Chapter 10

Dense Motion Estimation

MRGCV Computer Vision

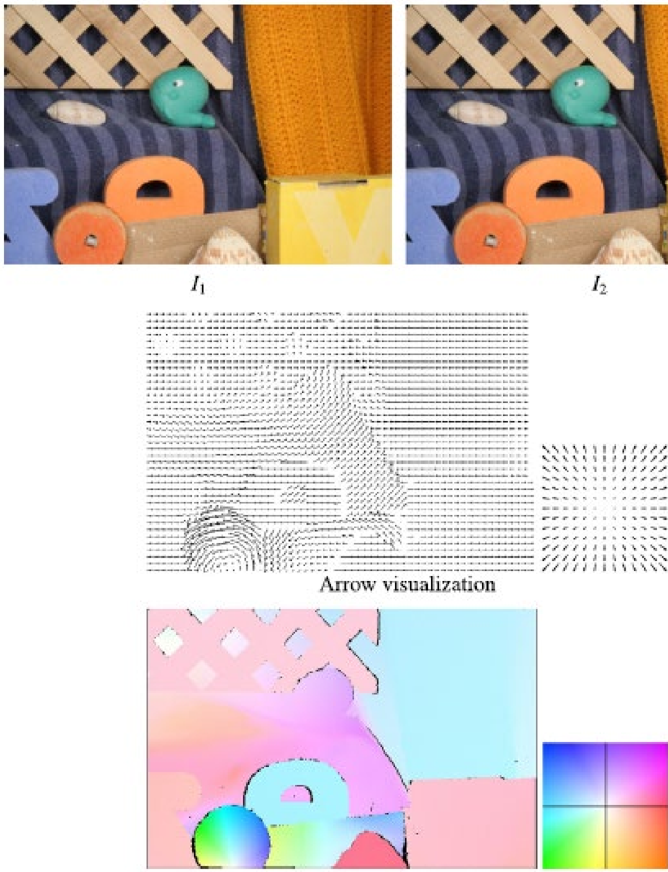
JM Martínez Montiel

Universidad Zaragoza

Dense Motion Estimation

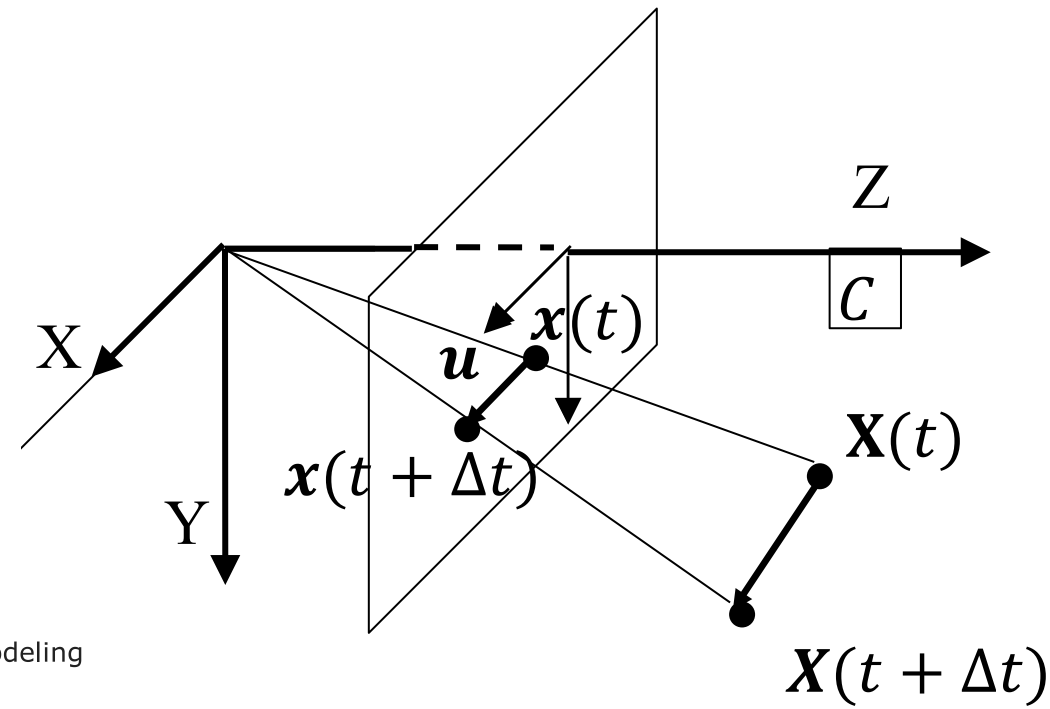
1. Introduction
2. Translational alignment
    1. Robust error metrics
    2. Bias and gain (exposure differences)
    3. Normalized Cross Correlation
3. Incremental refinement
    1. Bilinear interpolation
    2. Computing gradient by finite difference
    3. Conditioning and aperture problems
    4. Hierarchical motion estimation
4. Optical Flow, variational approach
    1. Horn-Shunk optical flow
    2. Data term
    3. Regularizers

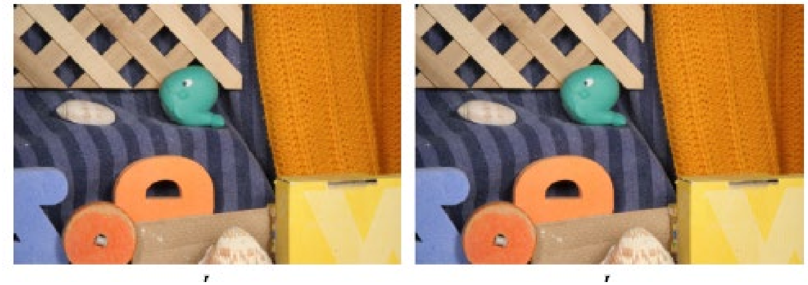
Dense motion estimation

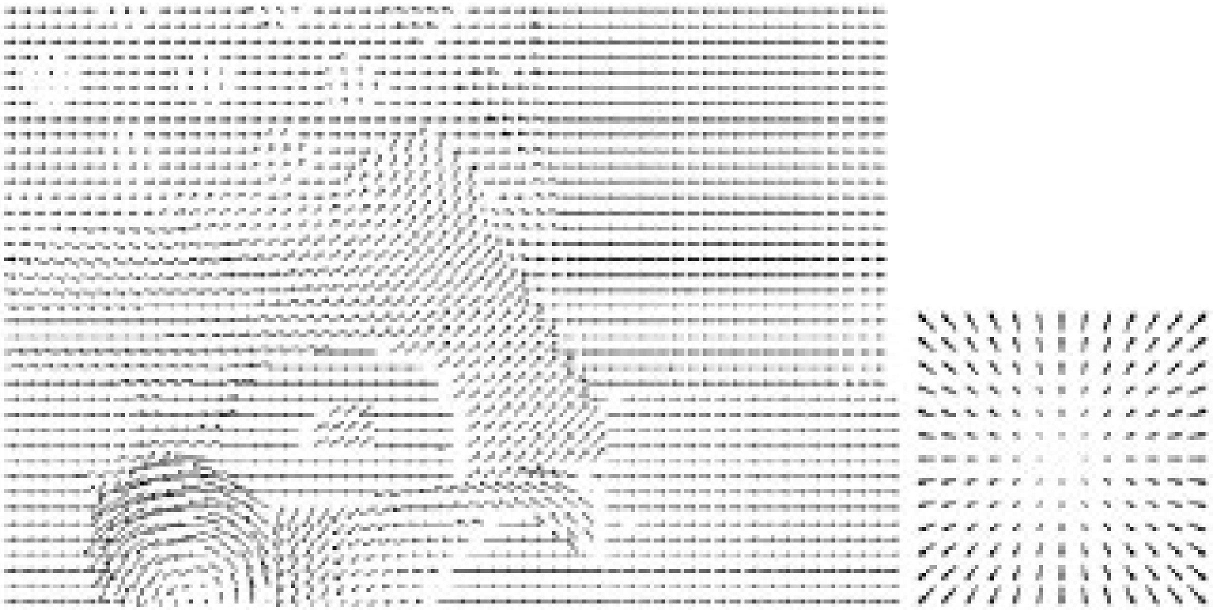


Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, *134*, 1-21.

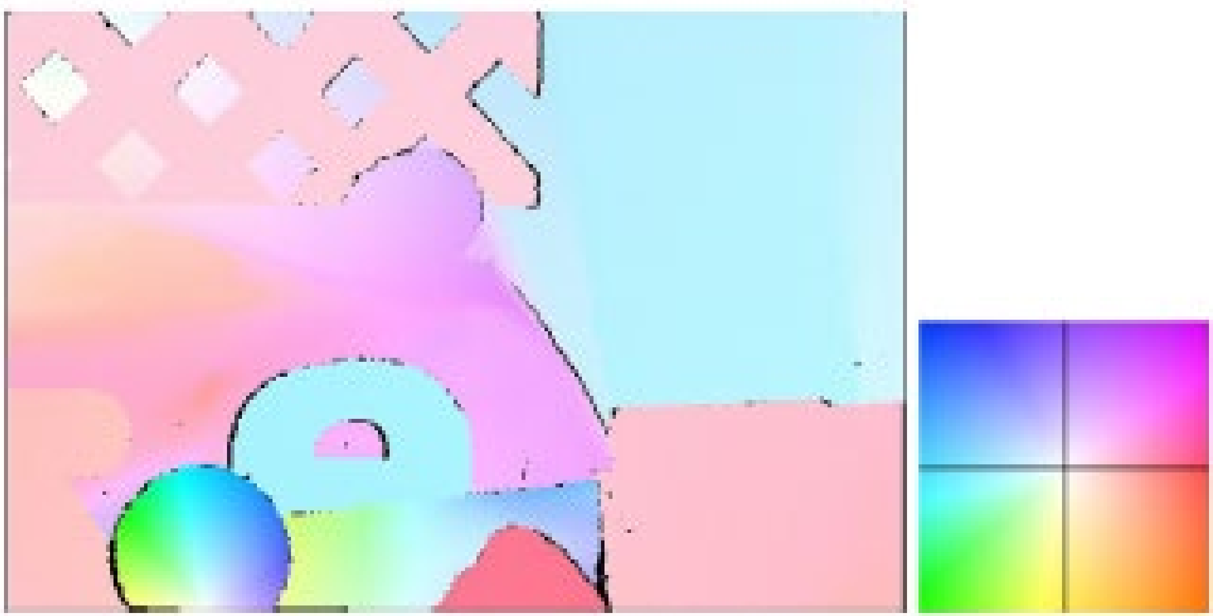
Motion Field, the projection on the image plane of the 3D motion in the scene







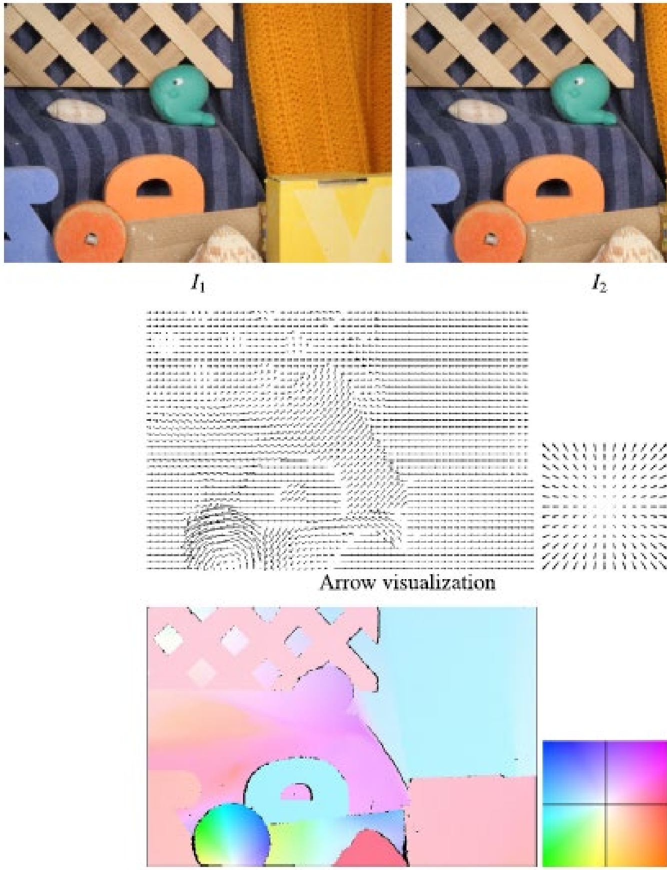
Arrow visualization



Color visualization

Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, *134*, 1-21.

