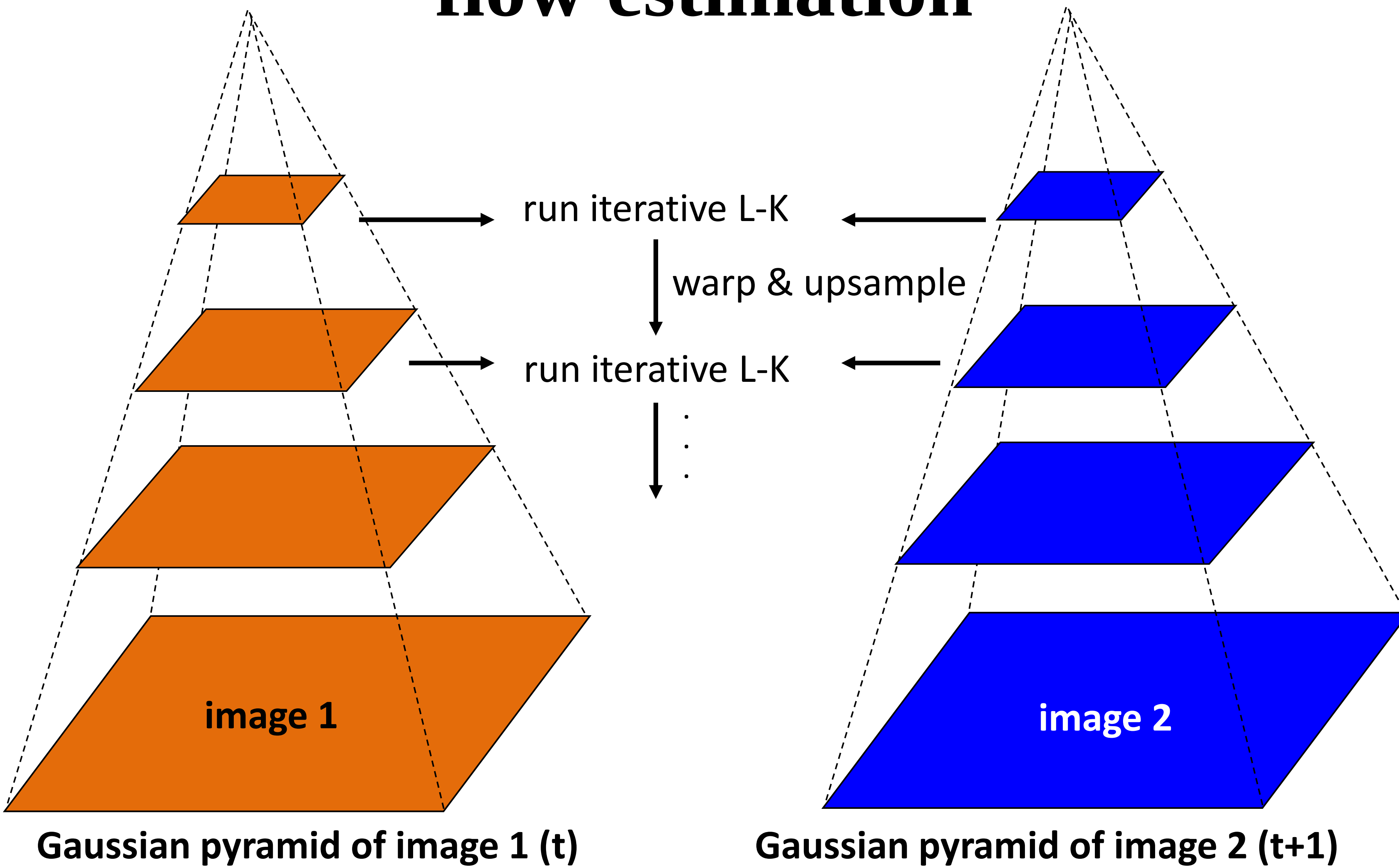
– gradients are different, large magnitudes

