Motion Estimation summary

S. Papadopoulos, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr
Version 3.4.1

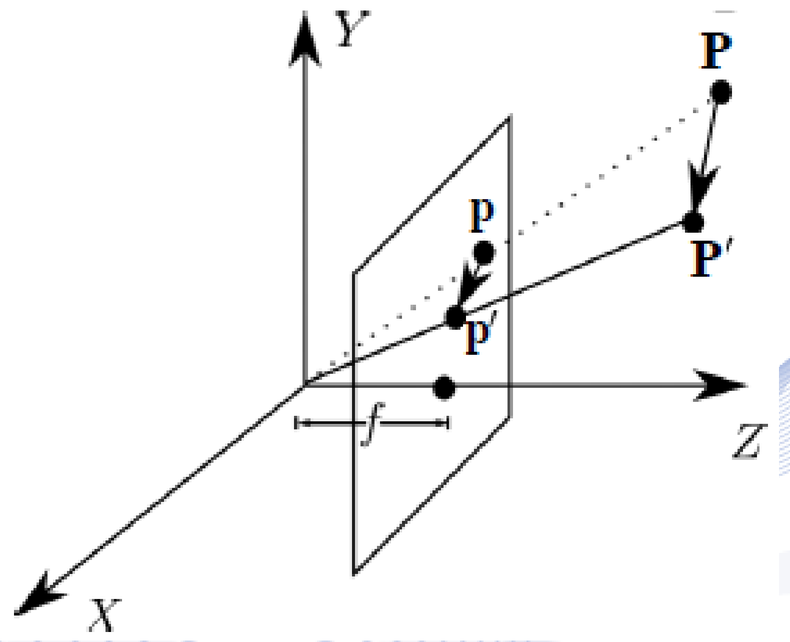
Artificial Intelligence &
Information Analysis Lab

Motion Estimation

- **2D motion**
- 3D motion models
- 2D motion models
- Estimation of 2D correspondence vectors
- Block matching
- Phase correlation
- Optical Flow Equation Methods
- Neural Optical Flow Estimation

2D motion

- Two-dimensional (2D) motion or ***projected motion*** is the perspective projection of the 3D motion on the image plane.
- Object point $\mathbf{P}$ at time t moves to point $\mathbf{P}'$ at t' and its perspective projection in the image plane from $\mathbf{p}$ to $\mathbf{p}'$.



2D motion

- The 2D displacement $t' = t + \ell\Delta t$ can be defined for all points $\mathbf{x}_t = [x, y, t]^T \in \mathbf{R}^3$ by the 2D ***displacement vector*** field $\mathbf{d}_c(x_t; \ell\Delta t)$ as a function of the continuous spatiotemporal variables $[x, y]^T$ and t.
- The sampled 2D displacement field over a sampling is given by:

$$\mathbf{d}(n_1, n_2, n_t; \ell) = \mathbf{d}_p(\mathbf{x}_t; \ell\Delta t)\Big|_{\mathbf{x}_t = \mathbf{V}\mathbf{n}}, \qquad (n_1, n_2, n_t) \in Z^3$$

where $\mathbf{V}$ is a sampling matrix of the grid Λ^3.

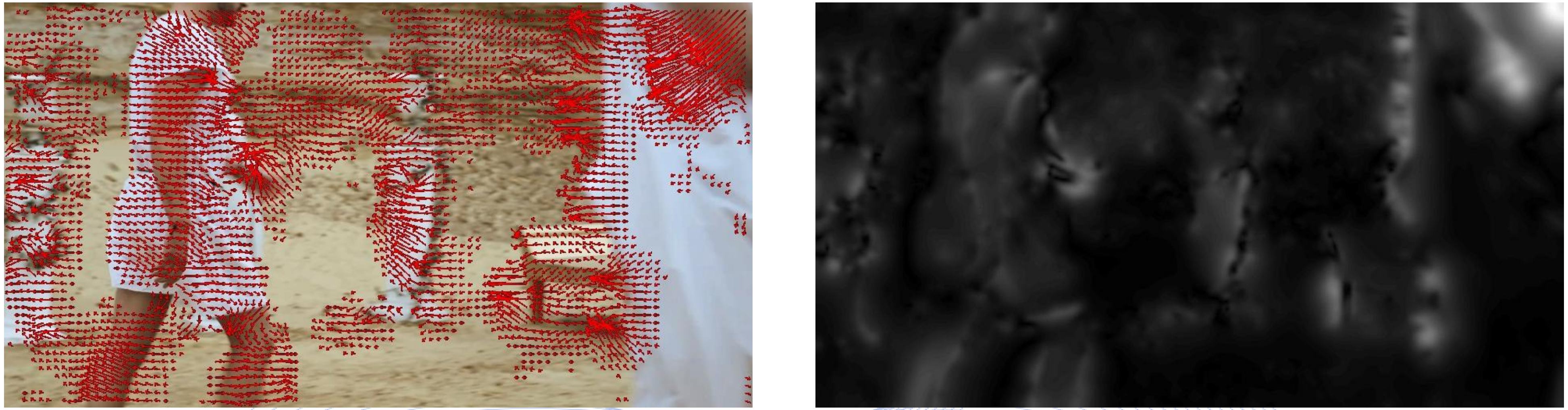
2D motion

- The 3D instantaneous velocity field $[dX/dt, dY/dt, dZ/dt]^T$ produces the projected velocity vector $\mathbf{v_p}(x, y, t)$ at time t.
- Discrete ***2D velocity vector*** field $\mathbf{v}(n_1, n_2, n_t) = \mathbf{v_p}(\boldsymbol{x}_t)$, for $\mathbf{x}_t = \mathbf{Vn} \in \Lambda^3$ and $\mathbf{n} = [n_1, n_2, n_t]^T \in \mathbf{Z}^3$.
- ***Correspondence vector*** denotes the displacement between the corresponding points $\mathbf{x} = [x, y]^T$ on the video frame at time t and $\mathbf{x}' = [x', y']^T$ at time t'.

2D motion

- ***Optical flow*** vector: the derivative of the correspondence vector: $[v_x, v_y]^T = [dx/dt\,, dy/dt]^T$.
- It describes the spatiotemporal changes of luminance $f_a(x, y, t)$.
- ***Motion speed***: magnitude of the motion vector.
- The correspondence or optical flow vectors determine the apparent motion.

2D motion

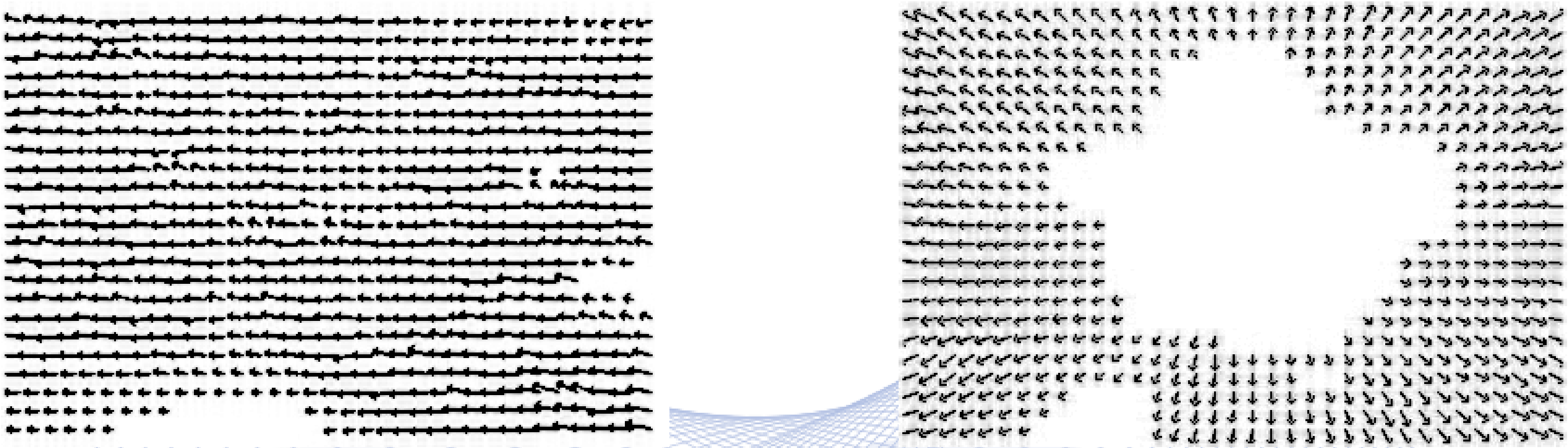


a) Motion field; b) motion speed.