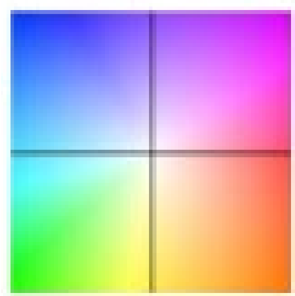
Dense motion estimation

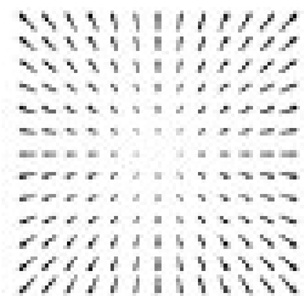


Motion Field, the projection on the image plane of the 3D motion in the scene

Per each image pixel, a motion vector $\boldsymbol{u}$

- $\boldsymbol{u}$ direction, colour hue
- $\boldsymbol{u}$ length, colour saturation

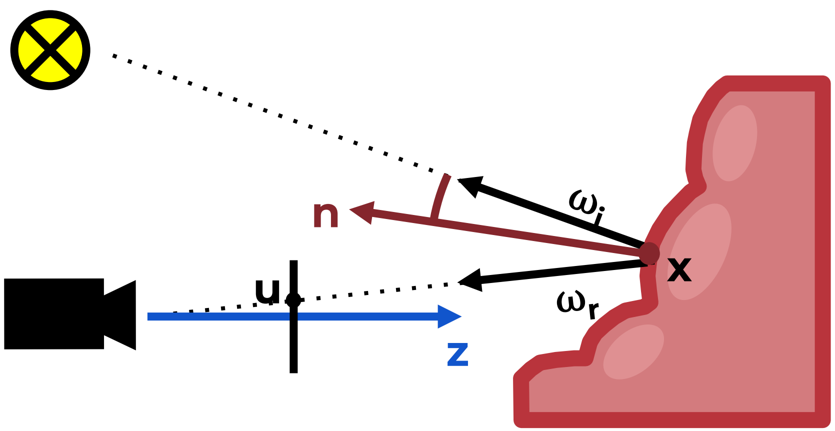




Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, *134*, 1-21.

2.Translational alignment

1. All the pixels $\boldsymbol{x}_i$ in a window have the same translational motion $\boldsymbol{u}$, a real 2D vector.
2. Subpixel accuracy!
3. Brightness constraint. Constant illumination, Lambertian reflectance.



$$I(\boldsymbol{x}) = \frac{\sigma}{d^2} k_d \cos\theta$$

Moving de camera does not affect radiance

Moving the light affects radiance:

Normal

Distance (squared decay)

2.Translational alignment

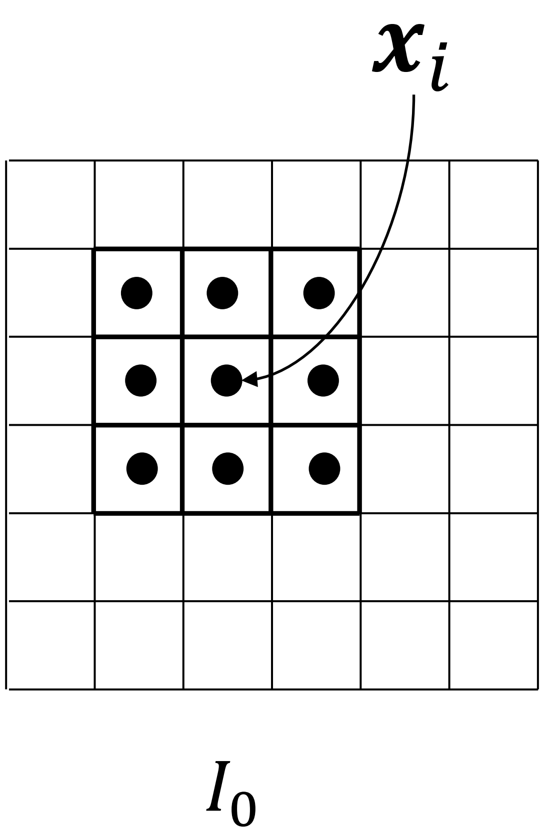
1. All the pixels $\boldsymbol{x}_i$ in a window have the same translational motion $\boldsymbol{u}$, a real 2D vector.
2. Subpixel accuracy!
3. Brightness constraint. Constant illumination, Lambertian reflectance.
4. Minimization of the Sum of the Squared Differences (SSD) of the pixels in the window gray level

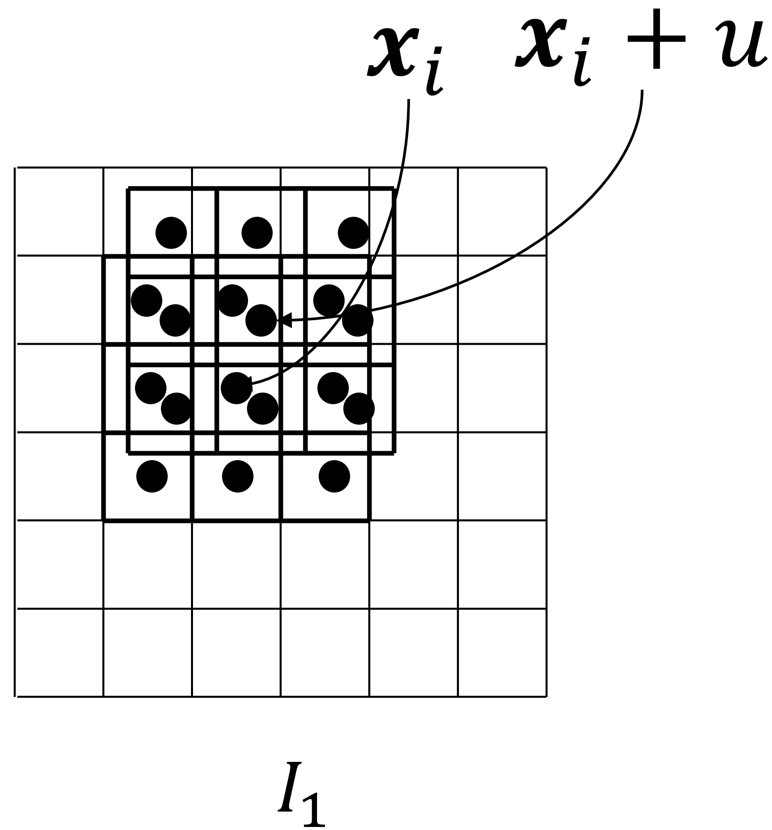
$$\underset{\boldsymbol{u}}{\operatorname{argmin}} E_{SSD}(\boldsymbol{u}) \qquad E_{SSD}(\boldsymbol{u}) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}) - I_0(\boldsymbol{x}_i)]^2$$