– large λ_1, large λ_2

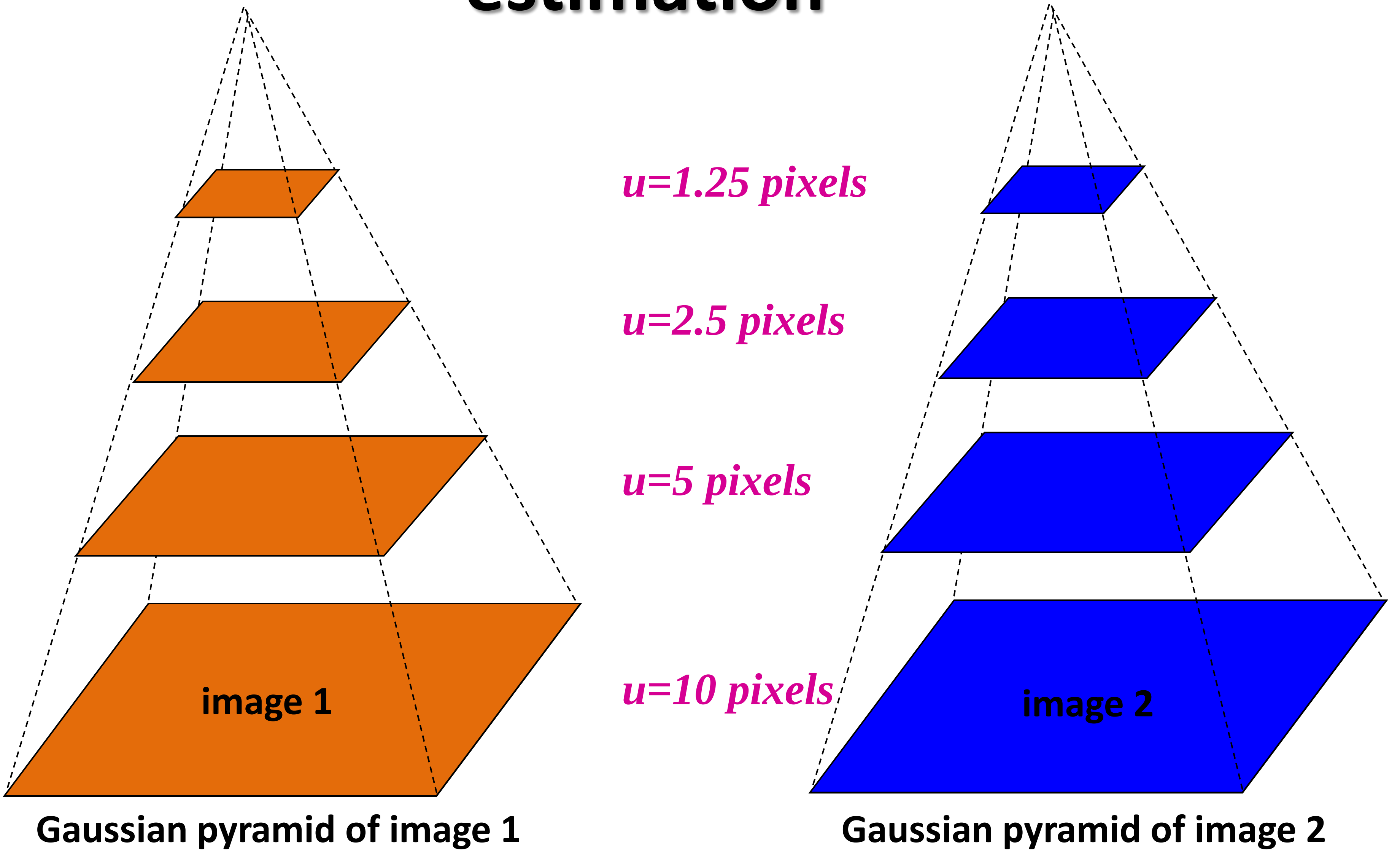
Errors in Lucas-Kanade Algorithm

- When the assumptions are violated
  - Brightness constancy is **not** satisfied
    - Gradient Constancy
  - The motion is **not** small
    - To estimate optical flow in a coarse-to-fine hierarchical way
  - A point does **not** move like its neighbors
    - What’s the ideal size of local analysis window?

