Sam Saltwick
October 16th 2018
CMSC426 Fall 2018

# Image Motion

## Information from Image Motion

- 3D motion between observer and scene + structure of the scene
  - Motion Parallax → two static points close by in image with different image motion
  - Larger translational motion corresponds to the point closer by (smaller depth
- Create new image out of magnitudes of horizontal and vertical motion components
  - Edge detection to get motion components

## Motion Analysis Problems

- Correspondence Problem
  - Track corresponding elements across frames
- Reconstruction PRoblem
  - Given a number of corresponding elements, and camera parameters, what can we say about 3D motion and structure of the observed scene?
- Segmentation Problem
  - What regions of the image correspond to *different* moving objects?

## The Aperture Problem

- Line moving through aperture (hole) → only information is motion perpendicular to line.
- Can only find one component of optical flow
  - Only the flow componenet perpendicular to the line feature → Normal Flow

- $I(x,y,t) = I(x,y,t) + \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t}\frac{dt}{dt}$
  - Taylor Series Expansion → Ignore non-linear terms since they are so small with regard to images

# Brightness Constraint Equation

- $E(x,y,t)$ → irradiance
- $u(x,y), v(x,y)$ → optical flow components
- $E(x+u\delta t, y+v\delta t, t+\delta t) = E(x,y,t)$
  - Taylor Expansion → $E(x,y,t) + \delta x\frac{\partial E}{\partial x} + \delta y\frac{\partial E}{\partial y} + \delta t\frac{\partial E}{\partial t} + e = E(x,y,t)$
  - Divide by $\delta t$ and take limit as $\delta t$ → $0$
- Vector Notation → $(\nabla E)^T \cdot v + E_t = 0$
- Interpretation → Values of $(u,v)$ satisfying the constraint equation lie on a straight line in velocity space. A local measurement only provides this constraint line (aperture problem).
- Normal flow $u_n$ → $(E_x, E_y)\cdot(u,v) = -E_t$
- Let **n** = $\frac{(E_x E_y)^T}{\|(E_x,E_y)^T\|}$
- $u_n = (u \cdot n)n = (\frac{-E_x E_t}{E_x^2+E_y}, \frac{-E_y E_t}{E_x^2+E_y^2})^T$

# Optical Flow Equation

- $0 = I_t + \nabla I \cdot [u,v]$
- Use more equations for a pixel
  - Impose additional constraints
  - Assume flow field is smooth → Same flow within small window $(5\times 5)$
    - $(5\times 5)$ window gives 25 equations per pixel
    - $0 = I_t(p_i) + \nabla I(p_i)\cdot[u\ v]$
    - $A_{25\times 2}\cdot d_{2\times 1} = b_{25\times 1}$
    - Use least squares to solve → Minimize $\|Ad-b\|^2\ A$
    - $A^T A = \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix}$
      - This is the corner detection matrix!
- Low texture regions (*i.e. sky*)
  - Gradients have small magnitudes
  - Small $\lambda_1$, small $\lambda_2$
- High texture regions (*i.e. ground*)
  - Gradients are different, large magnitudes

- Large $\lambda_1$, large $\lambda_2$
- **Improvement**
  - Assumption of constant flow is more likely to be wrong as we move away from the point of interest
  - Use weights to control the influence of points
  - Further from p → less weight

# Iterative Refinement

- Iterative Lukas-Kanade Algorithm
  1. Estimate velocity at each pixel by solving Lucas-Kanade equations
  2. Warp H towards I using estimated flow field
  3. Repeat until convergence
- Flow values $(u, v)$ should be on the order of a pixel
  - If they are larger → reduce resolution of image
- Course-to-Fine Optical Flow estimation
  - Gaussian pyramid → Gaussian smooth at each iteration
  - Find correspondance between smoothed versions
  - *i.e.* → $u = 10\,pixels \xrightarrow{\text{smooth}} u = 5 \xrightarrow{\text{smooth}} u = 2.5 \xrightarrow{\text{smooth}} u = 1.25$
  - Run L-K on appropriately scaled u → move to next layer

# Additional Constraints

- Additional constraints are necessary to estimate optical flow
  - Constraints on size of derivatives
  - Parametric models of velocity field
- Horn and Schunck (1981): global smoothness term
  - $e_s = \int\int_D (u_x^2 + u_y^2) + (v_x^2 + v_y^2) dxdy$ → departure from smoothness
  - $e_c = \int\int_D (E_x u + E_y u + E_t)^2 dxdy$ → error in optical flow constraint equation
  - Let $\nabla A = (A_x, A_y)^T$ denote the gradient of A
  - $\int\int (\nabla E \cdot \mathbf{u} + E_t)^2 + \lambda(\|\nabla u\|_2^2 + \|\nabla v\|_2^2) dxdy$ → min
- This is called *regularization*
- Solve by means of calculus of variation
- Lucas Kanade (1984) → Weighted least squares fit to a constant model of **u** in a small neighborhood $\Omega$
- Nagel (1983,87) → Oriented smoothness constraint
  - Smoothness is not imposed across edges

- Uras et al. (1988) → Use constraints on second-order derivatives

# Tichonov Acesnin Regularization

- $L(u, v) = I_x u + I_y v + I_t = 0$
- $\phi_{u,v} = \int \int L^2(u, v) + \lambda[(\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2 + (\frac{\partial v}{\partial x})^2 + (\frac{\partial v}{\partial y})^2]$
- $\frac{\partial \phi}{\partial u} = \frac{\partial \phi}{\partial v} = 0$