$\boldsymbol{x}_i$ i-pixel (integer values) pose in image I_0

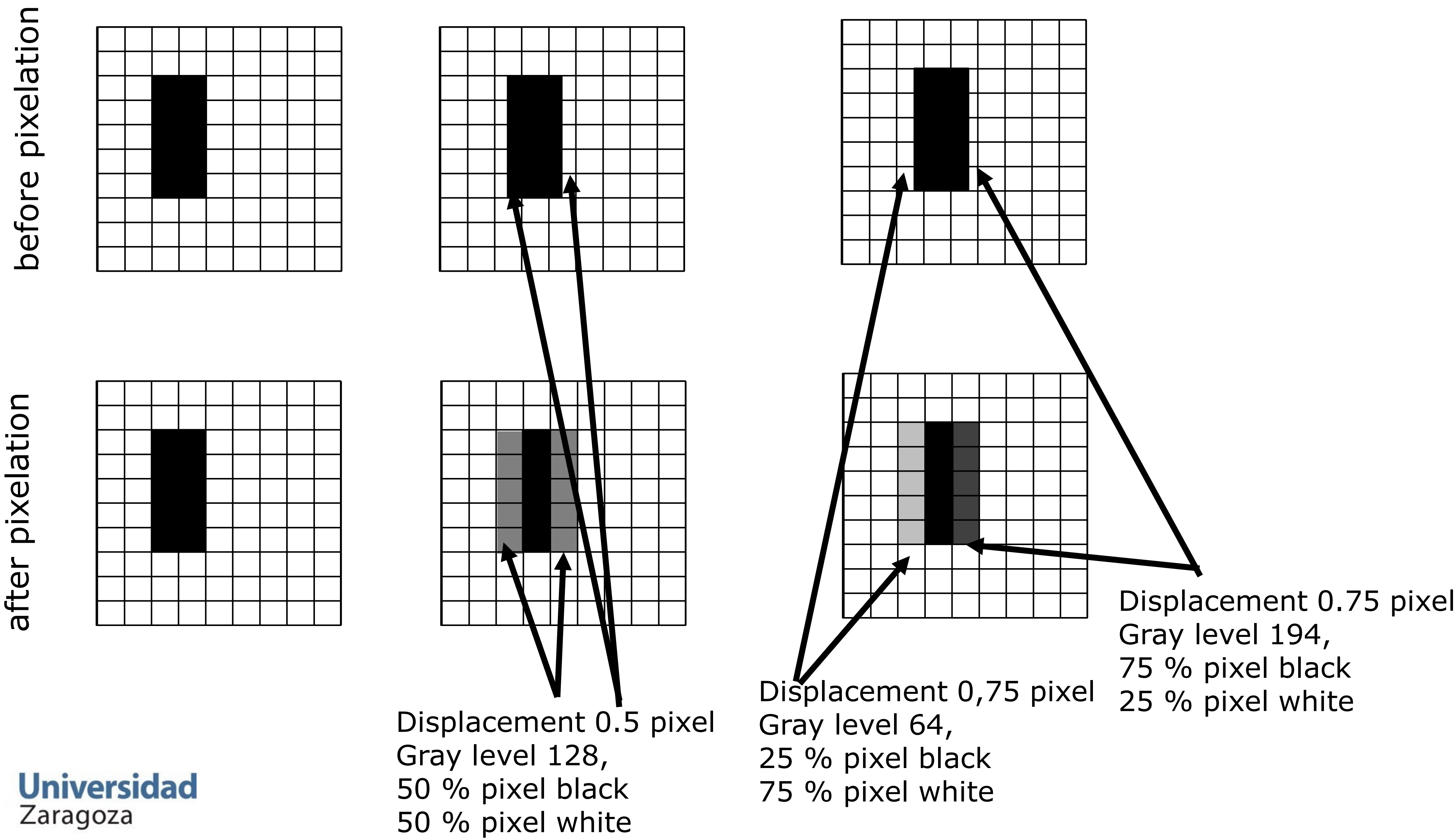
$\boldsymbol{u}$, motion real 2D vector

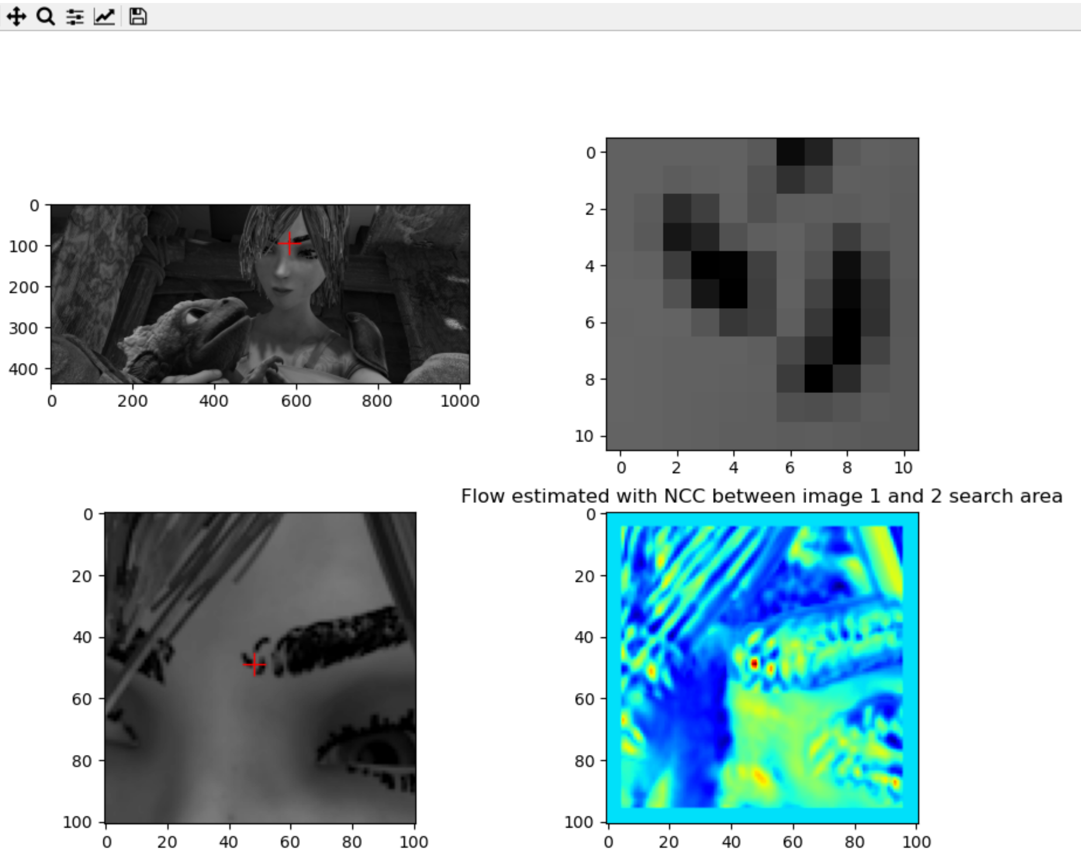
$I_1(\boldsymbol{x}_i + \boldsymbol{u})$ gray level at pixel $\boldsymbol{x}_i + \boldsymbol{u}$ (real values) in image I_1
values computed from bilinear, bicubic interpolation.


$\boldsymbol{x}_i$
I_0

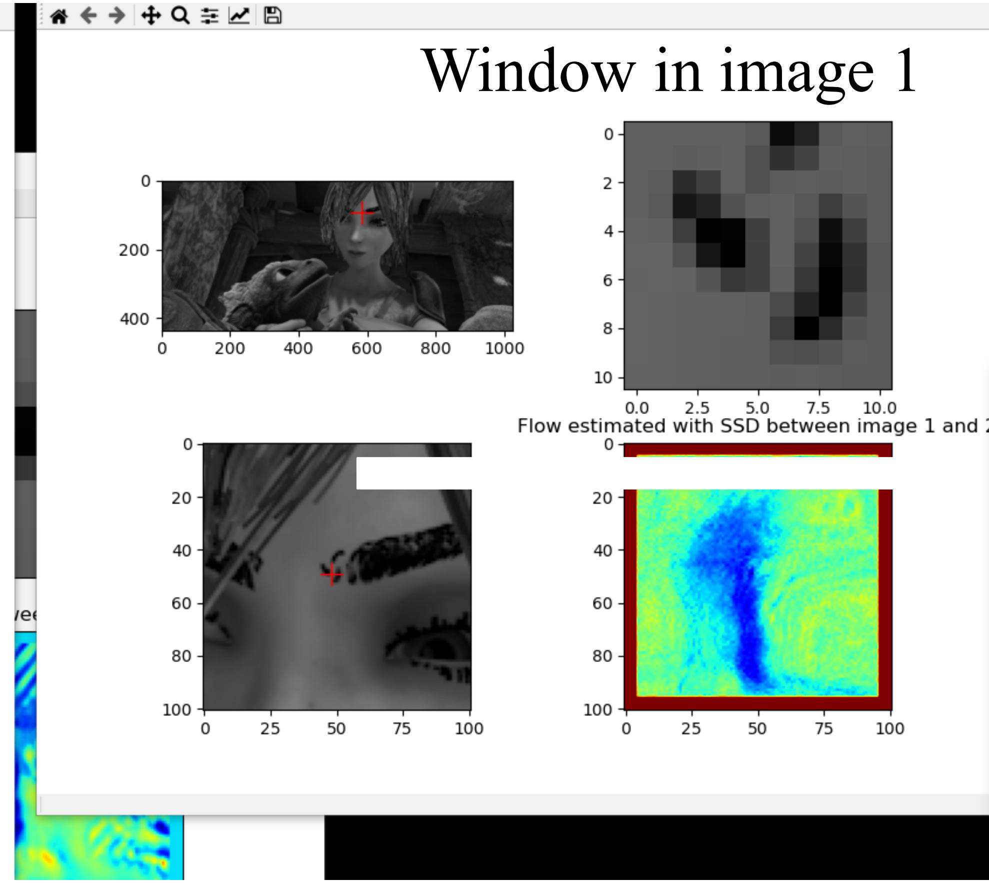

$\boldsymbol{x}_i$
$\boldsymbol{x}_i + u$
I_1

Photometric sub-pixel accuracy





Window in image 1



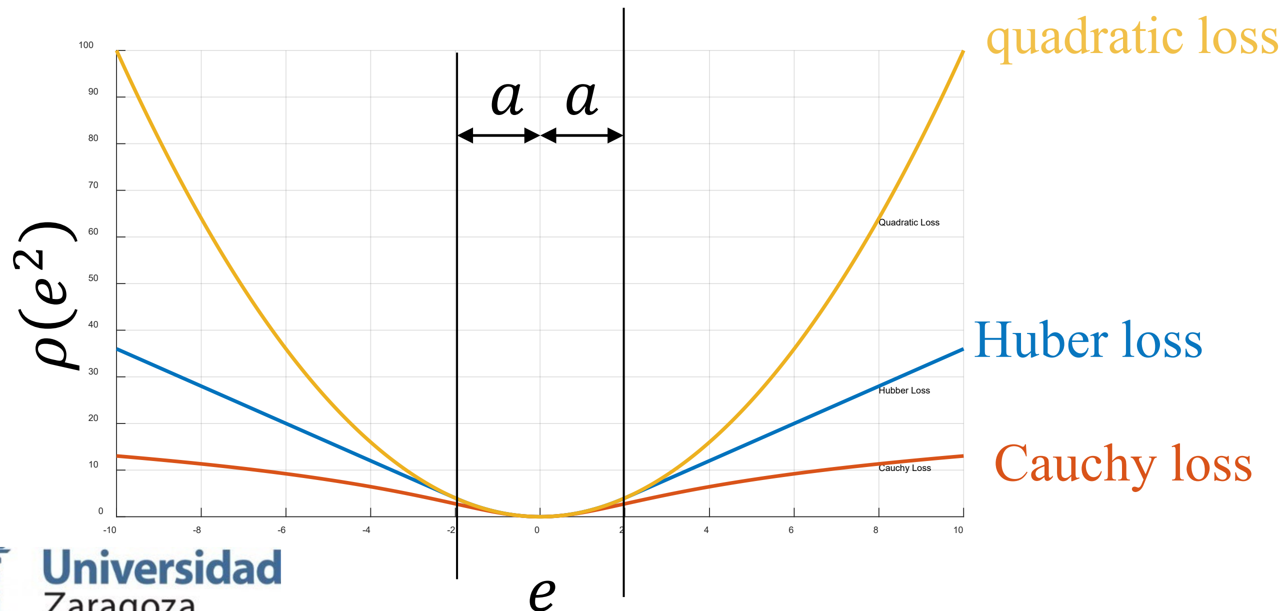
Two close-views of the same scene

Evaluation of the error for Different $\boldsymbol{u}$ displacements

2.1 Robust Error Metrics

$$\underset{\boldsymbol{u}}{\operatorname{argmin}}\, E_{SRD}(\boldsymbol{u}),\ E_{SRD}(\boldsymbol{u}) = \sum_i \rho([I_1(\boldsymbol{x}_i + \boldsymbol{u}) - I_0(\boldsymbol{x}_i)]^2)$$

1. Occlusions and specular reflections can produce outliers
2. As a LS fit, SSD is not robust with respect to outliers.
3. $\rho(\cdot)$, robust influence function to alleviate the effect of gross outliers, a robust threshold



Huber

$$\rho(s) = \begin{cases} s & s \le a^2 \\ 2a^2\sqrt{\dfrac{s}{a^2}} - 1 & s > a^2 \end{cases}$$

Cauchy

$$\rho(s) = a^2\left(\log\left(\frac{s}{a^2}\right) + 1\right)$$

2.Bias and Gain (exposure differences)

1. Images acquired different exposition due to differences in:
   1. shutter time
   2. aperture
   3. lighting
2. Affine model can cope with this changes, α, gain, β bias

$$I_1(\boldsymbol{x}_i + \boldsymbol{u}) = (1 + \alpha) I_0(\boldsymbol{x}_i) + \beta$$

$$\underset{\{\boldsymbol{u},\alpha,\beta\}}{\operatorname{argmin}} E_{BG}(\boldsymbol{u}, \alpha, \beta),\ E_{BG}(\boldsymbol{u}, \alpha, \beta) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}) - (1 + \alpha) I_0(\boldsymbol{x}_i) - \beta]^2$$

3. The translational alignment yields
   1. Translation vector $\boldsymbol{u}$
   2. Bias, gain of the affine intensity variation α, β

2.3 Normalized Cross Correlation(exposure differences)

1. Images acquired different exposition
2. Affine model can cope with this changes, α, gain, β bias

$$I_1(\boldsymbol{x}_i + \boldsymbol{u}) = (1+\alpha) I_0(\boldsymbol{x}_i) + \beta$$