2D motion

- 2D motion can be generated by:
  - Object(s) motion
- Global 2D motion can be generated by:
  - Camera motion (***pan***, ***tilt***)
  - Camera zoom
- 2D apparent motion can be generated by a motion of the illumination source.

2D motion



Global optical flow generated by: a) camera pan and b) zoom.

Motion Estimation

- 2D motion
- **3D motion models**
- 2D motion models
- Estimation of 2D correspondence vectors
- Block matching
- Phase correlation
- Optical Flow Equation Methods
- Neural Optical Flow Estimation

3D Motion Models

- 3D solid object motion can be described by the affine transformation:

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T},$$

where $\mathbf{T}$ is a 3×1 translation vector:

$$\mathbf{T} = \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}.$$

and $\mathbf{R}$ is a 3×3 rotation matrix (various forms).

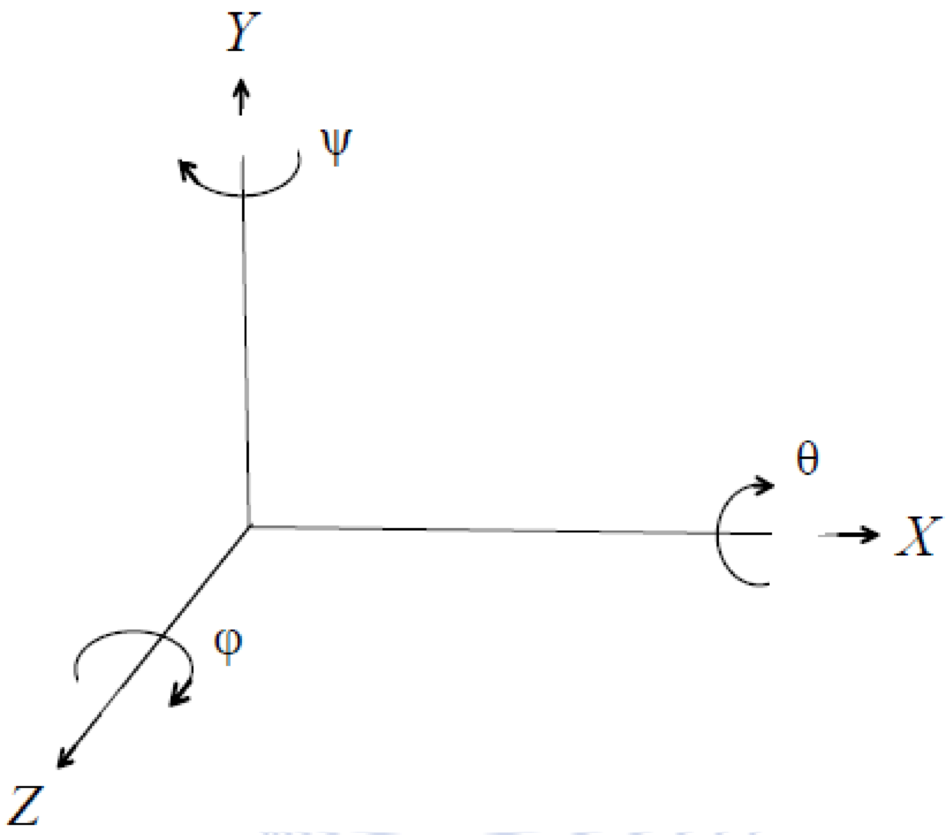
3D Motion Models

- In Cartesian coordinates, $\mathbf{R}$ can be described:
  - either by the ***Euler rotation angles*** about the three coordinate axes X, Y, Z.
  - or by a rotation axis and a rotation angle about this axis.
- The matrices describing the clockwise rotation around each axis in the three dimensional space, are given by:

$$\mathbf{R} = \mathbf{R}_Z \mathbf{R}_Y \mathbf{R}_X.$$

- Their order ***does matter***.
- $\mathbf{R}$ is ***orthonormal***, satisfying $\mathbf{R}^T = \mathbf{R}^{-1}$ and $det(\mathbf{R}) = \pm 1$.

3D Motion Models



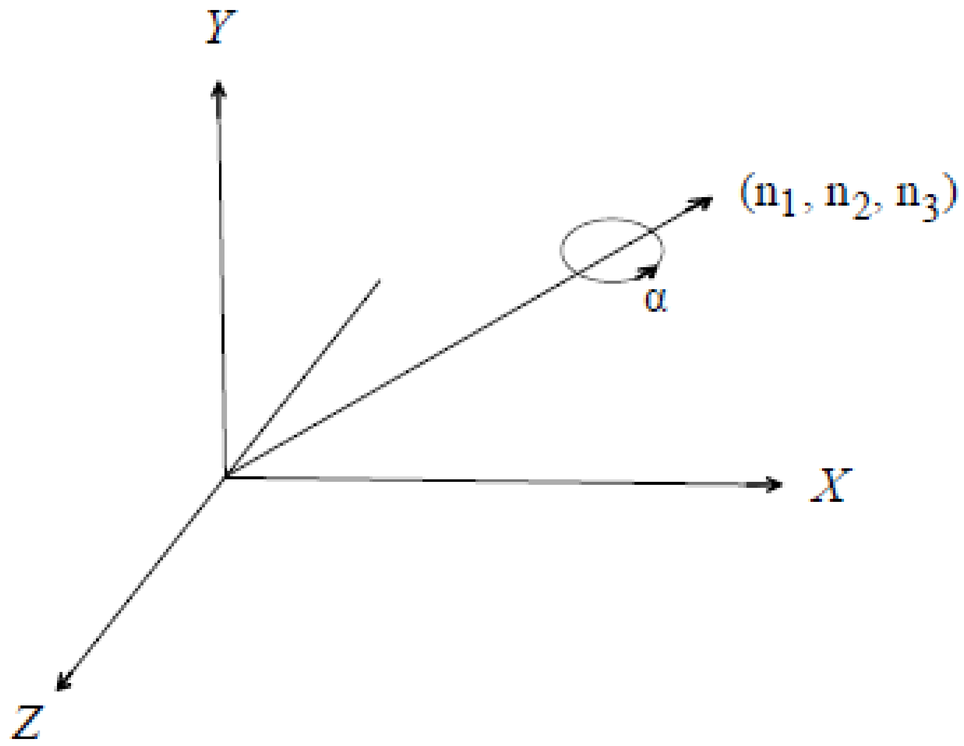
Euler rotation angles.

$$\mathbf{R}_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix},$$

$$\mathbf{R}_Y = \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix},$$

$$\mathbf{R}_Z = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

3D Motion Models



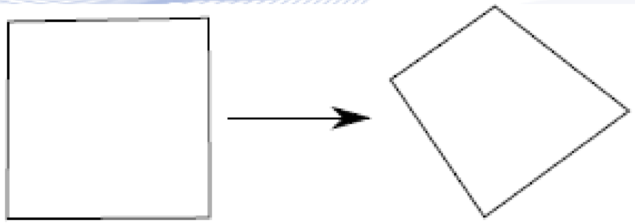
Object rotation about a rotation axis.

