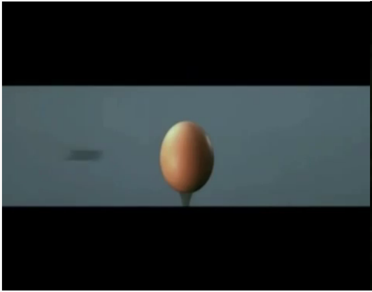
CS-465 Computer Vision

Nazar Khan

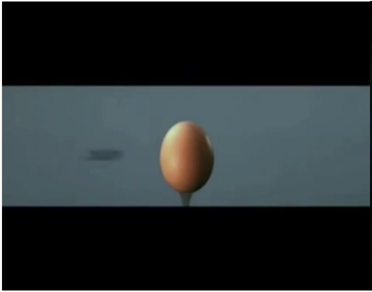
PUCIT

9. Optic Flow

Optic Flow



Optic Flow



Optic Flow

Where does pixel (x, y) in frame z move to in frame $z+1$?

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} u \\ v \end{bmatrix}$$

We want to find the displacement vector $(u, v)^T$ for every pixel.

- Input: image sequence $I(x, y, z)$, where (x, y) specifies the location and z denotes time/frame number
- Goal: displacement vector field of the image structures:
  - optic flow $(u(x, y, z), v(x, y, z))$
- Such correspondence problems are key problems in computer vision.

Grey Value Constancy Assumption

Corresponding pixels should have the same grey value.

Thus, the optic flow between frame z and $z+1$ should satisfy

$$\begin{aligned} & I(x+u, y+v, z+1) = I(x,y,z) \\ \Longrightarrow\ & I(x,y,z) + uI_x(x,y,z) + vI_y(x,y,z) + 1I_z(x,y,z) \approx I(x,y,z) \\ \Longrightarrow\ & I_x(x,y,z)u + I_y(x,y,z)v + I_z(x,y,z) \approx 0 \end{aligned}$$

assuming (u, v) is a small displacement.

Linearized optic flow constraint (OFC)

$$I_x u + I_y v + I_z = 0$$

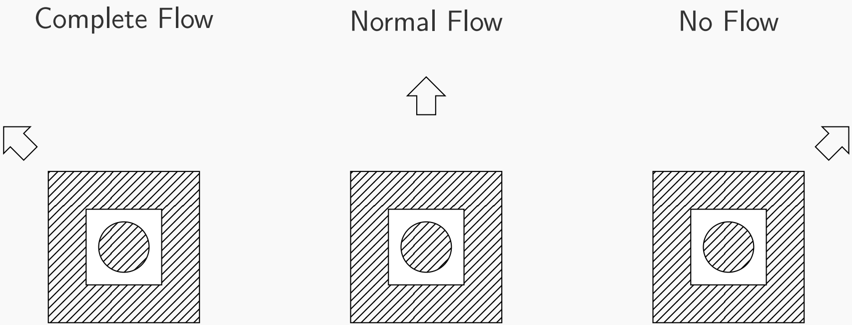
where location (x, y, z) is implied.

How good are the assumptions?

- We have made two assumptions
  1. Gray value constancy
  2. Small displacements (since we use first-order Taylor series approximation)
- Both assumptions are (almost) true in surprisingly many scenarios.
  1. Gray values do not change much between *consecutive*[1] frames.
  2. Objects do not move too much between *consecutive* frames.
     - For large displacements, image pyramid can be used.

[1] For a video recorded at 25 frames per second (fps), consecutive frames are only $\frac{1}{24}$ seconds apart.

Aperture Problem



When seen through an aperture, true movement cannot be determined. Only the component of movement normal to edge direction can be determined.