Coarse-to-fine Hierarchical optical flow estimation

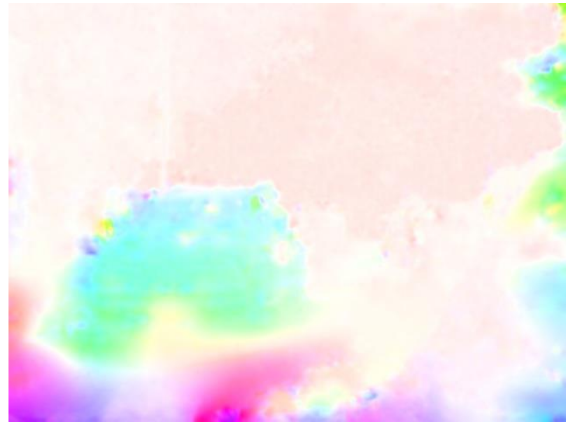


Coarse-to-fine optical flow estimation

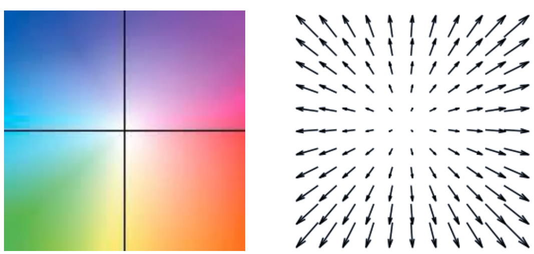


Example

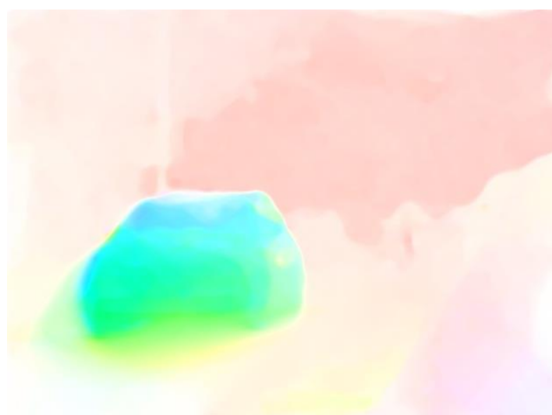
Input two frames



Coarse-to-fine LK



Flow visualization



Coarse-to-fine LK with median filtering

References

- [1] G. Johansson, “Visual perception of biological motion and a model for its analysis", *Perception and Psychophysics*, vol.14, 201-211, 1973.
- [2] B. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in Proc. of *International Joint Conf. On Artificial Intelligence*, pp.674-679, 1981.
- [3] B. Horn and B. Schunck, “Determining optical flow,” *Artificial Intelligence*, 17:185-203, 1981.
- [4] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, “Motion compensated interframe coding for video conferencing,” *Proceedings of national Telecommunications conference*, New Orleans, LA, pp.G5.3.1–G5.3.5, Dec. 1981.
- [5] R. Li, B. Zeng, and M. L. Liou, “A new three-step algorithm for block motion estimation,” *IEEE Trans. On Circuits and Systems for Video Technology*, 4(4): 438-442, 1994.
- [6] L.-M. Po and W.-C. Ma, “A novel four-step search algorithm for fast block motion estimation,” *IEEE Trans. On Circuits and Systems for Video Technology*, 6(3): 313-317, 1996.
- [7] S. Zhu and K.-K. Ma, “A new diamond search algorithm for fast block-matching motion estimation,” *IEEE Trans. On Image Processing*, 9(2): 287-290, 2000.

Thank You!

Dr. Xiqun Lu

xqlu@zju.edu.cn