2D Motion Models

- ***Projective mapping transformation*** for no camera or object translation along the Z axis, or planar object:

$$x' = \frac{a_1 + a_2 x + a_3 y}{1 + a_7 x + a_8 y}, \qquad y' = \frac{a_4 + a_5 x + a_6 y}{1 + a_7 x + a_8 y}.$$

  - Parallel lines in the 3D space are represented by straight lines, converging to a vanishing point, on the image plane
  - Two successive projective mappings can be synthesized in one projective mapping.

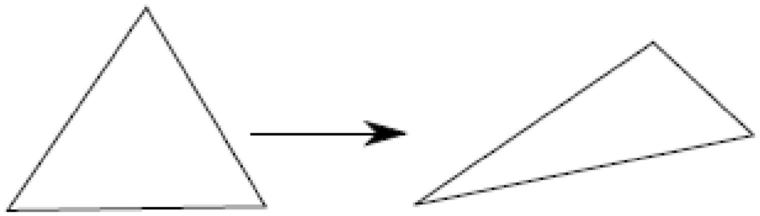


2D Motion Models

- Affine mapping transformation. The projected 2D motion of several camera motions as well as an arbitrary 3D motion of a planar object can be approximated by an affine transformation:

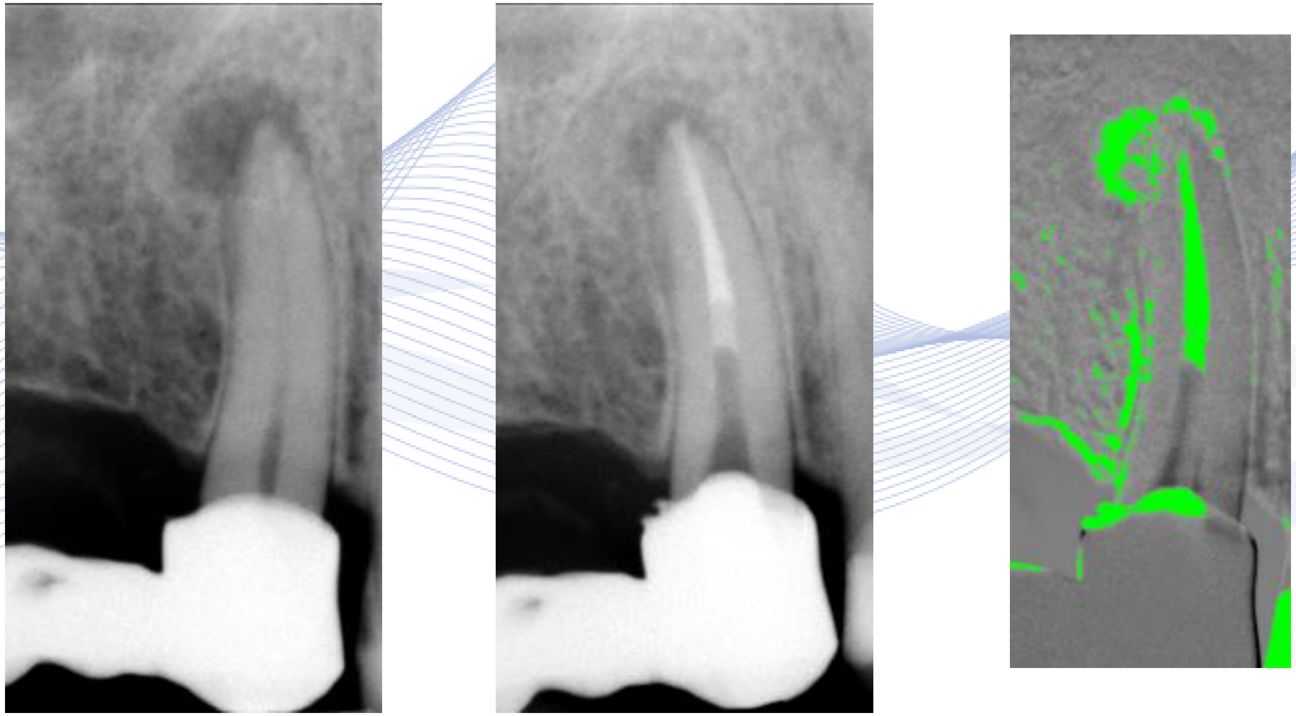
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 + a_2 x + a_3 y \\ a_4 + a_5 x + a_6 y \end{bmatrix}$$

- Deforms a triangle to another by shifting the triangle corners.



2D Motion Models

- 2D affine mapping transformation: it describes 2D rotation, translation and scaling.
- It can be used for 2D image registration.



Subtractive radiography.

2D Motion Models

- 2D affine mapping transformation for image mosaicing.



Motion Estimation

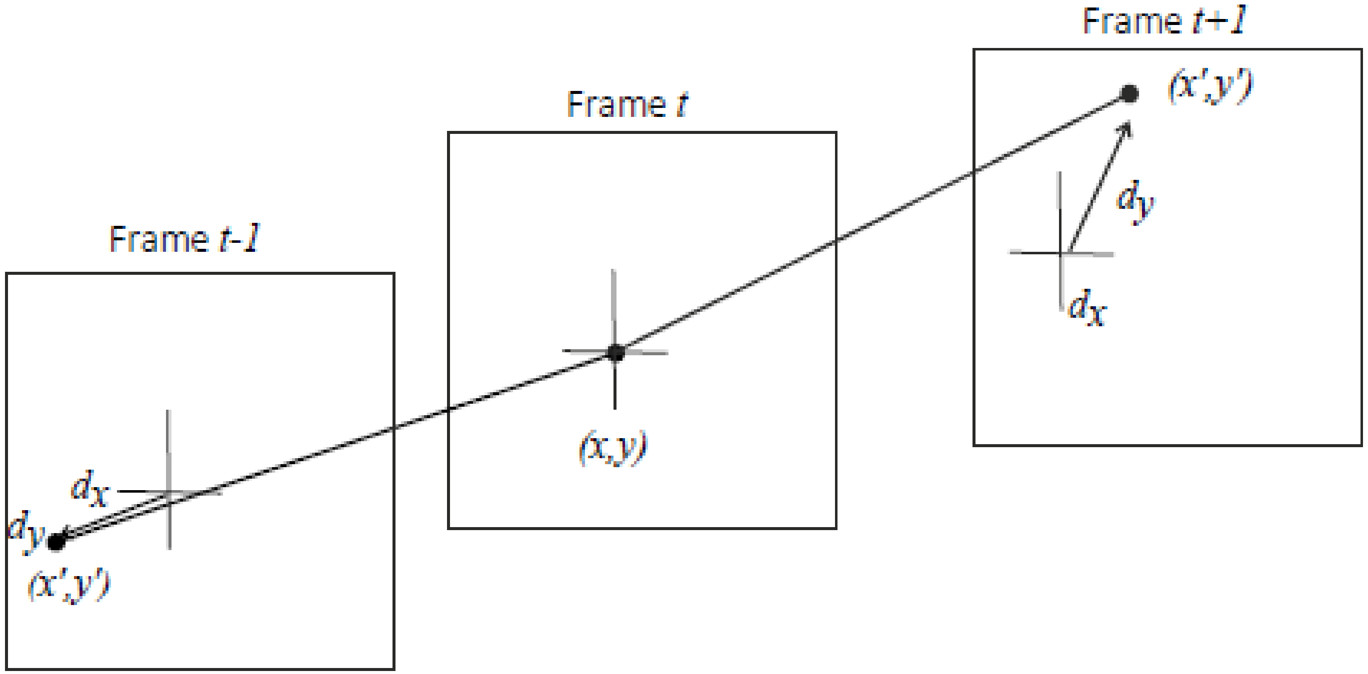
- 2D motion
- 3D motion models
- 2D motion models
- **Estimation of 2D correspondence vectors**
- Block matching
- Phase correlation
- Optical Flow Equation Methods
- Neural Optical Flow Estimation