3. Maximization of NCC avoids computing explicitly α, β

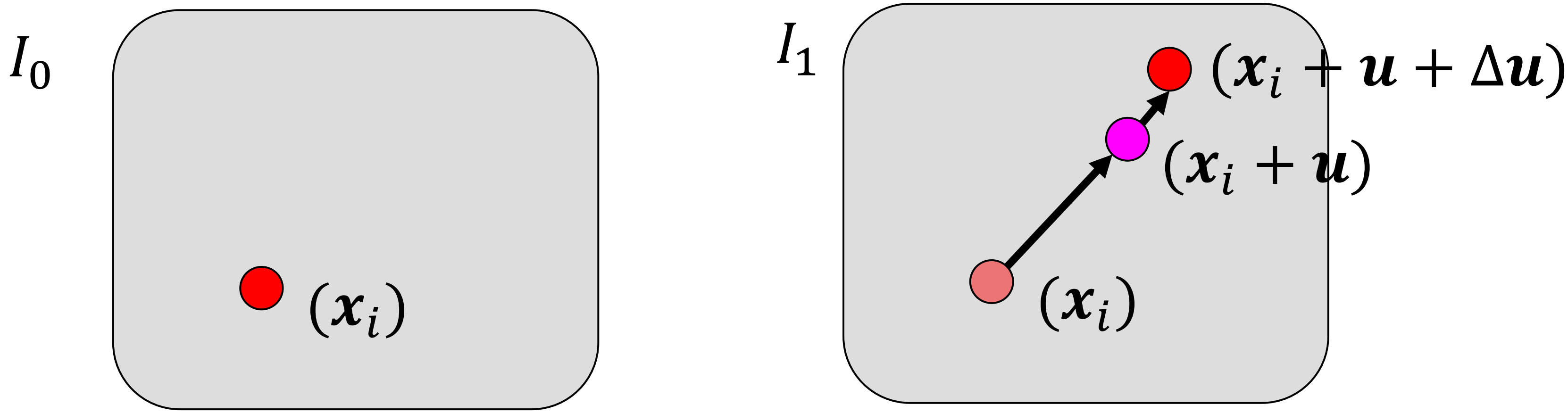
$$\underset{\{\boldsymbol{u}\}}{\operatorname{argmax}} E_{NCC}(\boldsymbol{u},),\ E_{NCC}(\boldsymbol{u}) = \frac{\sum_i [I_0(\boldsymbol{x}_i) - \bar{I}_0]\,[I_1(\boldsymbol{x}_i + \boldsymbol{u}) - \bar{I}_1]}{\sqrt{\sum_i [I_0(\boldsymbol{x}_i) - \bar{I}_0]^2}\sqrt{\sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}) - \bar{I}_1]^2}}$$

$$\bar{I}_0 = \frac{1}{N}\sum_i I_0(\boldsymbol{x}_i)$$

$$\bar{I}_1 = \frac{1}{N}\sum_i I_1(\boldsymbol{x}_i + \boldsymbol{u})$$

4. NCC always values between [-1,1], desirable scores, 0.7-1.0
5. Invalid in texture-less areas where the denominator goes to zero

Brightness constancy



$$I_1(\boldsymbol{x}_i + \boldsymbol{u} + \Delta\boldsymbol{u}) \approx I_1(\boldsymbol{x}_i + \boldsymbol{u}) + \boldsymbol{J_1}(\boldsymbol{x}_i + \boldsymbol{u})\,\Delta\boldsymbol{u}$$

$$\boldsymbol{J_1}(\boldsymbol{x}_i + \boldsymbol{u}) = \boldsymbol{\nabla} I_1(\boldsymbol{x}_i + \boldsymbol{u}) = \left(\frac{\partial I_1}{\partial x}, \frac{\partial I_1}{\partial y}\right)(\boldsymbol{x}_i + \boldsymbol{u}) = \left(I_x, I_y\right)(\boldsymbol{x}_i + \boldsymbol{u})$$

$$0 = I_1(\boldsymbol{x}_i + \boldsymbol{u} + \Delta\boldsymbol{u}) - I_0(\boldsymbol{x}_i) \approx I_1(\boldsymbol{x}_i + \boldsymbol{u}) + \boldsymbol{J_1}(\boldsymbol{x}_i + \boldsymbol{u})\,\Delta\boldsymbol{u} - I_0(\boldsymbol{x}_i)$$

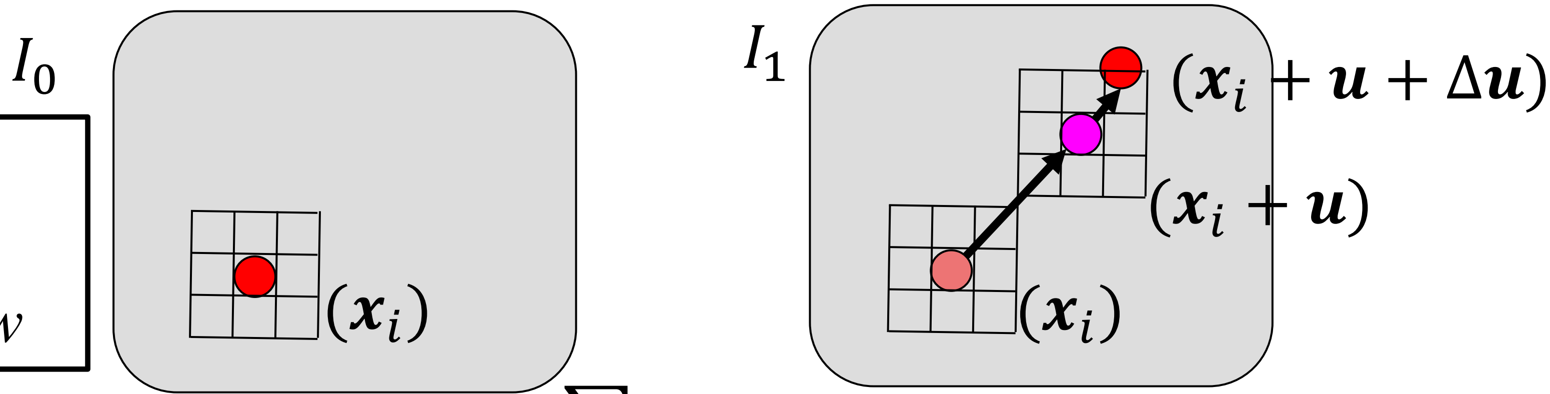
$$\Delta\boldsymbol{u}?$$

$e_i = I_t = I_1(\boldsymbol{x}_i + \boldsymbol{u})$- $I_0(\boldsymbol{x}_i)$ solve $\boldsymbol{J_1}(\boldsymbol{x}_i + \boldsymbol{u})\,\Delta\boldsymbol{u} + e_i = 0$

Per pixel: 2 unknowns, 1 equation

3. Lucas-Kanade. Incremental refinement

same motion
all pixels
in small window



$$E_{LK-SSD}(\boldsymbol{u}+\Delta\boldsymbol{u}) \approx \sum_i [\boldsymbol{J_1}(\boldsymbol{x}_i+\boldsymbol{u})\,\Delta\boldsymbol{u} + e_i]^2$$

$$\boldsymbol{J_1}(\boldsymbol{x}_i+\boldsymbol{u}) = \boldsymbol{\nabla} I_1(\boldsymbol{x}_i+\boldsymbol{u}) = \left(\frac{\partial I_1}{\partial x}, \frac{\partial I_1}{\partial y}\right)(\boldsymbol{x}_i+\boldsymbol{u}) = \left(I_x, I_y\right)(\boldsymbol{x}_i+\boldsymbol{u})$$

$$e_i = I_1(\boldsymbol{x}_i+\boldsymbol{u}) - I_0(\boldsymbol{x}_i) = I_t \qquad \Delta\boldsymbol{u} = (u, v)$$

Solving a least squares problem which yields the next linear system: $\boldsymbol{A}\Delta\boldsymbol{u} = \boldsymbol{b}$

$$\boldsymbol{A} = \sum_i \boldsymbol{J}_1^T(\boldsymbol{x}_i+\boldsymbol{u})\boldsymbol{J_1}(\boldsymbol{x}_i+\boldsymbol{u}) = \begin{bmatrix} \sum_i I_x^2 & \sum_i I_x I_y \\ \sum_i I_x I_y & \sum_i I_y^2 \end{bmatrix}$$

$$\boldsymbol{b} = -\sum_i e_i \boldsymbol{J}_1^T(\boldsymbol{x}_i+\boldsymbol{u}) = -\begin{bmatrix} \sum_i I_x I_t \\ \sum_i I_y I_t \end{bmatrix}$$

Lucas, B. D., Kanade, T. (1981). An Iterative Image Registration Technique with an Application to Stereo Vision. *Proceedings DARPA image Workshop*.

3. Incremental refinement

while $\|\Delta \boldsymbol{u}\| < \epsilon$

compute $I_1(\boldsymbol{x}_i + \boldsymbol{u})$; might imply image wrapping

compute $(I_x, I_y)(\boldsymbol{x}_i + \boldsymbol{u}) = \boldsymbol{J}_1(\boldsymbol{x}_i + \boldsymbol{u})$

compute $e_i = I_1(\boldsymbol{x}_i + \boldsymbol{u}) - I_0(\boldsymbol{x}_i) = I_t$

compute $\boldsymbol{A}$

compute $\boldsymbol{b}$

solve $\boldsymbol{A}\Delta\boldsymbol{u} = \boldsymbol{b}$

update $\boldsymbol{u} = \boldsymbol{u} + \Delta\boldsymbol{u}$

end

$\boldsymbol{x}_i$, dim 2 vector, integer values
$\boldsymbol{u}$ dim 2 vector, float values
$\Delta\boldsymbol{u}$ dim 2 vector, float values

For efficiency to avoid recomputing $\boldsymbol{J}_1(\boldsymbol{x}_i + \boldsymbol{u})$ at each iteration

$$\boldsymbol{J}_1(\boldsymbol{x}_i + \boldsymbol{u}) \approx \boldsymbol{J}_0(\boldsymbol{x}_i)$$

Can be extended to consider:

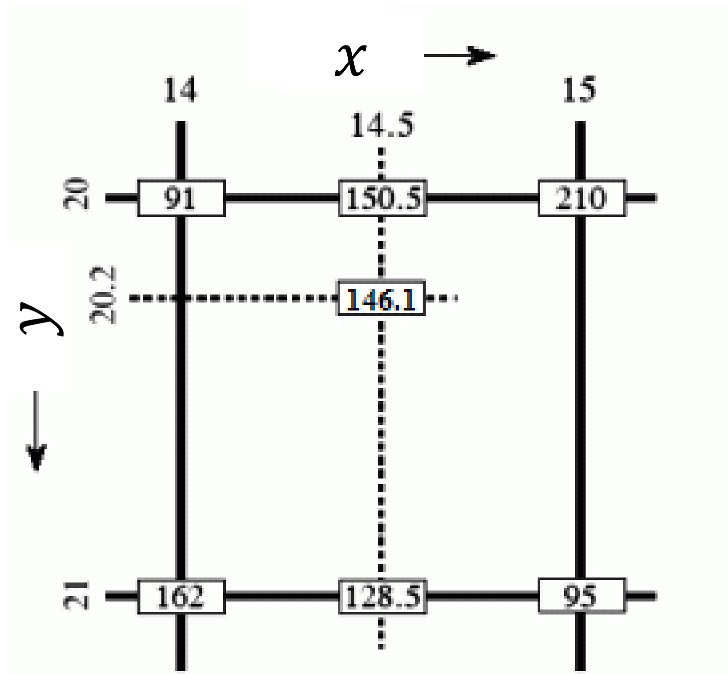
1. gain and bias
2. robust influence functions

Shree Nayar. Lucas-Kanade Method Optical Flow https://youtu.be/6wMoHgpVUn8?si=FRNApqBO7DKHSfRN
Shree Nayar. Coarse-to-Fine Flow Estimation Optical Flow https://youtu.be/VSSyPskheaE?si=O40Qwp0OYe6fcVod
Shree Nayar. Application of Optical Flow Optical Flow https://youtu.be/_6OJcLGuRQw?si=coZjDZ2raoDlpDnY

3.1 Bilinear Interpolation

$$I(x+\Delta x, y+\Delta y) \approx \begin{bmatrix} 1-\Delta x & \Delta x \end{bmatrix} \begin{bmatrix} I(x,y) & I(x,y+1) \\ I(x+1,y) & I(x+1,y+1) \end{bmatrix} \begin{bmatrix} 1-\Delta y \\ \Delta y \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} 91 & 162 \\ 210 & 95 \end{bmatrix} \begin{bmatrix} 0.8 \\ 0.2 \end{bmatrix} = 146.1$$