Normal Flow

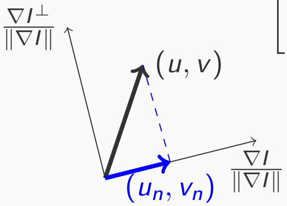
- The OFC is one equation in two unknowns (infinite solutions).
- Can be written as

$$\begin{bmatrix} u \\ v \end{bmatrix}^T \nabla I + I_z = 0$$

- Adding any flow component orthogonal to image gradient does not affect the OFC.

$$\begin{aligned}\left(\begin{bmatrix} u \\ v \end{bmatrix} + k\nabla I^{\perp}\right)^T \nabla I + I_z &= \begin{bmatrix} u \\ v \end{bmatrix}^T \nabla I + k \underbrace{\nabla I^{\perp T} \nabla I}_{0} + I_z \\ &= \begin{bmatrix} u \\ v \end{bmatrix}^T \nabla I + I_z \\ &= 0\end{aligned}$$

Normal Flow



$$\begin{bmatrix} u_n \\ v_n \end{bmatrix} = \left(\begin{bmatrix} u \\ v \end{bmatrix} \bullet \frac{\nabla I}{\|\nabla I\|} \right) \frac{\nabla I}{\|\nabla I\|}$$

$$= \frac{-I_z}{\|\nabla I\|} \frac{\nabla I}{\|\nabla I\|} \quad \left(\because \begin{bmatrix} u \\ v \end{bmatrix} \bullet \nabla I + I_z = 0 \right)$$

$$= \frac{-1}{I_x^2 + I_y^2} \begin{bmatrix} I_x I_z \\ I_y I_z \end{bmatrix}$$

- Only the component of flow in the direction of the gradient ∇I can be computed.
- Since gradient is normal to the edge direction, this flow vector is called the *normal flow*.
- To compute a better estimate of optic flow, we need to make some assumptions.

Local Optic Flow Method of Lucas & Kanade

- Lucas & Kanade make the following assumption:

> Pixels around (i, j) all have the same displacement (u, v).

- For 3×3 neighbourhoods, this gives 9 OFCs all having the same 2 unknowns (u, v).
- The optimal unknown displacement minimizes the sum-squared-error

$$E(u, v) = \frac{1}{2} \sum_{\mathcal{N}_{ij}} (I_x u + I_y v + I_z)^2$$

Local Optic Flow Method of Lucas & Kanade

- Setting $\frac{\partial E}{\partial u} = 0$ and $\frac{\partial E}{\partial v} = 0$ yields a linear system

$$\begin{bmatrix} \sum_{\mathcal{N}_{ij}} I_x^2 & \sum_{\mathcal{N}_{ij}} I_x I_y \\ \sum_{\mathcal{N}_{ij}} I_x I_y & \sum_{\mathcal{N}_{ij}} I_y^2 \end{bmatrix} \begin{bmatrix} u^* \\ v^* \end{bmatrix} = \begin{bmatrix} -\sum_{\mathcal{N}_{ij}} I_x I_z \\ -\sum_{\mathcal{N}_{ij}} I_y I_z \end{bmatrix}$$

- Replacing the sums by Gaussian averaging yields

$$\underbrace{\begin{bmatrix} G_\rho * I_x^2 & G_\rho * I_x I_y \\ G_\rho * I_x I_y & G_\rho * I_y^2 \end{bmatrix}}_{A} \begin{bmatrix} u^* \\ v^* \end{bmatrix} = \begin{bmatrix} -G_\rho * I_x I_z \\ -G_\rho * I_y I_z \end{bmatrix} \tag{1}$$

- Notice the re-appearance of the structure tensor which now serves as the system matrix. Previously, we used it for corner detection.
- Flow vector can be found if $\text{rank}(A) = 2$.

Local Optic Flow Method of Lucas & Kanade

- If $\text{rank}(A) = 0$, no gradients exist in the neighbourhood. So no optic flow can be computed.
- If $\text{rank}(A) = 1$, gradient vectors over all pixels in the neighbourhood are identical. Only normal flow can be computed.

$$\begin{bmatrix} u_n \\ v_n \end{bmatrix} = \frac{-1}{I_x^2 + I_y^2} \begin{bmatrix} I_x I_z \\ I_y I_z \end{bmatrix}$$

- To save computations, avoid computing rank.

$$\text{trace}(A) = A_{11} + A_{22} \approx 0 \implies \text{rank}(A) = 0$$

$$\text{trace}(A) \not\approx 0 \text{ and } \det(A) = A_{11}A_{22} - A_{12}^2 \approx 0 \implies \text{rank}(A) = 1$$

Lucas & Kanade
Algorithm

Input: Frames I_1 and I_2.