Estimation of 2D correspondence vectors

- The correspondence problem can be studied:
  - As ***forward motion estimation***:
    - the motion vector is defined from frame t to $t+1$;
    - displacement vectors $\mathbf{d}(x,y) = [dx(x,y),\ dy(x,y)]^T$ should satisfy:

$$f(x,y,t) = f(x+dx(x,y), y+dy(x,y), t+1).$$

Estimation of 2D correspondence vectors



Forward and backward 2D motion estimation.

Estimation of 2D correspondence vectors

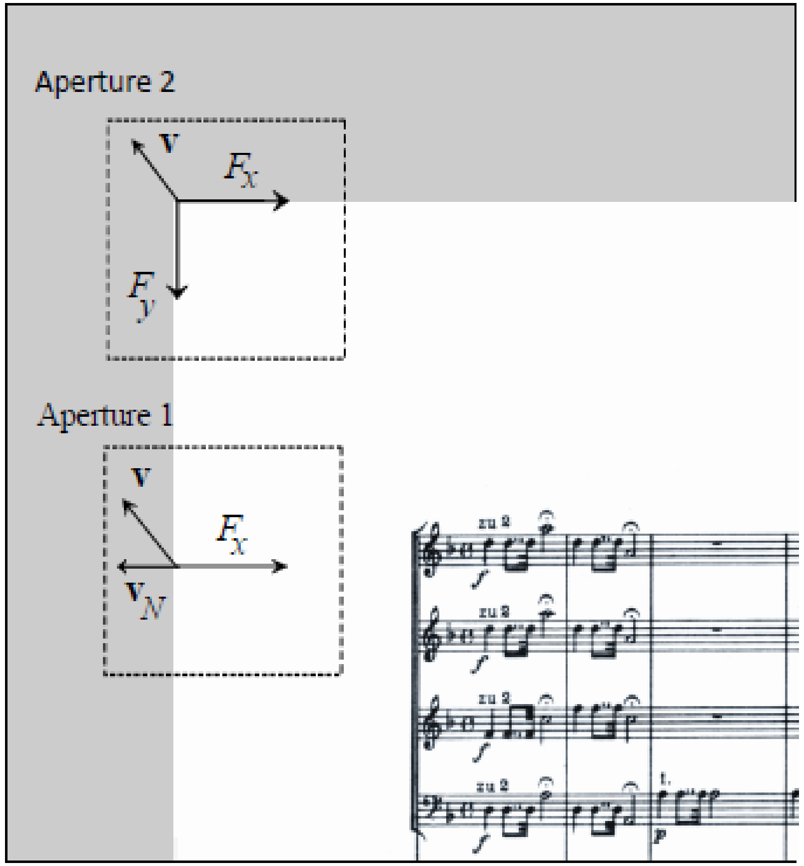
- For video compression, backward motion estimation is preferred.
- Problems associated with the uniqueness of object point matching over successive video frames:
  - ***Occlusion***: no correspondence can be found between occluded and un-occluded object or background region, due to object motion.
  - Partial or total occlusion. ***Self-occlusion***.

Estimation of 2D correspondence vectors

Object occlusion (right) and de-occlusion (left).

Estimation of 2D correspondence vectors

- ***Aperture problem***: only local spatial information (within the camera aperture) is used for motion estimation.



Quality metrics for motion estimation

- ***Peak Signal to Noise Ratio*** (***PSNR***): Metric for testing the quality of motion estimator results, measured in dB:

$$PSNR = 10\log_{10}\frac{N\times M}{\sum[f(x,y,t)-f(x+dx(x,y),\ y+dy(x,y),\ t-1)]^2}.$$

- $N\times M$: video frame size in pixels.
- Video luminance scaled in the range $[0,1]$.
- dx, dy: the displacement components resulting from motion estimation at pixel $\mathbf{p}=[x,y]^T$.

Motion Estimation

- 2D motion
- 3D motion models
- 2D motion models
- Estimation of 2D correspondence vectors
- **Block matching**
- Phase correlation
- Optical Flow Equation Methods
- Neural Optical Flow Estimation

Block matching

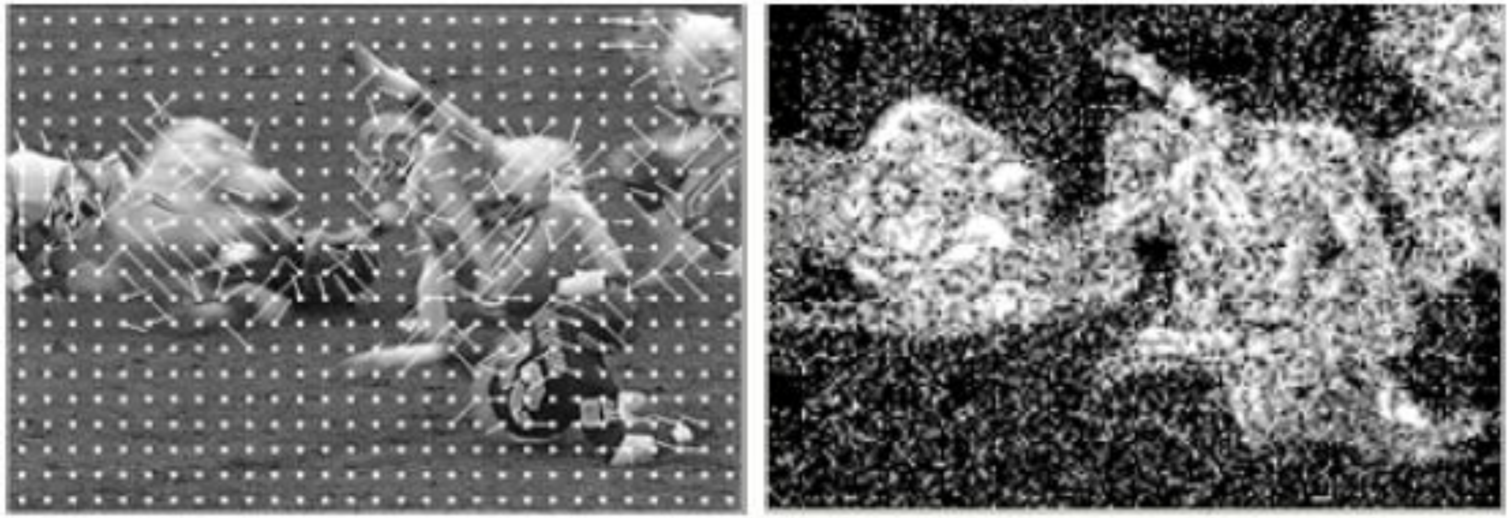
Block matching matches image blocks in consecutive video frames.

Block displacement $\mathbf{d}$ can be estimated by minimizing the displaced section difference for selecting the optimal displacement $\mathbf{d} = [dx, dy]^T$:

$$\min_{dx,dy} E(\mathbf{d}) = \sum_{n_1} \sum_{n_2} \|f(n_1, n_2, t) - f(n_1 + dx, n_2 + dy, t - 1)\|.$$

- n_1, n_2 are pixel coordinates.
- L_1, L_2, L_p norms can be used for displaced frame difference estimation.