By Bilin2.png: Jayson Kostelykderivative work: Vizerai (talk) - This file was derived from: Bilin2.png, Public Domain, https://commons.wikimedia.org/w/index.php?curid=12527191

3.2 Computing gradient by finite difference

Forward difference

$$I_x(x,y) \approx I(x+1,y)\text{-}I(x,y)$$

$$I_y(x,y) \approx I(x,y+1)\text{-}I(x,y)$$

Backward difference

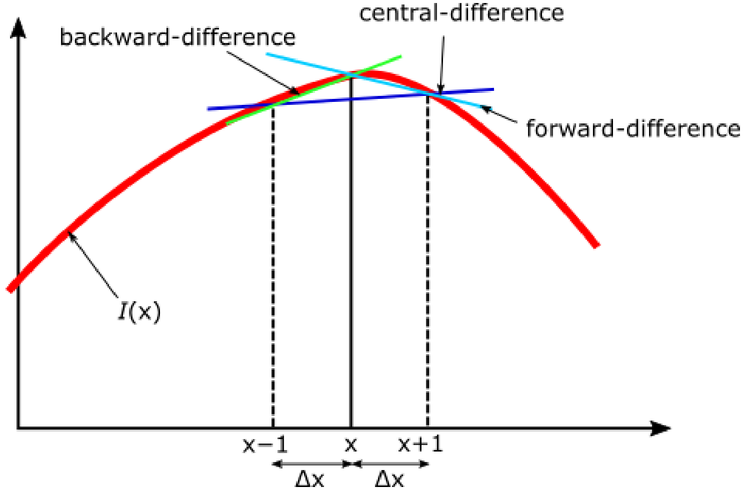
$$I_x(x,y) \approx I(x,y) - I\,(x-1,y)$$

$$I_y(x,y) \approx I(x,y) - I(x-1,y)$$

Central difference

$$I_x(x,y) \approx \frac{I(x+1,y) - I(x-1,y)}{2}$$

$$I_y(x,y) \approx \frac{I(x,y+1) - I(x,y-1)}{2}$$



By Kakitc - Own work, CC BY-SA 4.0,
https://commons.wikimedia.org/w/index.php?curid=63327976

3.3 Conditioning and aperture problems

The linear system

$$\boldsymbol{A}\Delta\boldsymbol{u} = \boldsymbol{b}$$

Only well conditioned if $\lambda_1 \geq \lambda_2$, eignevalues of $\boldsymbol{A}$ are not close to 0

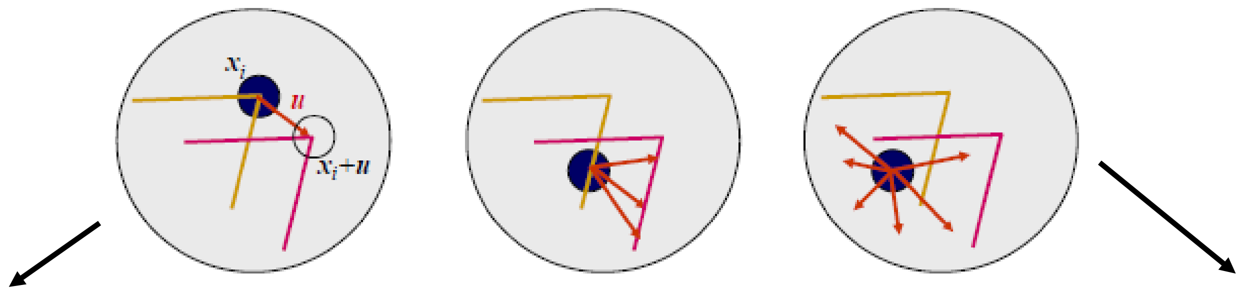
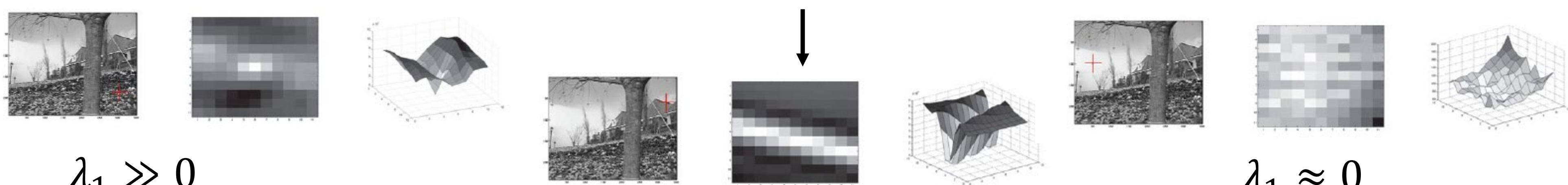


Fig 8.3, 8.4
Szeliski 2010



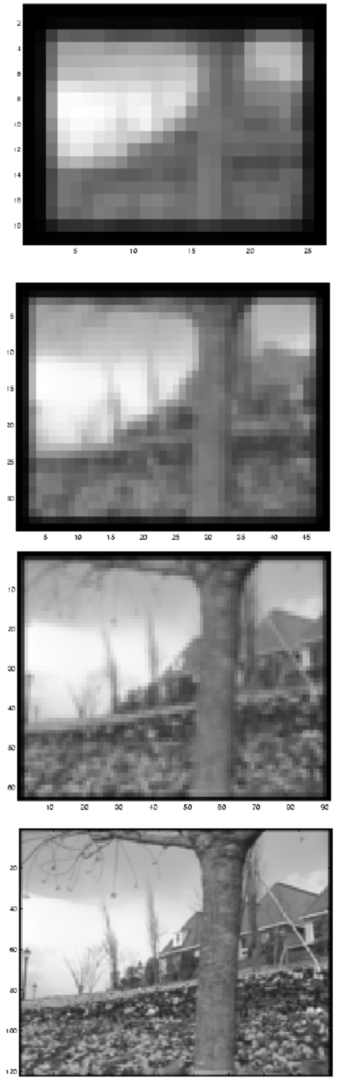
$\lambda_1 \gg 0$
$\lambda_2 \gg 0$

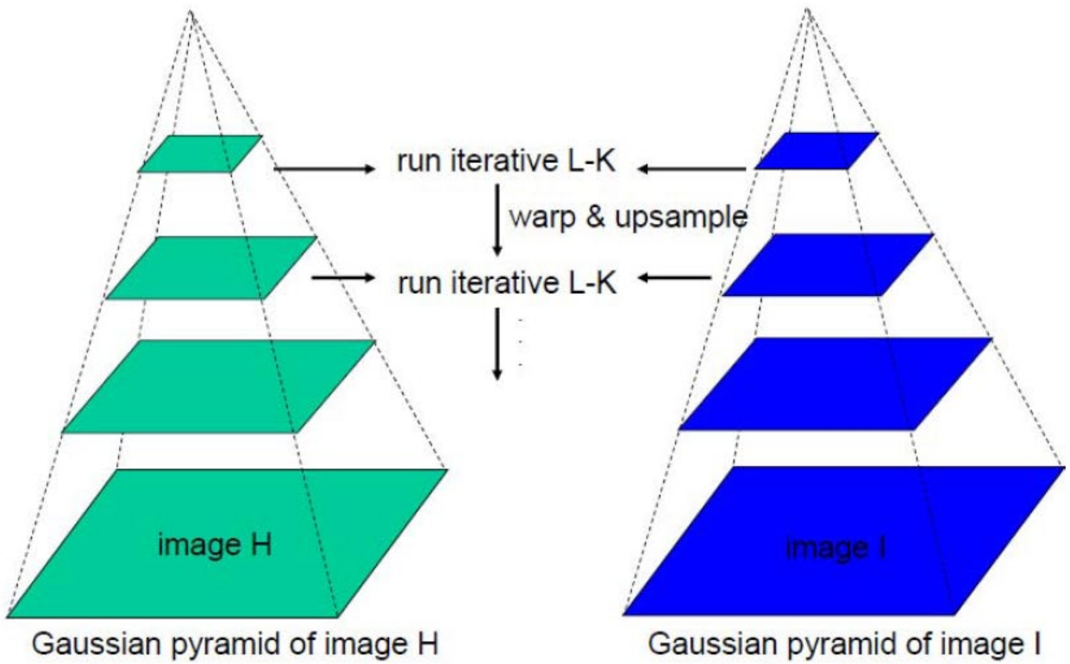
$\lambda_1 \gg 0$
$\lambda_2 \approx 0$

$\lambda_1 \approx 0$
$\lambda_2 \approx 0$

J. Shi and C. Tomasi, "Good features to track," *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 1994, pp. 593-600, doi: 10.1109/CVPR.1994.323794.

3.4 Hierarchical motion estimation





Cordelia Schmid. Optical Flow Tutorial.
https://lear.inrialpes.fr/~verbeek/mlor.slides.16.17/optical_flow.pdf

3.4 Hierarchical motion estimation

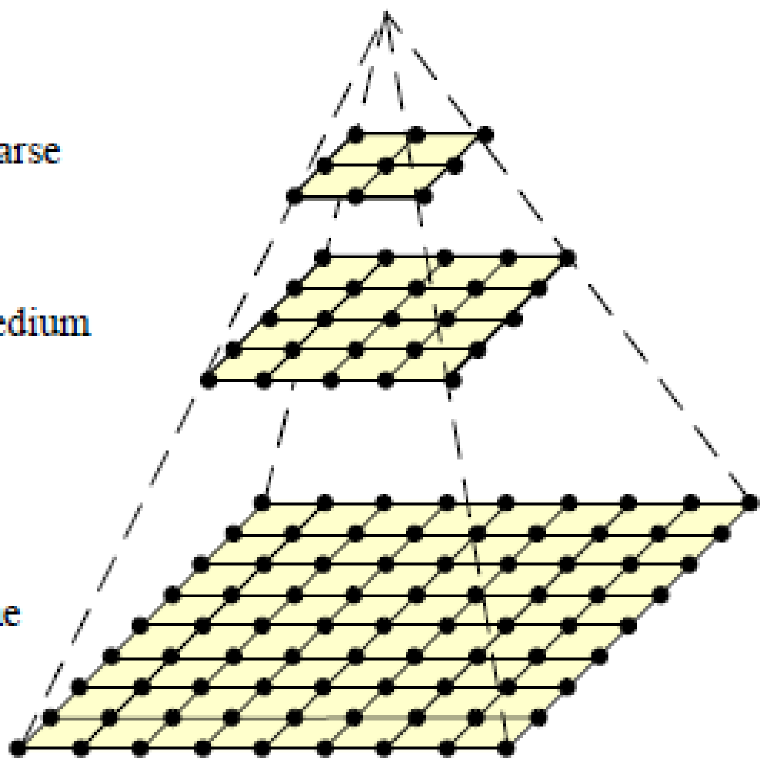
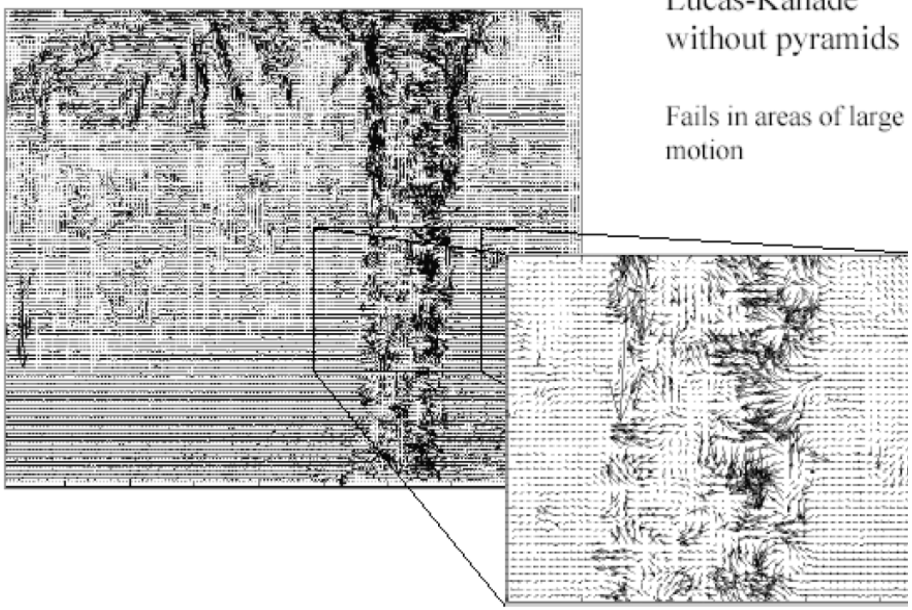