Parameters:

1) Noise smoothing scale σ,
2) Gradient smoothing scale ρ,
3) Thresholds τ_{trace} and τ_{det}.

1. Compute Gaussian derivatives at noise smoothing scale σ
$$I_x = \frac{\partial G_\sigma}{\partial x} * I_1 \quad \text{and} \quad I_y = \frac{\partial G_\sigma}{\partial y} * I_1$$
2. Compute temporal derivative $I_z = I_2 - I_1$.
3. Compute the products
$$I_x^2 \quad I_y^2 \quad I_x I_y \quad I_x I_z \quad \text{and} \quad I_y I_z$$
4. Smooth the products at gradient smoothing scale ρ
$$G_\rho * I_x^2 \quad G_\rho * I_y^2 \quad G_\rho * I_x I_y \quad G_\rho * I_x I_z \quad \text{and} \quad G_\rho * I_y I_z$$
and construct the linear system in (1) at every pixel.

Lucas & Kanade
Algorithm

5. For every pixel, solve the linear system conditioned on the rank.
   if $A_{11} + A_{22} < \tau_{\text{trace}}$
   rank(A)=0 so no flow
   else if $A_{11}A_{22} - A_{12}^2 < \tau_{\text{det}}$
   rank(A)=1 so normal flow
   else
   rank(A)=2 so complete optic flow

Visualising Displacement Vectors

The HSV Color Space

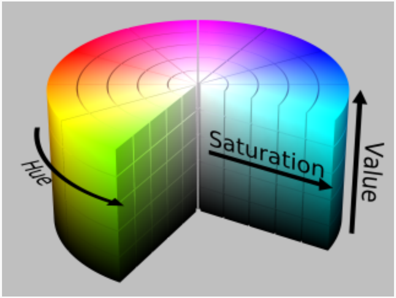


Figure: The HSV color space. Taken from `http://reilley4color.blogspot.com/2016/05/munsell-hue-circle.html`.

Each color is represented by 3 values

1. **H**ue or shade as an angle from 0° to 360°.
2. **S**aturation or strength of the color
3. **V**alue or brightness

Visualising Displacement Vectors

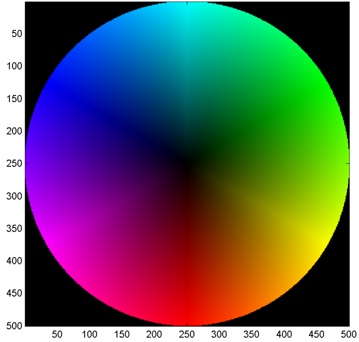


Figure: Vector angle represented by hue/shade of color and vector magnitude represented by the saturation/strength of color. HSV color space is useful for such a mapping. $H(x,y) = \theta(x,y)$, $S(x,y) = \sqrt{u(x,y)^2 + v(x,y)^2}$ and $V(x,y) = \text{constant}$.

Lucas & Kanade

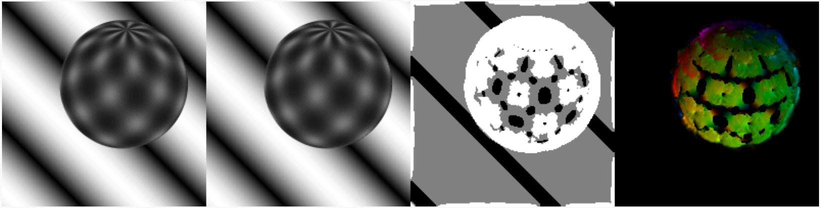


Figure: Left to right: frame 1, frame 2, flow classification and false color visualization of optic flow vectors. For flow classification: white = optic flow, gray = normal flow and black = no flow. Integration scale was $\rho = 1$. Author: N. Khan (2015)