Block matching



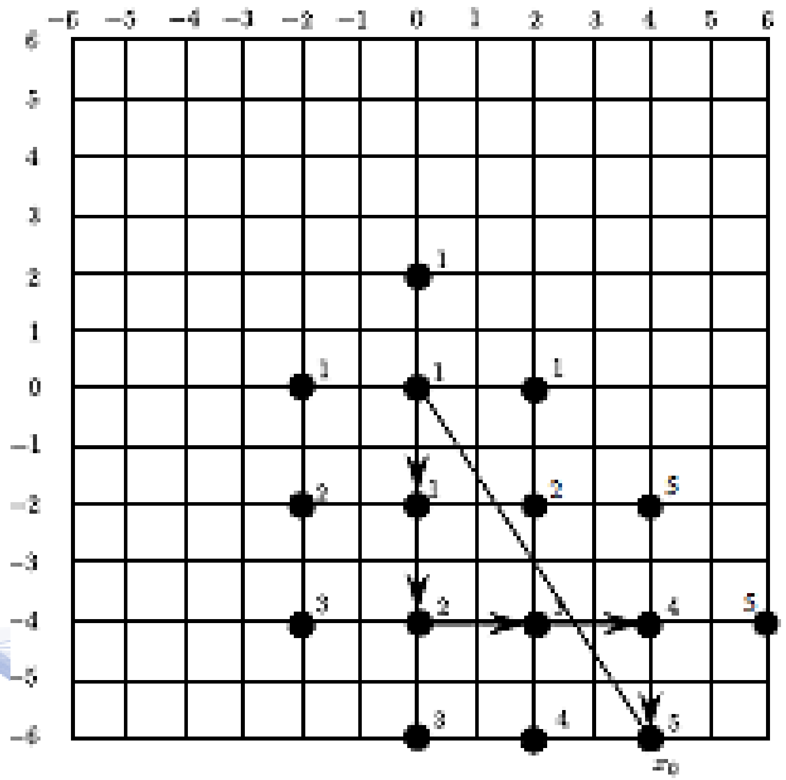
Sparse and dense motion fields.

Artificial Intelligence & Information Analysis Lab

Block matching

- Supposing a $N \times N$ video frame and a $m \times m$ ***pixel block*** $\mathcal{B}$ centered at $\mathbf{x}_0$ at frame t:
  - The ***search area*** at frame $t - 1$ for the $E(\mathbf{d})$ minimum is a $(2d_{max} + 1) \times (2d_{max} + 1)$ block.
  - Block $\mathcal{B}$ is moved by $\pm d_{max}$ horizontally and vertically around $\mathbf{x}_0$ and the minimum $E(\mathbf{d})$ in $(2d_{max} + 1)^2$ positions is calculated.

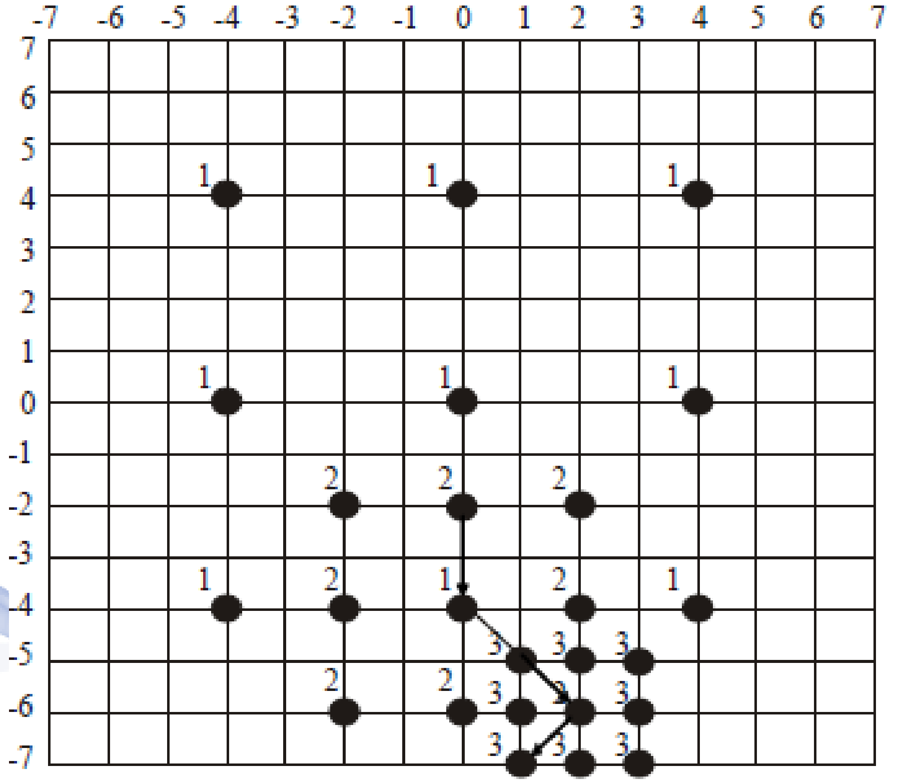
Block matching



- $d_{max} = 6$ pixels.
- Displacement from $\mathbf{x}_0 = [0,0]^T$ to $\mathbf{x}_0' = [4,-6]^T$.

Artificial Intelligence & Information Analysis Lab

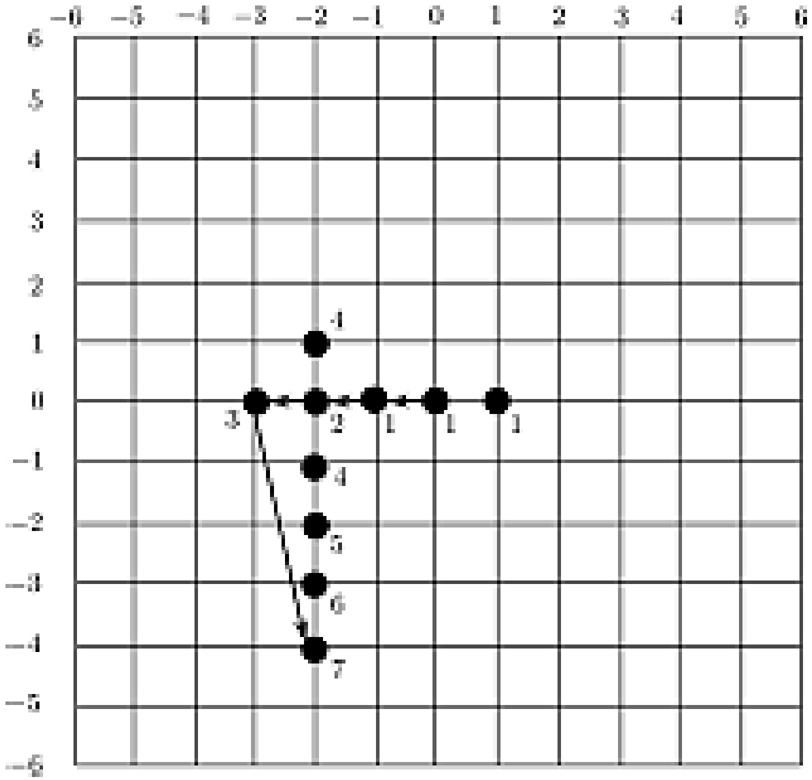
Block matching



Three step search:

- 1st step: Eight pixels around $\mathbf{x}_0$ are checked.
- 2nd step: Eight pixels around the pixel of minimum $E(\mathbf{d})$ of step 1 are searched.
- …
- Search step size reduces at each step.