Fig 8.3, 8.4
Szeliski 2010

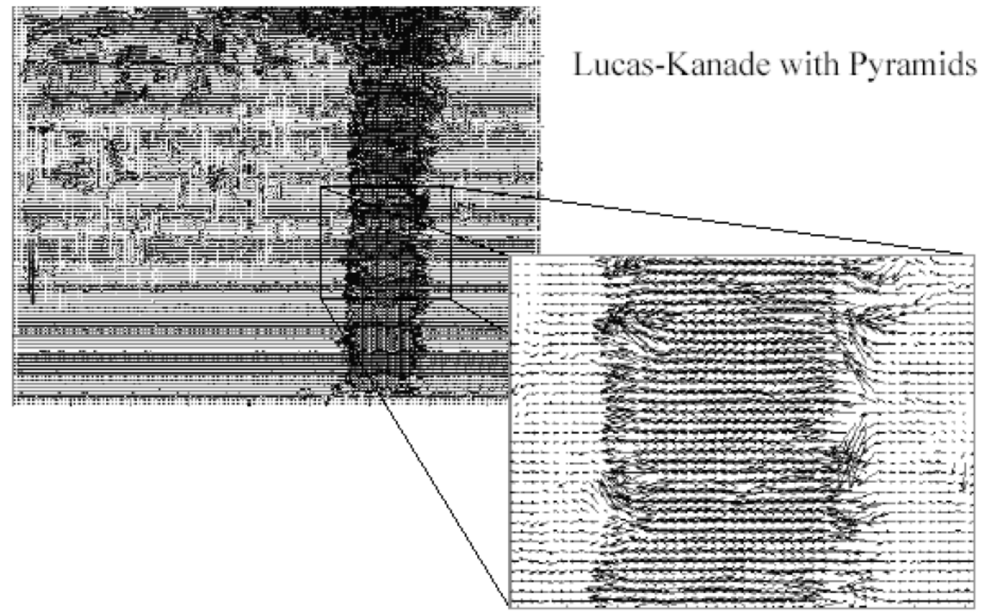
1. Exhaustive search in the coarsest level (l)
2. LK incremental refinement → $\boldsymbol{u}^{(l)}$
3. Prediction at the next level
$$\hat{\boldsymbol{u}}^{(l-1)} = 2\boldsymbol{u}^{(l)}$$
4. Exhaustive search in small window around the prediction
5. LK incremental refinement → $\boldsymbol{u}^{(l-1)}$

3.4 Hierarchical motion estimation



Lucas-Kanade without pyramids

Fails in areas of large motion



Lucas-Kanade with Pyramids

Cordelia Schmid. Optical Flow Tutorial.
https://lear.inrialpes.fr/~verbeek/mlor.slides.16.17/optical_flow.pdf

Dense Motion. Optical Flow

- 2D projection of the 3D scene motion.
- Computed for every single pixel of the scene.

$$\operatorname*{argmin}_{\boldsymbol{u}} E_{SSD}(\boldsymbol{u}) \qquad E_{SSD}(\boldsymbol{u}) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}) - I_0(\boldsymbol{x}_i)]^2$$

All the pixels $\{\boldsymbol{x}_i\}$in a **window**, have the **same** motion $\boldsymbol{u}$

$$\operatorname*{argmin}_{\{\boldsymbol{u}_i\}} E_{SSD-OF}(\{\boldsymbol{u}_i\}) \qquad E_{SSD-OF}(\{\boldsymbol{u}_i\}) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}_i) - I_0(\boldsymbol{x}_i)]^2$$

Each pixel $\boldsymbol{x}_i$ of the **image** has a **different** motion $\boldsymbol{u}_i$

Aperture problem

$$\underset{\boldsymbol{u}}{\operatorname{argmin}}\, E_{SSD}(\boldsymbol{u}) \qquad E_{SSD}(\boldsymbol{u}) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}) - I_0(\boldsymbol{x}_i)]^2$$

All the pixels $\{x_i\}$in a **window**, have the **same** motion $\boldsymbol{u}$

- \# equations, pixels in the window, e.g 25 in a 5x5 window
- \# unknowns, 2, one per each component of the $\boldsymbol{u}$ vector

$$\underset{\{\boldsymbol{u}_i\}}{\operatorname{argmin}}\, E_{SSD-OF}(\{\boldsymbol{u}_i\}) \qquad E_{SSD-OF}(\{\boldsymbol{u}_i\}) = \sum_i [I_1(\boldsymbol{x}_i + \boldsymbol{u}_i) - I_0(\boldsymbol{x}_i)]^2$$

Each pixel $\boldsymbol{x}_i$ of the image has a different motion $\boldsymbol{u}_i$

- \# equations, 1 per pixel in the image
- \# unknowns, 2 per pixel in the image,
  2 per each $\boldsymbol{u}_i$ 2-compoents vector
- \# double number of unknown than equations
  under constrained
  aperture problem

Continuous Modelling

$$I\colon \Omega \times T \longrightarrow \mathcal{R}$$
$$(x, y, t) \rightsquigarrow I(x, y, t) = I(\boldsymbol{x}, t)$$

In corresponding pixels the grey level do no change

$$I(x + u, y + v, t + 1) - I(x, y, t) = 0$$

Linearizing the equation

$$I(x + u, y + v, t + 1) \approx I(x, y, t) + I_x(x, y, t)u + I_y(x, y, t)v + I_t(x, y, t)1$$

$$I(x + u, y + v, t + 1) - I(x, y, t) = 0 \Longrightarrow \boxed{I_x u + I_y v + I_t = 0}$$

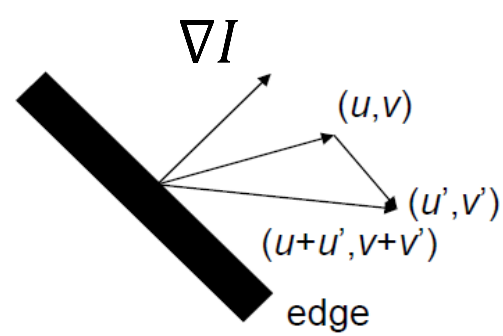
Brightness Constancy Constraint Equation BCCE

The Aperture Problem

- The BCCE provides at most 1 equation for unknowns
- Motion, ill posed problem

$$I_x u + I_y v + I_t = 0 \qquad \begin{pmatrix} I_x & I_y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + I_t = 0 \qquad \nabla I \begin{pmatrix} u \\ v \end{pmatrix} + I_t = 0$$

- Only the flow component perpendicular to image gradient can be computed, so-called normal flow
- If $\begin{pmatrix} u \\ v \end{pmatrix}$ satisfies BCCE equation so does $\begin{pmatrix} u + u' \\ v + v' \end{pmatrix}$ if $\nabla I \begin{pmatrix} u' \\ v' \end{pmatrix}$=0
- If $\nabla I \begin{pmatrix} u \\ v \end{pmatrix} \approx 0$ motion cannot be computed either poorly textured areas, provide no information



Cordelia Schmid. Optical Flow Tutorial.
https://lear.inrialpes.fr/~verbeek/mlor.slides.16.17/optical_flow.pdf

Variational Approach

- Compute the motion field $\boldsymbol{u}$ as a minimizer of a suitable energy functional:

$$E(\boldsymbol{u}) = \int_{\Omega} D(\boldsymbol{u}) + \lambda\, S(\boldsymbol{u}) dxdy$$

- **Data term**, $D(\boldsymbol{u})$, penalizes deviations from BCCE
- **Smoothness (regularization) term**, $S(\boldsymbol{u})$, penalizes deviations from smoothness in the motion field
- ***Regularization weight*** $\lambda > 0$ determines the degree of smoothness
- The solution $\boldsymbol{u}$ fit best the model assumptions, compromise between contradictory assumptions.

Horn-Schunk method

$$E(\boldsymbol{u}) = \int_{\Omega} \left(I_x u + I_y v + I_t\right)^2 + \lambda \left(|\nabla u|^2 + |\nabla v|^2\right) dxdy$$

Horn, B. K., and Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence*, *17*(1-3), 185-203

- Data term penalizes deviations from BCCE
- Smoothness term penalizes deviation from smoothness i.e. variations of u, v given by they first derivatives

Data Term

In the literature variations around the BCCE to boost performance

- Quadratic error function + robust influence
- Filtering
- NCC in a patch around the pixel
- …

Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, *134*, 1-21. http://bigwww.epfl.ch/publications/fortun1501.pdf

Regularization

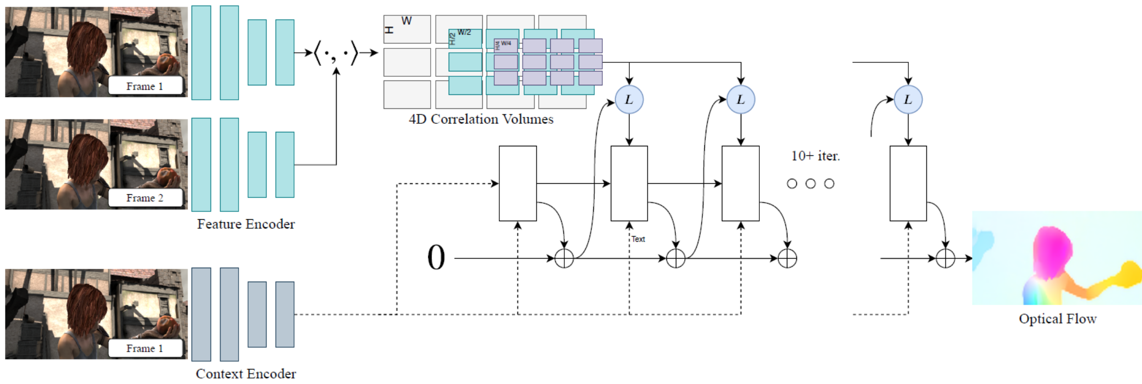
- Horn-Shrunk L_2 norm on the motion field gradient
$$|\nabla u|^2 + |\nabla v|^2$$
- Total Variation TV L_1norm on the motion field gradient
$$|\nabla u| + |\nabla v|$$
  - Preserves the motion field discontinuities in the occluding boundaries.
  - Efficient in GPU algorithms

- Inclusion a weight with image gradient information
$$e^{-\|\nabla I(\boldsymbol{x})\|^2/\zeta}$$
  - Reduces the regularization in the image borders
  - Reduce the regularization in textured areas

Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding*, *134*, 1-21. http://bigwww.epfl.ch/publications/fortun1501.pdf

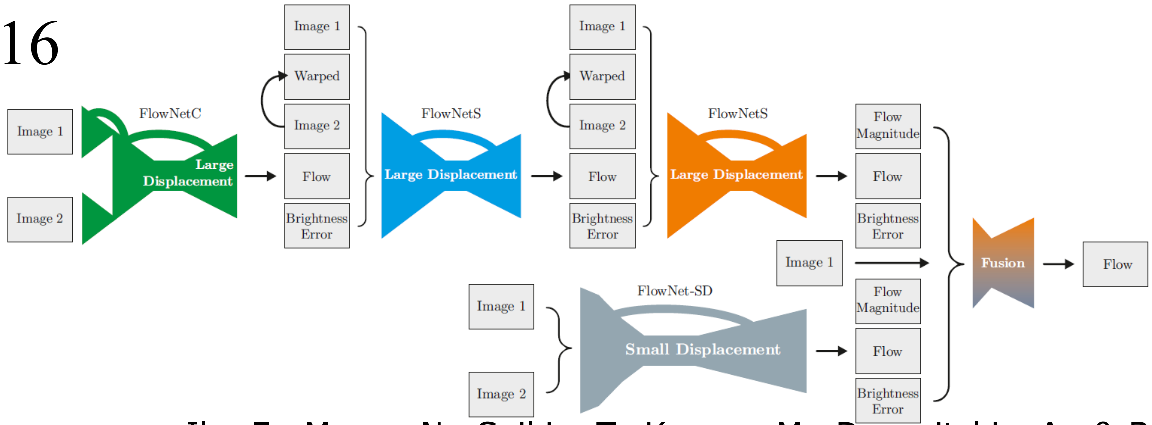
Learning Methods in Optical Flow

RAFT, 2020



Teed, Z., & Deng, J. (2020, August). Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision* (pp. 402-419). Springer, Cham.