Lucas & Kanade

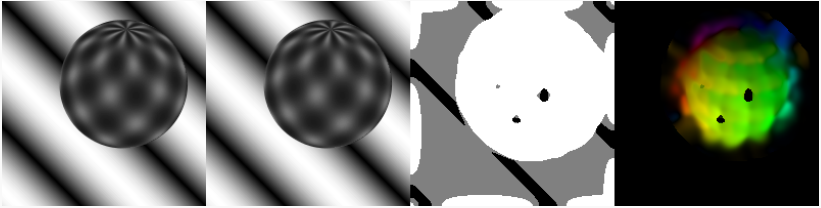


Figure: Left to right: frame 1, frame 2, flow classification and false color visualization of optic flow vectors. For flow classification: white = optic flow, gray = normal flow and black = no flow. Increasing the integration scale ρ to 4 fills up pixels with no flow using values from neighbouring pixels having normal or complete optic flow. Author: N. Khan (2015)

Lucas & Kanade

Summary

Advantages

- Simple and fast method.
- Requires only two frames (low memory requirements).
- Good value for money: results often superior to more complicated approaches.

Disadvantages

- Problems at locations where the local constancy assumption is violated: flow discontinuities and non-translatory motion (e.g. rotation).
- Local method that does not compute the flow field at all locations.

> Next we study a global method that produces dense flow fields (*i.e.*, at every pixel).

Variational Method of Horn & Schunck

- At some given time z the optic flow field is determined as minimising the function $(u(x,y), v(x,y))^T$ of the energy functional

$$E(u,v) = \frac{1}{2}\sum_{x,y}\left(\underbrace{(I_x u + I_y v + I_z)^2}_{\text{data term}} + \alpha \underbrace{\left(\|\nabla u\|^2 + \|\nabla v\|^2\right)}_{\text{smoothness term}}\right)$$

- Has a unique solution that depends continuously on the image data.
- Global method since optic flow at (x,y) depends on all pixels in both frames.

> Notation Alert!
> u and v are 2D arrays of the same size as the frame but *inside the summation* they are also used to refer to a pixel location.

Variational Method of Horn & Schunck

- Regularisation parameter $\alpha > 0$ determines smoothness of the flow field.
  - $\alpha \to 0$ yields the normal flow.
  - The larger the value of α, the smoother the flow field.
- Dense flow fields due to filling-in effect:
  - At locations, where no reliable flow estimation is possible (small $\|\nabla I\|$), the smoothness term dominates over the data term.
- This propagates data from the neighbourhood.
- No additional threshold parameters necessary.

Functionals and Calculus of Variations

- Since u is a function, $E(u, v)$ is a function of a function. A function of a function is also called a *functional*.
- Normal calculus can optimize functions $f(x)$ by requiring $\frac{d}{dx}f|_{x^*} = 0$.
- Functionals are optimized via *calculus of variations*.
- Optimizer of an energy functional

$$E(u, v) = \sum_{x,y} F(x, y, u, v, u_x, u_y, v_x, v_y)$$

must satisfy the so-called *Euler-Lagrange* equations

$$\partial_x F_{u_x} + \partial_y F_{u_y} - F_u = 0$$
$$\partial_x F_{v_x} + \partial_y F_{v_y} - F_v = 0$$

with some boundary conditions.

Functionals and Calculus of Variations

- For our energy functional $E(u, v)$,

$$F = \frac{1}{2}\left(I_x u + I_y v + I_z\right)^2 + \frac{\alpha}{2}\left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right)$$

with partial derivatives

$$\begin{aligned}
F_u &= I_x\left(I_x u + I_y v + I_z\right) \\
F_v &= I_y\left(I_x u + I_y v + I_z\right) \\
F_{u_x} &= \alpha u_x \\
F_{u_y} &= \alpha u_y \\
F_{v_x} &= \alpha v_x \\
F_{v_y} &= \alpha v_y
\end{aligned}$$

Variational Method of Horn & Schunck

- So the Euler-Lagrange equations can be written as

$$\alpha(u_{xx} + u_{yy}) - I_x\left(I_x u + I_y v + I_z\right) = 0$$
$$\alpha(v_{xx} + v_{yy}) - I_y\left(I_x u + I_y v + I_z\right) = 0$$

- At the ith pixel, after writing out the first and second order derivatives, we obtain

$$\frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i}(u_j - u_i) - I_{xi}\left(I_{xi}u_i + I_{yi}v_i + I_{zi}\right) = 0$$
$$\frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i}(v_j - v_i) - I_{yi}\left(I_{xi}u_i + I_{yi}v_i + I_{zi}\right) = 0$$

  where h is the grid size (usually 1).
- Two equations for every pixel.