Block matching



In ***1D search***, $E(\mathbf{d})$ minimum is searched first along the horizontal and then along the vertical direction:

- ***1st step***. Search along the horizontal direction.
- ***2nd step***. Based on the results of step 1, the minimum is searched for along the vertical direction.

Motion Estimation

- 2D motion
- 3D motion models
- 2D motion models
- Estimation of 2D correspondence vectors
- Block matching
- **Phase correlation**
- Optical Flow Equation Methods
- Neural Optical Flow Estimation

Phase correlation

- Relative image blocks displacement is calculated using a normalized cross-correlation function calculated on the 2D spatial or Fourier domain.
- ***Cross-correlation*** between two video frames of size $N_1 \times N_2$ at times t and $t-1$:

$$r_{t,t-1}(n_1,n_2) =$$

$$\sum_{k_1=0}^{N_1-1}\sum_{k_2=0}^{N_2-1} f(k_1,k_2,t)f(n_1+k_1,n_2+k_2,t-1) =$$

$$f(n_1,n_2,t) ** f(-n_1,-n_2,t-1).$$

** denotes a 2D convolution.

Artificial Intelligence &
Information Analysis Lab

Phase correlation

- Taking the Fourier on both sides, we get the expression of complex cross-correlation spectrum:

$$R_{t,t-1}(\omega_x, \omega_y) = F_t^*(\omega_x, \omega_y) F_{t-1}(\omega_x, \omega_y).$$

  * denotes complex conjugation.

- Phase of the cross-correlation spectrum:

$$\tilde{R}_{t,t-1}(\omega_x, \omega_y) = \frac{F_t^*(\omega_x,\omega_y) F_{t-1}(\omega_x,\omega_y)}{|F_t^*(\omega_x,\omega_y) F_{t-1}(\omega_x,\omega_y)|}.$$

Phase correlation

- Effects of using the 2D DFT:

  - Boundary problems,
  - Spectrum leakage,
  - Support area of displacement estimators.

Motion Estimation

- 2D motion
- 3D motion models
- 2D motion models
- Estimation of 2D correspondence vectors
- Block matching
- Phase correlation
- **Optical Flow Equation Methods**
- Neural Optical Flow Estimation

Optical flow equation methods

- The continuous spatiotemporal video luminance $f_a(x,y,t)$, not $f_a(x,y,t)$ does not change along the object motion trajectory.
- For $\mathbf{x}_t = [x,y,t]^T$ on motion trajectory, the ***total derivative*** $\frac{df_a(\mathbf{x}_t)}{dt} = 0$ leads to ***optical flow equation*** (***OFE***):

$$\frac{\partial f_a(\mathbf{x}_t)}{\partial x} v_x(\mathbf{x},t) + \frac{\partial f_a(\mathbf{x}_t)}{\partial y} v_y(\mathbf{x},t) + \frac{\partial f_a(\mathbf{x}_t)}{\partial t} = 0.$$

- $\mathbf{x} = [x,y]^T$, $\mathbf{x}_t = [x,y,t]^T$, $v_x(\mathbf{x},t) = dx/dt$, $v_y(\mathbf{x},t) = dy/dt$.

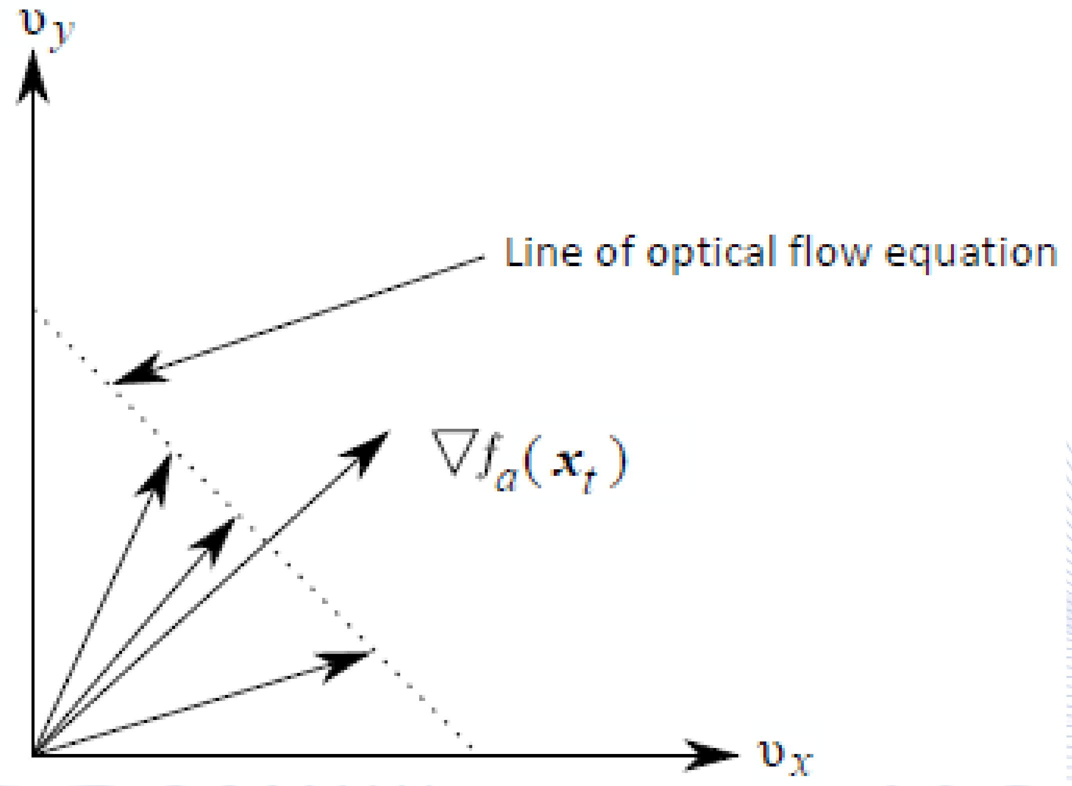
Optical flow equation methods

- OFE has two unknown factors, $v_x(\mathbf{x}, t)$ and $v_y(\mathbf{x}, t)$ for each $(\mathbf{x}, t)$, thus another equation is needed.
- The two velocity vector components are located on a straight line in the space (v_x, v_y).
- OFE can be expressed as:

$$\frac{\partial f_a(\mathbf{x}_t)}{\partial t} + \nabla f_a(\mathbf{x}_t)\mathbf{v}^T(\mathbf{x}_t) = 0,$$

where $\mathbf{v}(\mathbf{x}_t) = \left[v_x(\mathbf{x}_\mathrm{t}, t), v_y(\mathbf{x}_t, t)\right]^T$ and $\nabla f_a(\mathbf{x}_t) = \left[\frac{\partial f_a(\mathbf{x}_t)}{\partial x}, \frac{\partial f_a(\mathbf{x}_t)}{\partial y}\right]^T$.

Optical flow equation methods



Line of optical flow equation.

Adaptive OFE methods

- Directional motion field smoothing constraint:

$$E_2^2\big(\mathbf{v}(\mathbf{x},t)\big) = (\nabla v_x)^T \mathbf{W}(\nabla v_x) + \left(\nabla v_y\right)^T \mathbf{W}\left(\nabla v_y\right).$$

- **W**: a weight matrix punishing changes in the motion field, depending on the spatial image luminance changes:

$$\mathbf{W} = \frac{\mathbf{F} + \alpha\mathbf{I}}{trace(\mathbf{F} + \alpha\mathbf{I})}.$$

- **I**: the identity matrix, α: a scale factor.
- **F**: matrix containing spatial derivatives of $f_a(\mathbf{x}_t)$.