FlowNet2.0 2016



Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). Flownet 2.0: Evolution of optical flow estimation with deep networks. CVPR (pp. 2462-2470).

Universidad Zaragoza 1542

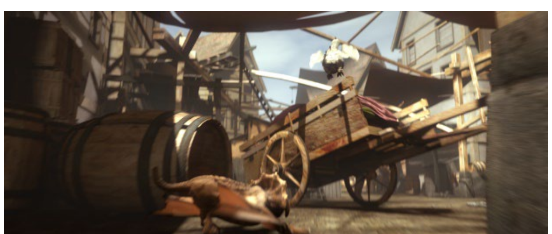
Training on synthetic datasets

Flying Chairs

Dosovitskiy,et al. Flownet: Learning optical flow with convolutional networksCVPR 2015

FlyingThings3D

Mayer, N et al. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. CVPR 2016



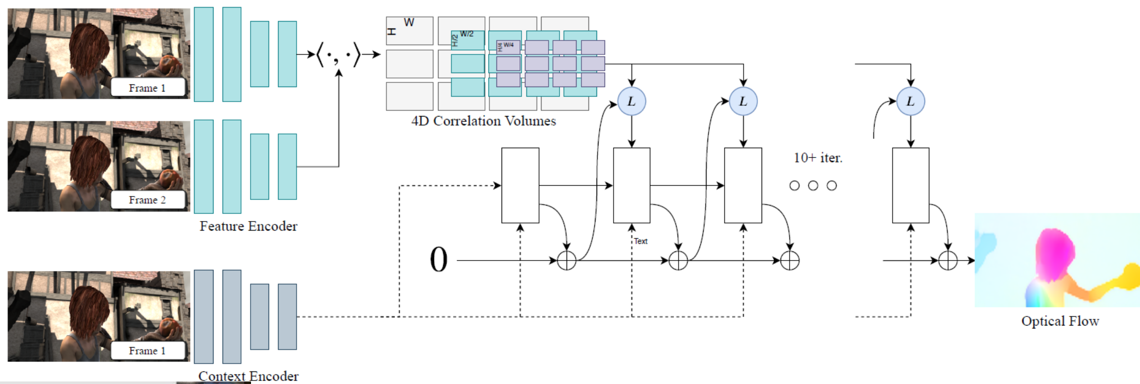
Sintel

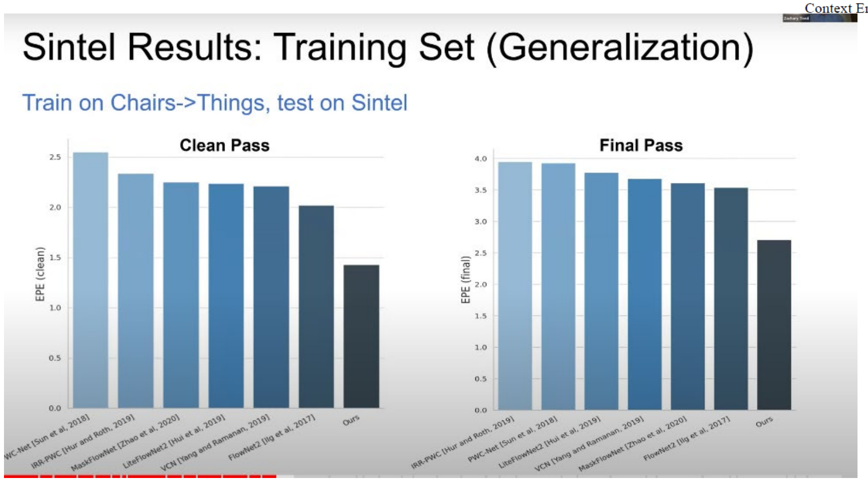
Butler, D. J., et al. (2012, October). A naturalistic open source movie for optical flow evaluation. ECCV 2012.

https://youtu.be/ZmiBI4tPk_o

RAFT

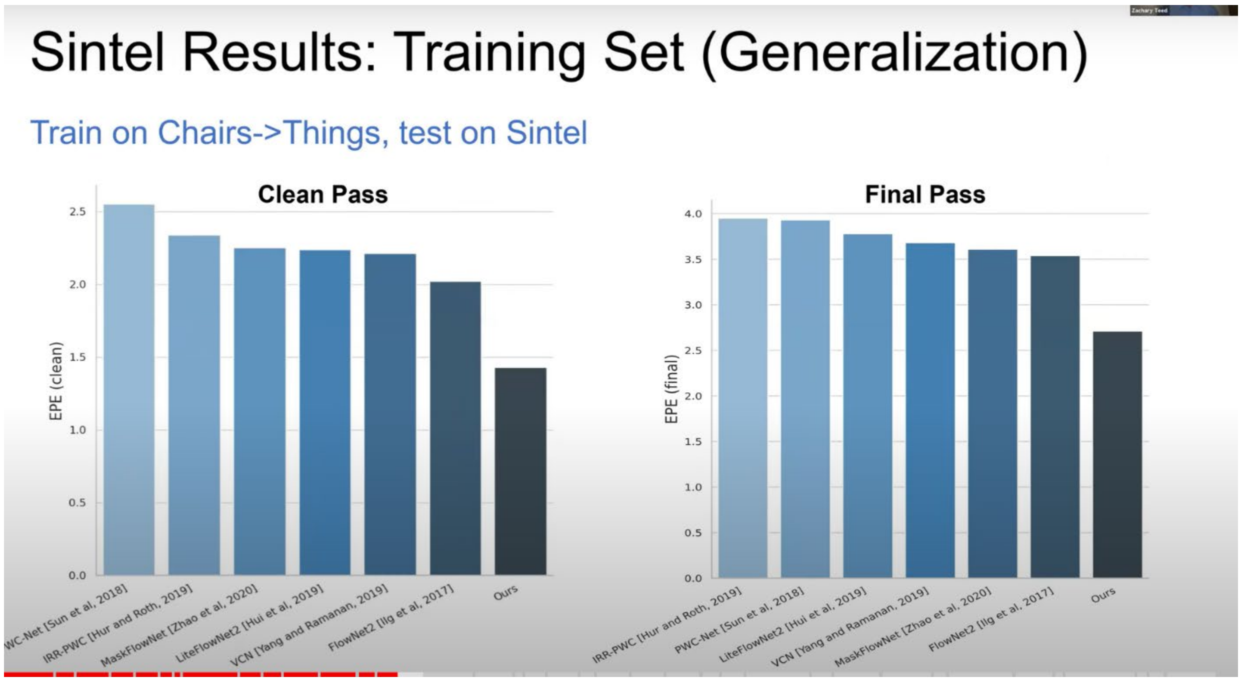
RAFT, 2020





Teed, Z., & Deng, J. (2020, August). Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision* (pp. 402-419). Springer, Cham.

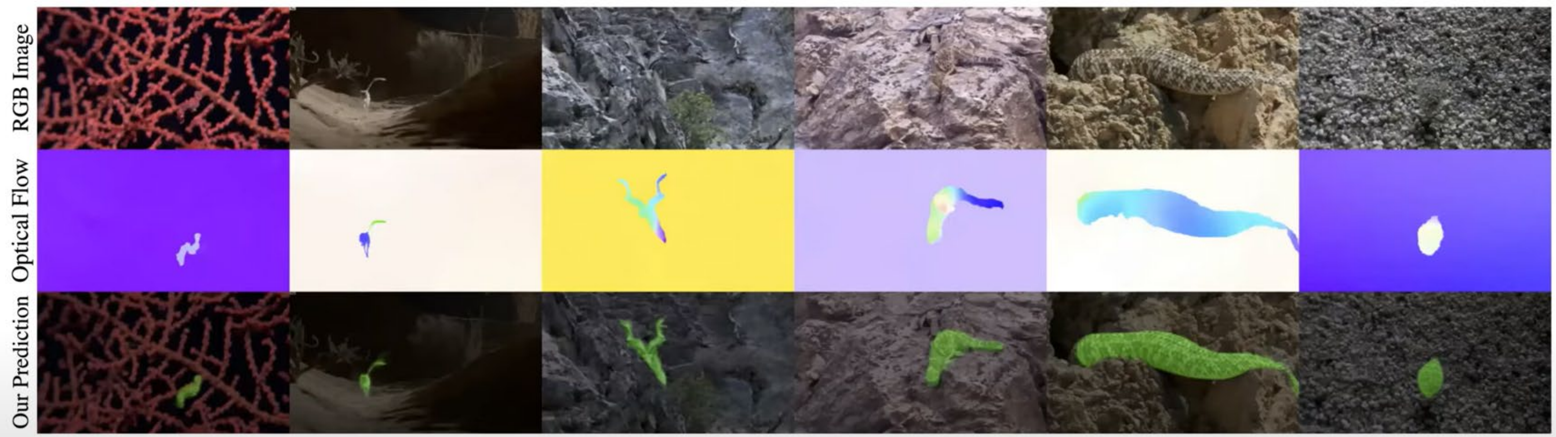
RAFT, test on Sintel



https://youtu.be/ul6pXRGKmco?si=GDfqyJl-f3ZyOXDh&t=1336

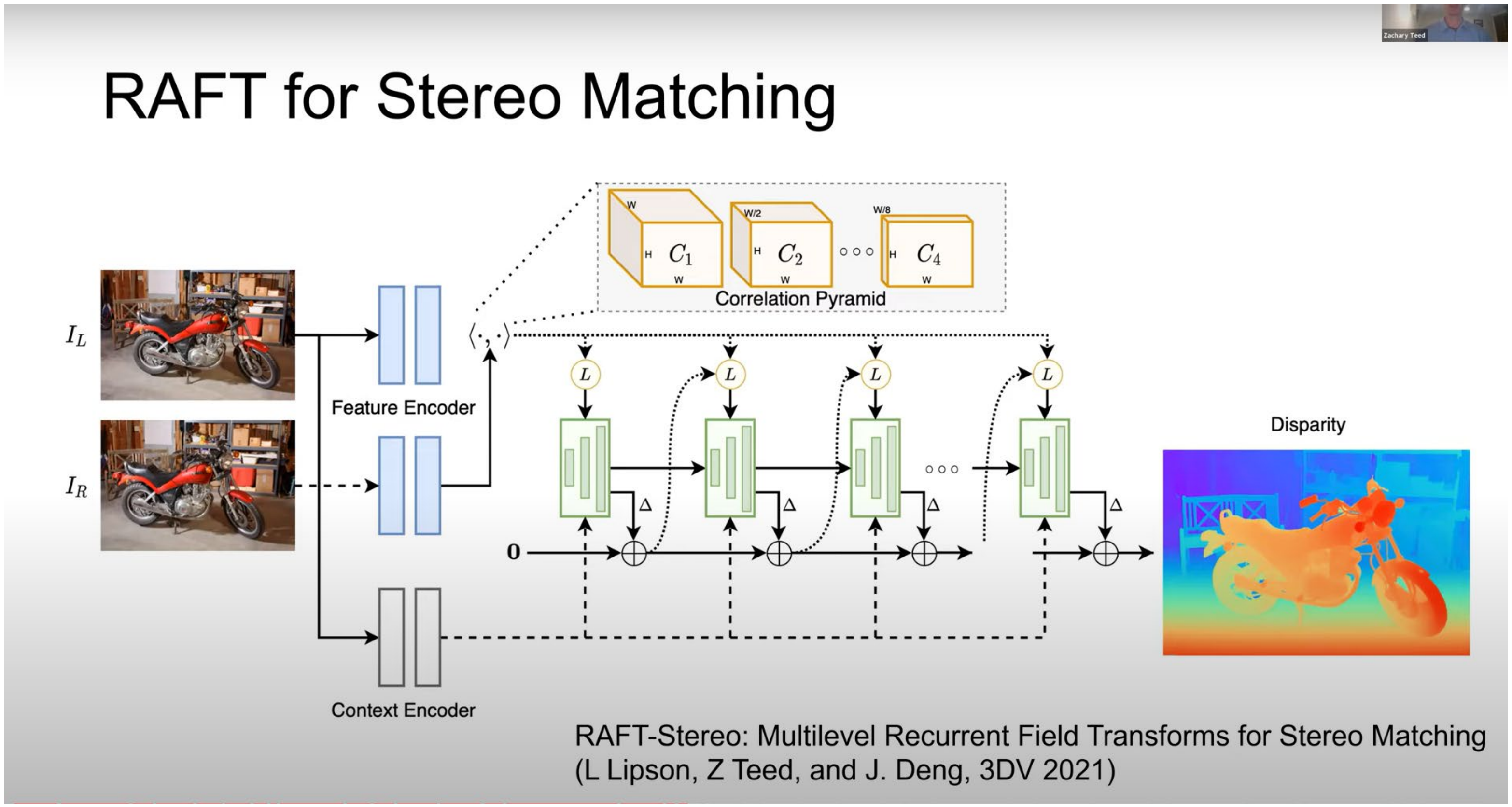
Discovering camouflage

Video Object Segmentation using RAFT



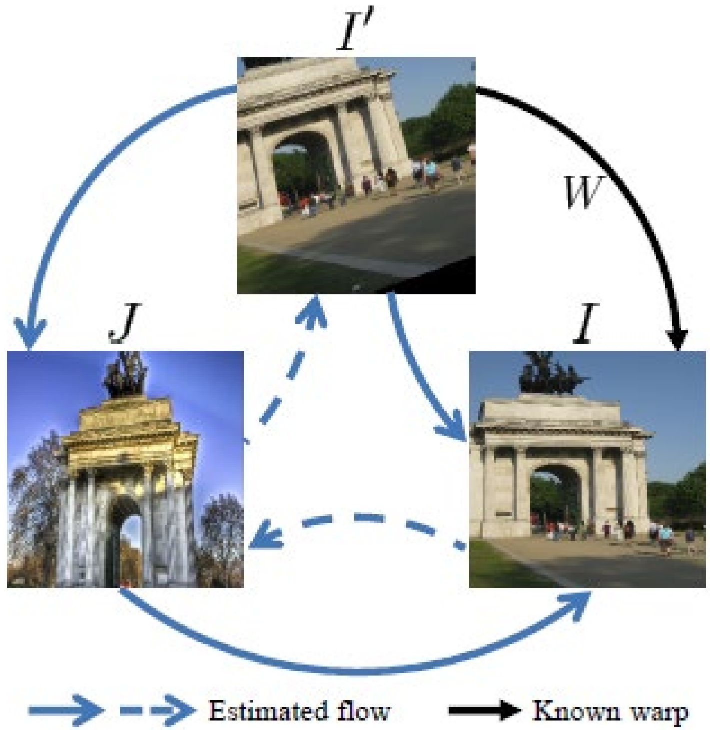
Self-supervised Video Object Segmentation by Motion Grouping, Yang et al. (ICCV), 2021

Stereo matching



Only searching along epiploar line

Warp Consistency



Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Warp Consistency for Unsupervised Learning of Dense Correspondences. ICCV 2021

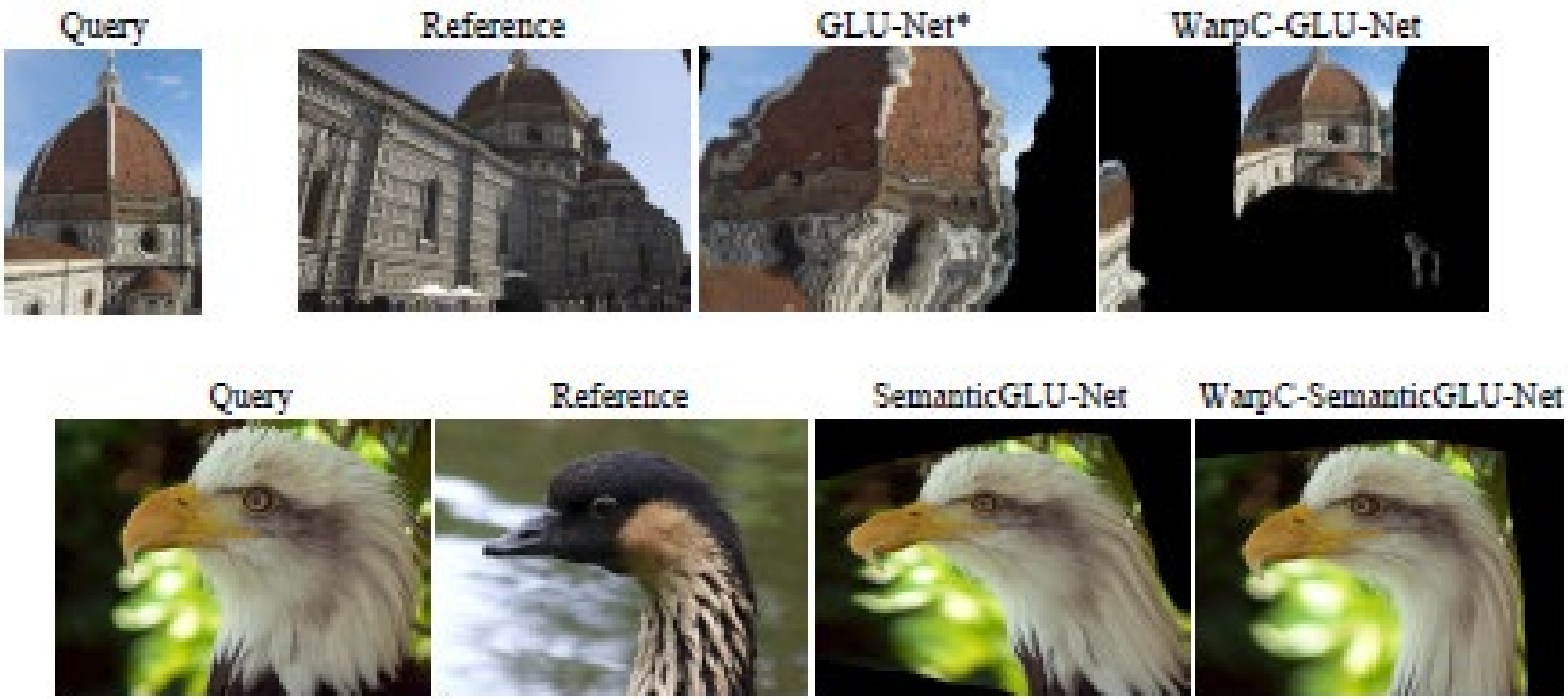
Bridge the simulation gap

Data training is covisible image pairs

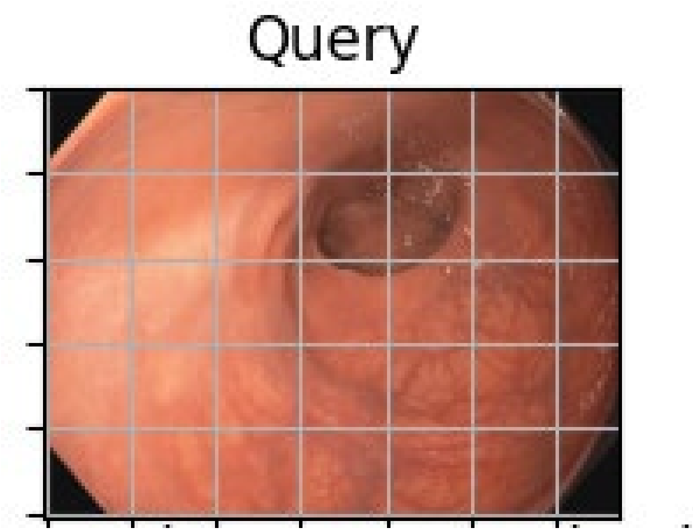
Warp Consistency Results

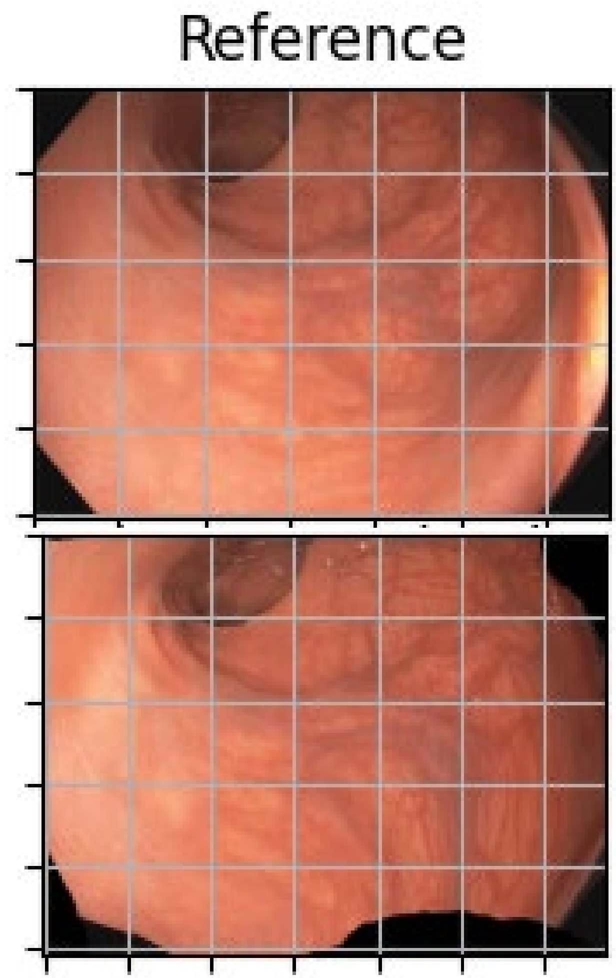
Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Warp Consistency for Unsupervised Learning of Dense Correspondences. ICCV 2021

Warp Consistency Results



Test on Endoscopy





Ivan Gonzalo. Cálculo de flujo óptico denso en imágenes de colonoscopias mediante aprendizaje no supervisado. TFG. Universidad de Zaragoza. 2023