Variational Method of Horn & Schunck

- For all pixels, this can be written as a sparse but very large linear system $\mathbf{Bx} = \mathbf{d}$.
  - Size of $\mathbf{B}$ will be 69GB for a 256×256 image!
- Large, sparse linear systems can be solved efficiently by *Jacobi's iterative method*.
  1. Let $\mathbf{B} = \mathbf{D} - \mathbf{N}$ with a diagonal matrix $\mathbf{D}$ and a remainder $\mathbf{N}$.
  2. Then the problem $\mathbf{Dx} = \mathbf{Nx} + \mathbf{d}$ is solved iteratively using

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{Nx}^{(k)} + \mathbf{d})$$

- 1 matrix-vector product, 1 vector addition, 1 vector scaling per iteration.

Variational Method of Horn & Schunck

- All of the above boils down to a very simple iterative scheme

$$u_i^{(k+1)} = \frac{\frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} u_j^{(k)} - I_{xi}\left(I_{yi}v_i^{(k)} + I_{zi}\right)}{\frac{\alpha}{h^2}|\mathcal{N}_i| + I_{xi}^2}$$

$$v_i^{(k+1)} = \frac{\frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} v_j^{(k)} - I_{yi}\left(I_{xi}u_i^{(k)} + I_{zi}\right)}{\frac{\alpha}{h^2}|\mathcal{N}_i| + I_{xi}^2}$$

with $k = 0, 1, 2, \ldots$ and an arbitrary initialisation (e.g. zero vector).

Variational Method of Horn & Schunck

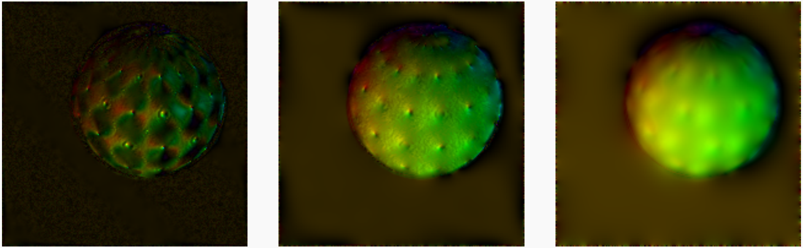


Figure: Left to right: Dense and smooth optic flow fields obtained via Horn & Schunck's variational method for smoothness parameter $\alpha = 0.0000001, 0.00001$ and 0.001 after 400 iterations. Noise smoothing scale was $\sigma = 0.5$. Author: N. Khan (2018)

Variational Method of Horn & Schunck

Variational Method of Horn & Schunck

Variational Method of Horn & Schunck

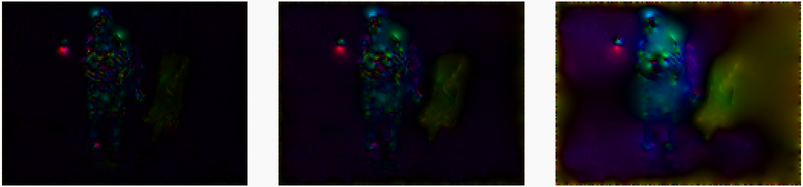


Figure: Left to right: Dense and smooth optic flow fields obtained via Horn & Schunck's variational method for smoothness parameter $\alpha = 0.0001, 0.001$ and 0.01 after 400 iterations. Noise smoothing scale was $\sigma = 0.5$. Author: N. Khan (2018)

Variational Method of Horn & Schunck

Summary

- Variational methods for computing optic flow are global methods.
- Create dense flow fields by filling-in.
- Model assumptions of the variational Horn and Schunck approach:
  1. grey value constancy,
  2. smoothness of the flow field
- Mathematically well-founded method.
- Minimising the energy functional leads to coupled differential equations.
- Discretisation creates a large, sparse linear system of equations that can be solved iteratively, *e.g.*, using the Jacobi method.
- Variational methods can be extended and generalised in numerous ways, with respect to both models and algorithms.