Partial Differentiation in Motion Estimation

Numerical differentiation for spatiotemporal signals (digital video) $f(n_1, n_2. n_t)$:

$$\widehat{f_x} = \frac{1}{4}\{f(n_1+1, n_2, n_t) - f(n_1, n_2, n_t) + f(n_1+1, n_2+1, n_t) - f(n_1, n_2+1, n_t) + f(n_1+1, n_2, n_t+1) - f(n_1, n_2, n_t+1) + f(n_1+1, n_2+1, n_t+1) - f(n_1, n_2+1, n_t+1)\}.$$

Motion Estimation

- 2D motion
- 3D motion models
- 2D motion models
- Estimation of 2D correspondence vectors
- Block matching
- Phase correlation
- Optical Flow Equation Methods
- **Neural Optical Flow Estimation**

Neural Optical Flow estimation

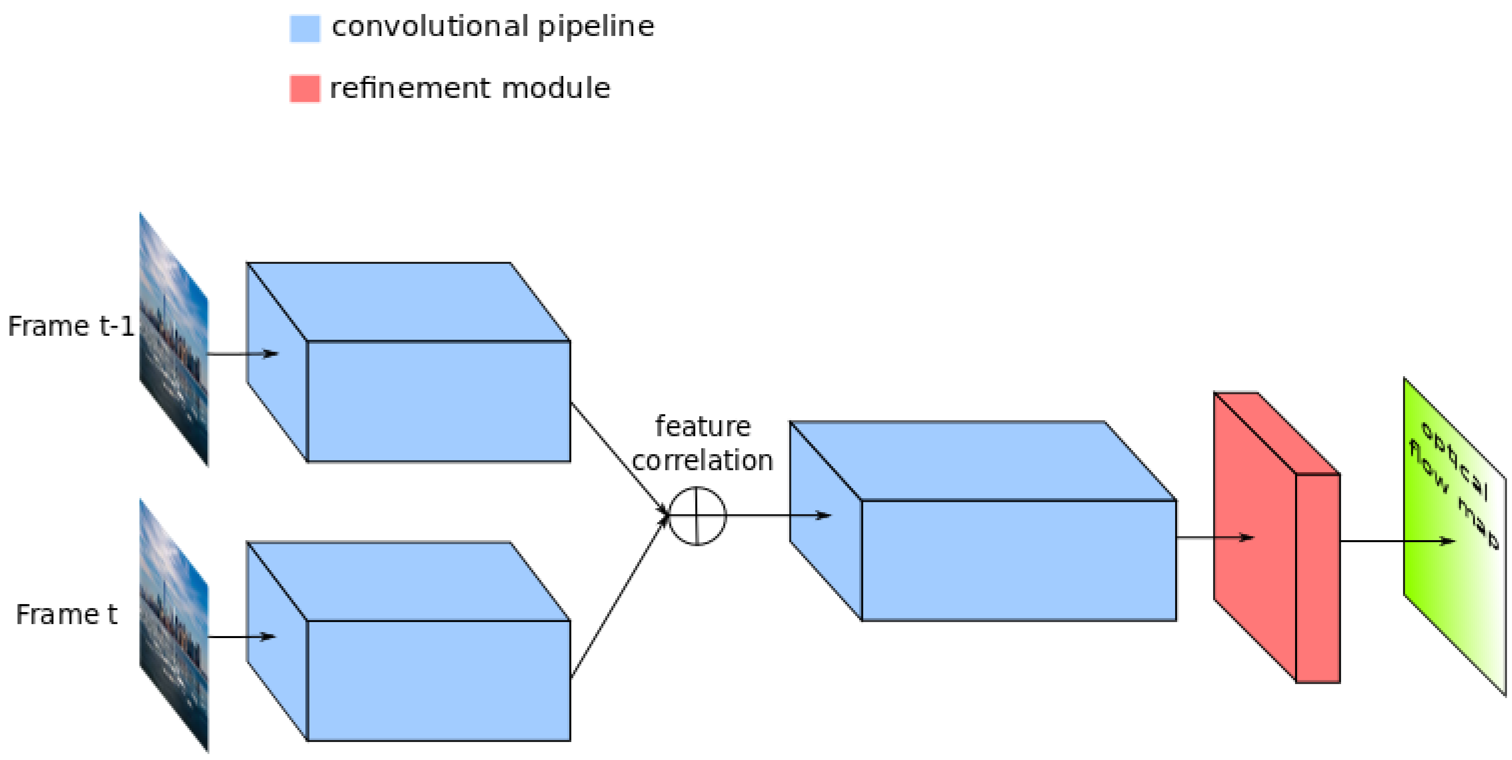
- Optical flow estimation by using ***Convolutional Neural Networks*** (***CNN***).
- High accuracy, dense flow field, fast implementations.
- Supervised methods:
  - Highest accuracy;
  - Ground truth for real world video sequences is required.
- Unsupervised methods:
  - Lower, but comparable accuracy;
  - No need for optical flow ground truth.

Neural Optical Flow estimation

Flownet: Supervised NN optical flow estimation.

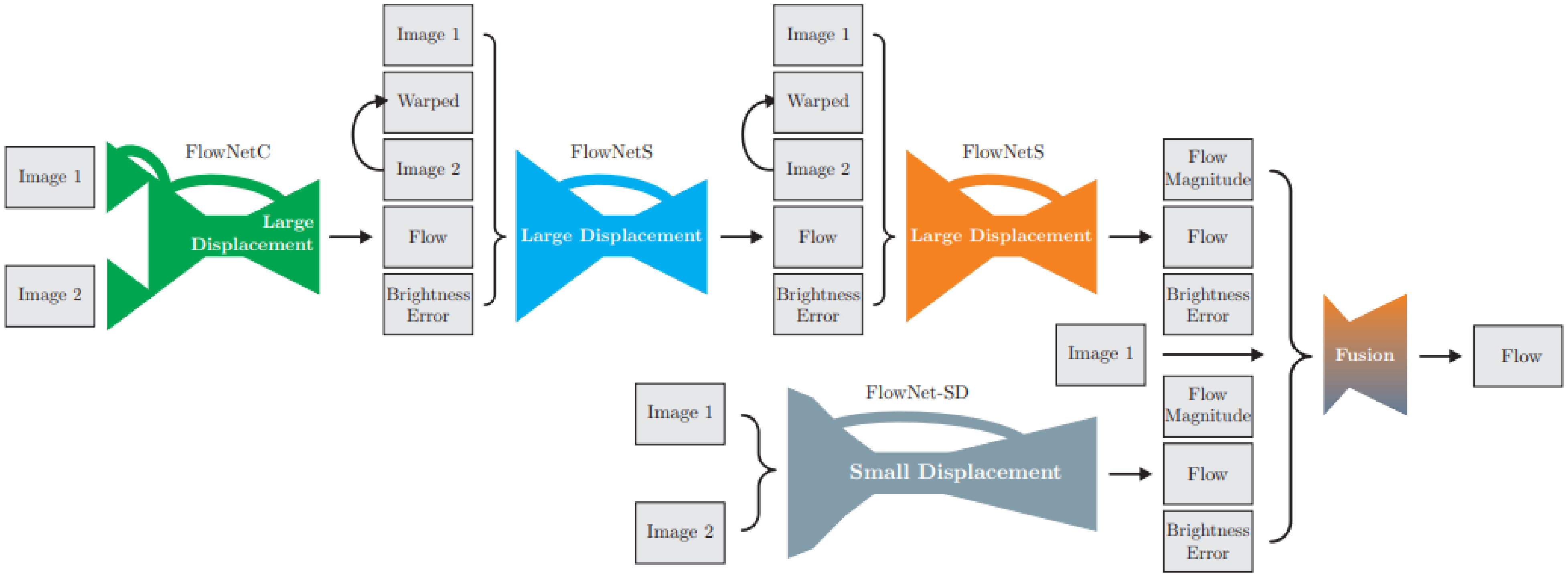
- Foundation stone for almost all later supervised networks.
- ***FlowNetS*** (**S**imple):
  - A single network branch.
  - Refinement module upscales conv6 output, using outputs from various intermediate stages.
  - Two consecutive input frames, concatenated in the channel dimension.

Neural Optical Flow estimation



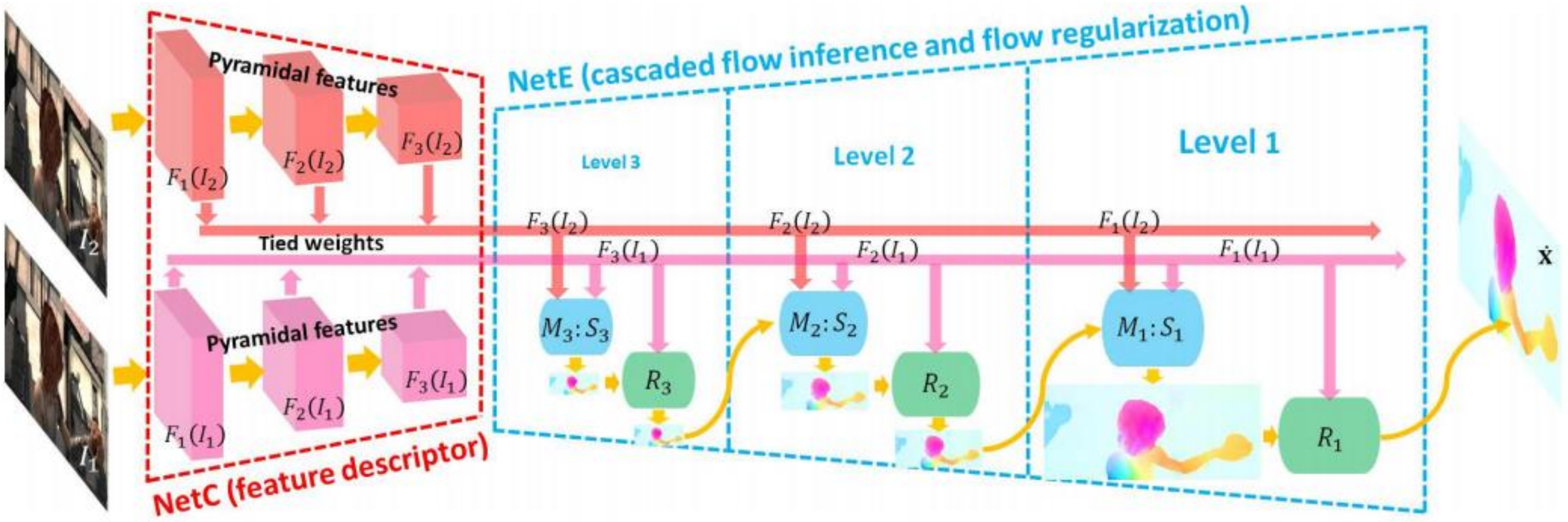
[DOS2015].

Neural Optical Flow estimation



FlowNet 2.0 [ILG2017].

Neural Optical Flow estimation



LightFlowNet. M: descriptor matching, S: sub-pixel refinement, R: a regularization module [HUI2018].

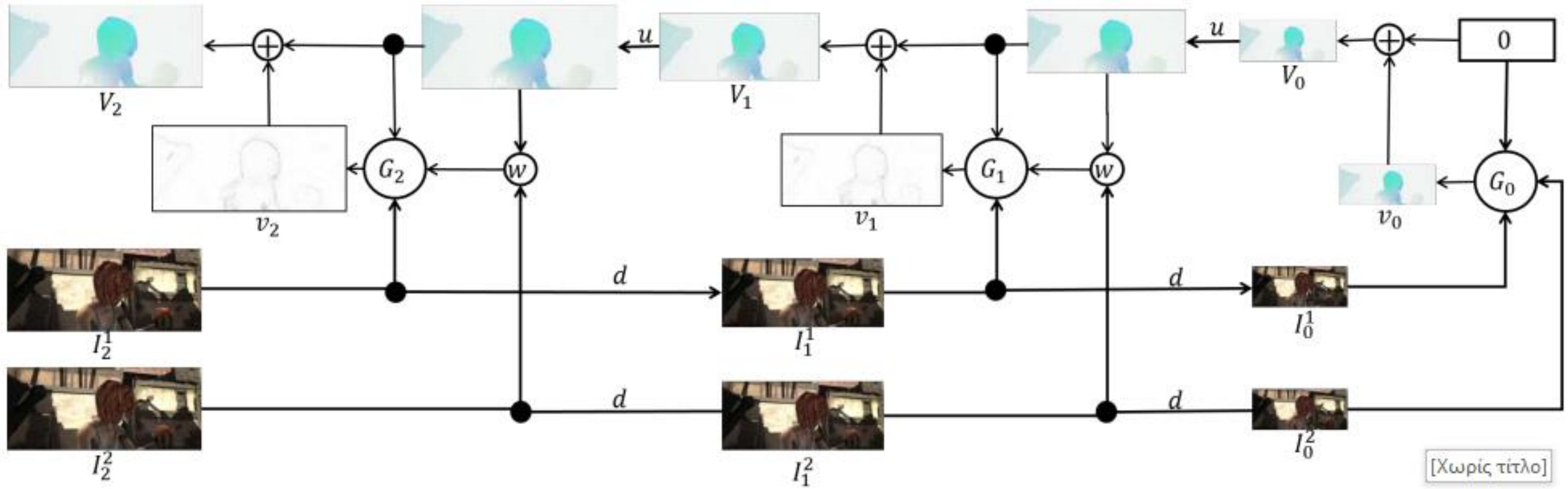
Neural Optical Flow estimation

SPyNet:

- 3-Level Pyramid Network.

- Better performance in many metrics than FlowNetC.
- More than twice as fast as FlowNetC.
- It uses the coarse-to-fine spatial pyramid structure to learn residual flow at each pyramid level.

Neural Optical Flow estimation



SPyNet 3-Level Pyramid Network [RAN2017].

Neural Optical Flow estimation

Qualitative comparison of neural optical flow estimators [RAN2017].

Object detection and Tracking

- Motion estimation estimates motion vectors on entire video frames.
- Object tracking relies on:
  - Object detection on a video frame.
  - Tracking of this object (essentially estimating its motion) over subsequent video frames.

Object Detection and Tracking

1st frame | 6th frame | 11th frame | 16th frame

- Problem statement:
  - To detect an object (e.g. human face) that appear in each video frame and localize its ***Region-Of-Interest*** (***ROI***).
  - To track the detected object over the video frames.

Bibliography

[PIT2017] I. Pitas, “Digital video processing and analysis” , China Machine Press, 2017 (in Chinese).

[PIT2013] I. Pitas, “Digital Video and Television” , Createspace/Amazon, 2013.

[PIT2021] I. Pitas, “Computer vision”, Createspace/Amazon, in press.

[NIK2000] N. Nikolaidis and I. Pitas, “3D Image Processing Algorithms”, J. Wiley, 2000.

[PIT2000] I. Pitas, “Digital Image Processing Algorithms and Applications”, J. Wiley, 2000.

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr