Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

CS7GV1 Computer vision

Optical flow and edge-aware filters

Dr. Martin Alain

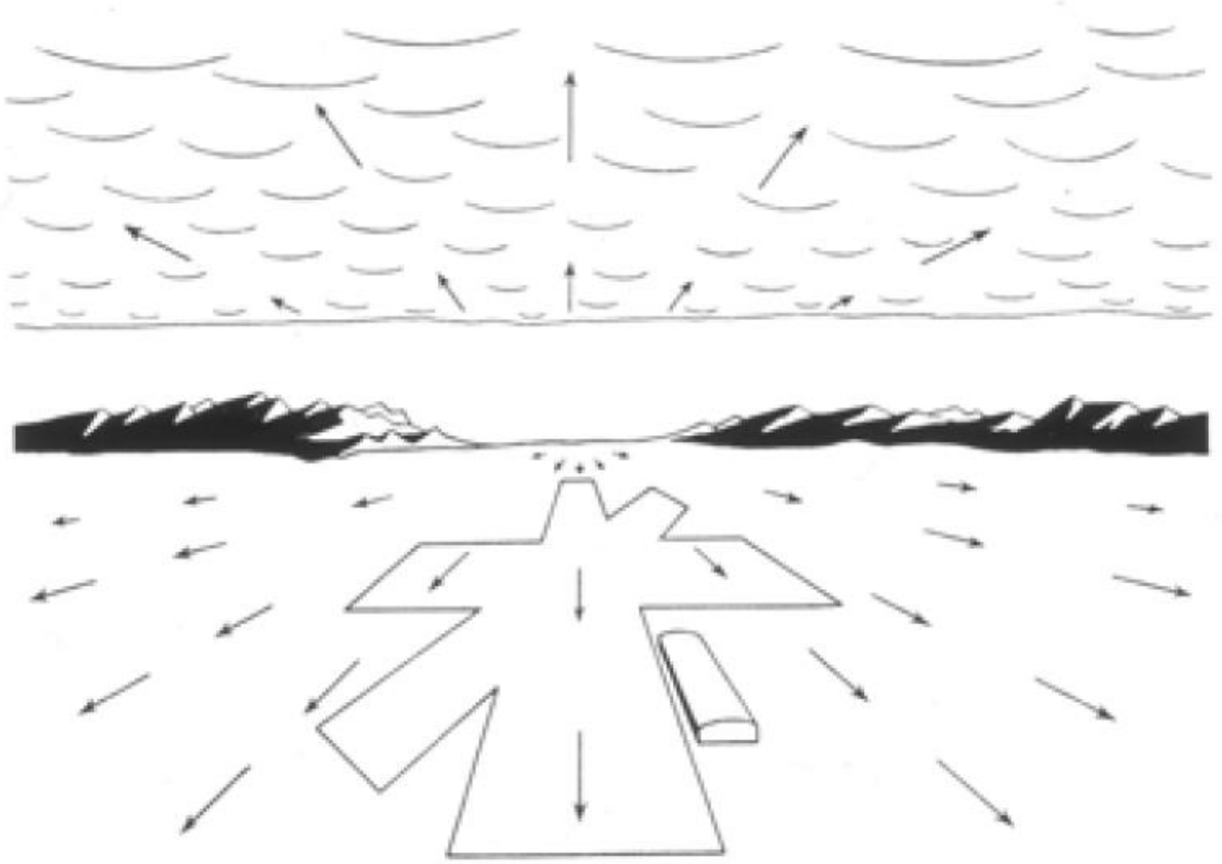
Introduction

- Videos are temporal sequences of images, sometimes associated with an audio stream (ignored here), sometimes with editing effects (e.g. transitions such as cut, fades, dissolves between shots).

- *'In filmmaking and video production, a shot is a series of frames, that runs for an uninterrupted period of time. Film shots are an essential aspect of a movie where angles, transitions and cuts are used to further express emotion, ideas and movement. The term "**shot**" can refer to two different parts of the filmmaking process:*
  - *In production, a shot is the moment that the camera starts rolling until the moment it stops.*
  - *In film editing, a **shot is the continuous footage or sequence between two edits or cuts**.'*

https://en.wikipedia.org/wiki/Shot_(filmmaking)

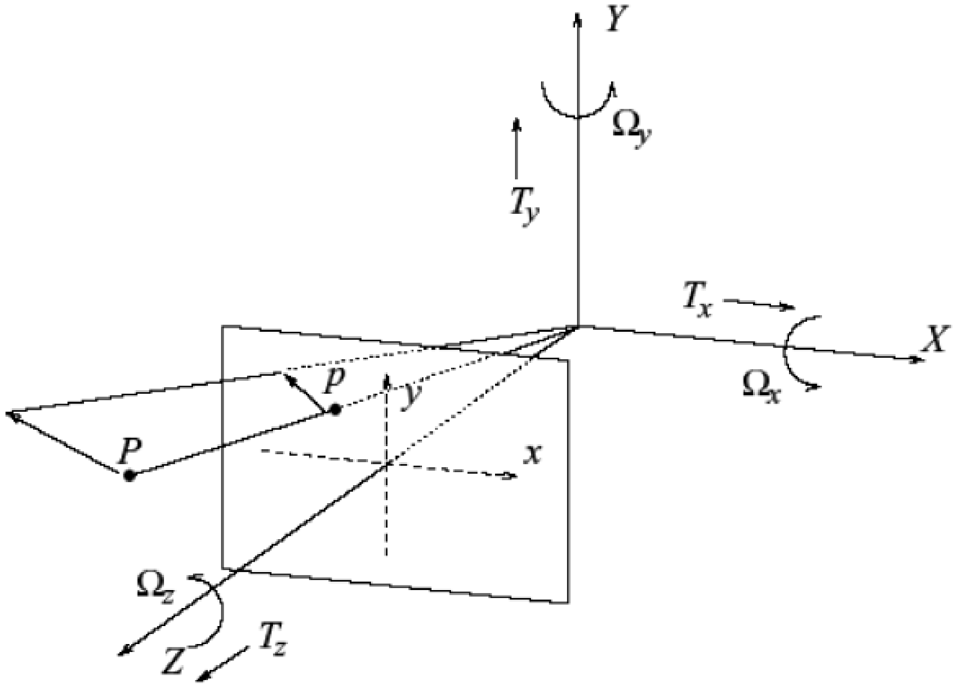
Optical Flow



J. J. Gibson, The Ecological Approach to Visual Perception

Motion Field

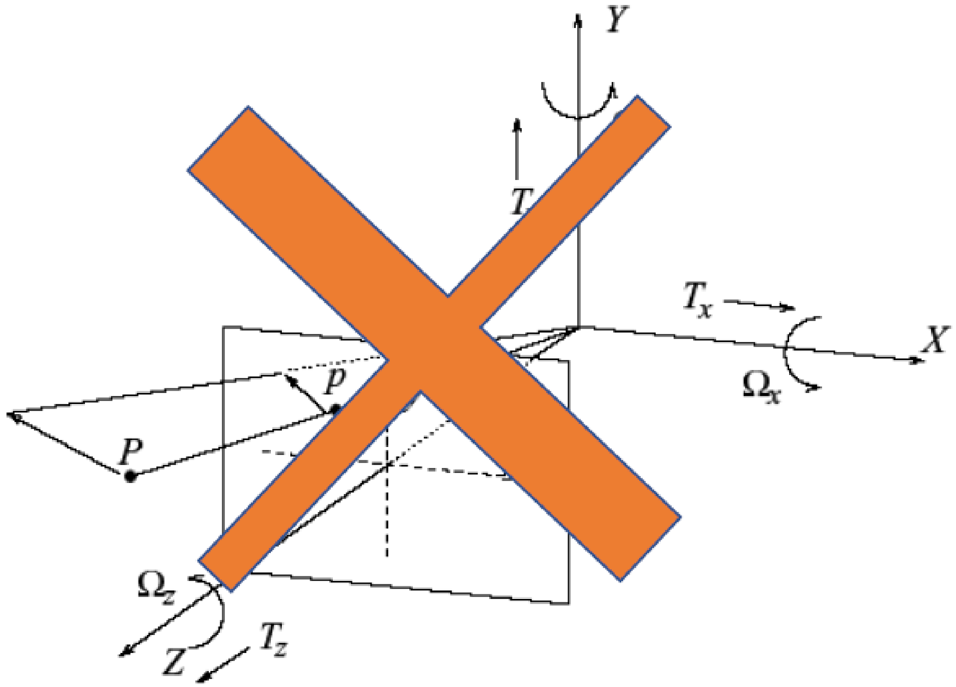
B. Horn, Robot Vision, MIT Press



Motion field = 2D motion field representing the projection of the 3D motion of points in the scene onto the image plane.

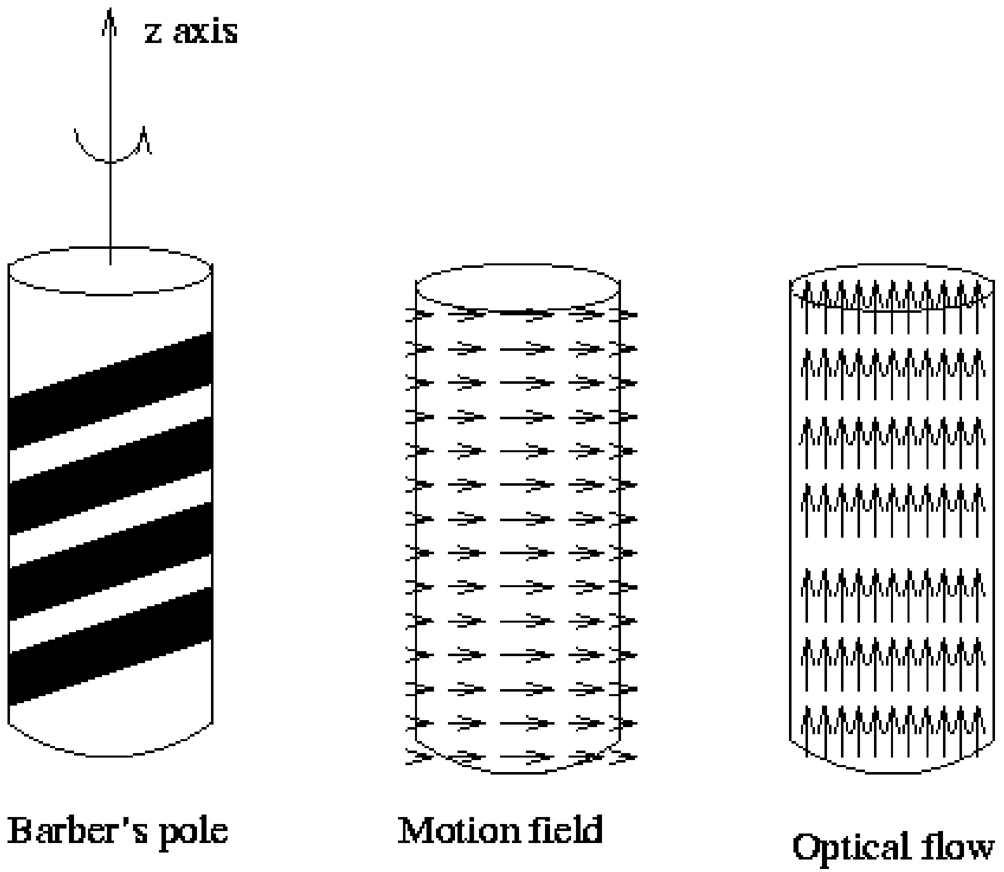
B. Horn, Robot Vision, MIT Press

Apparent motion



Optical flow = 2D velocity field describing the apparent motion in the images.

Apparent motion



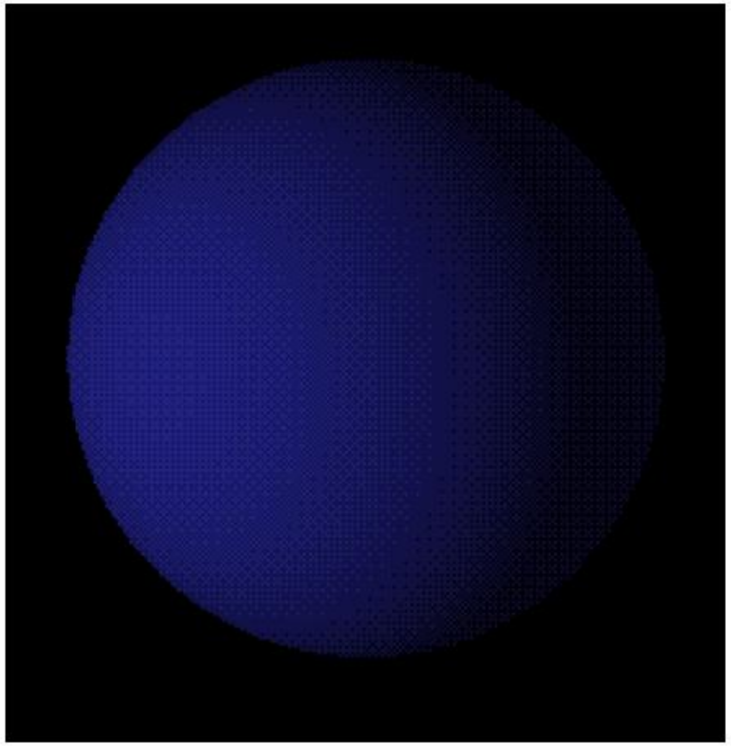
Robyn Owens

Thought Experiment 1

Lambertian (matte) ball rotating in 3D

What does the 2D motion field look like?

What does the 2D optical flow field look like?



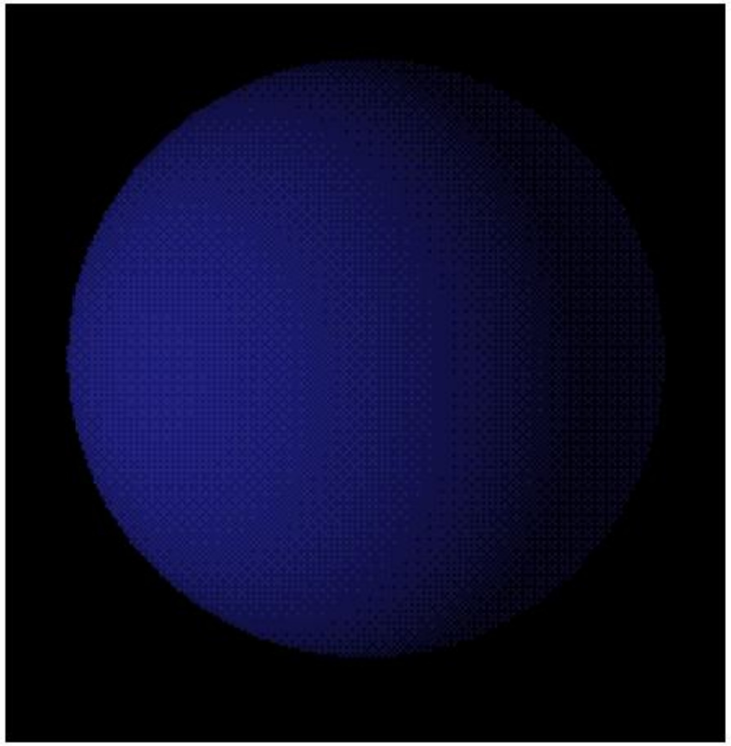
http://www.evl.uic.edu/aej/488/lecture12.html

Thought Experiment 2

Stationary Lambertian (matte) ball, moving light source.

What does the 2D motion field look like?

What does the 2D optical flow field look like?



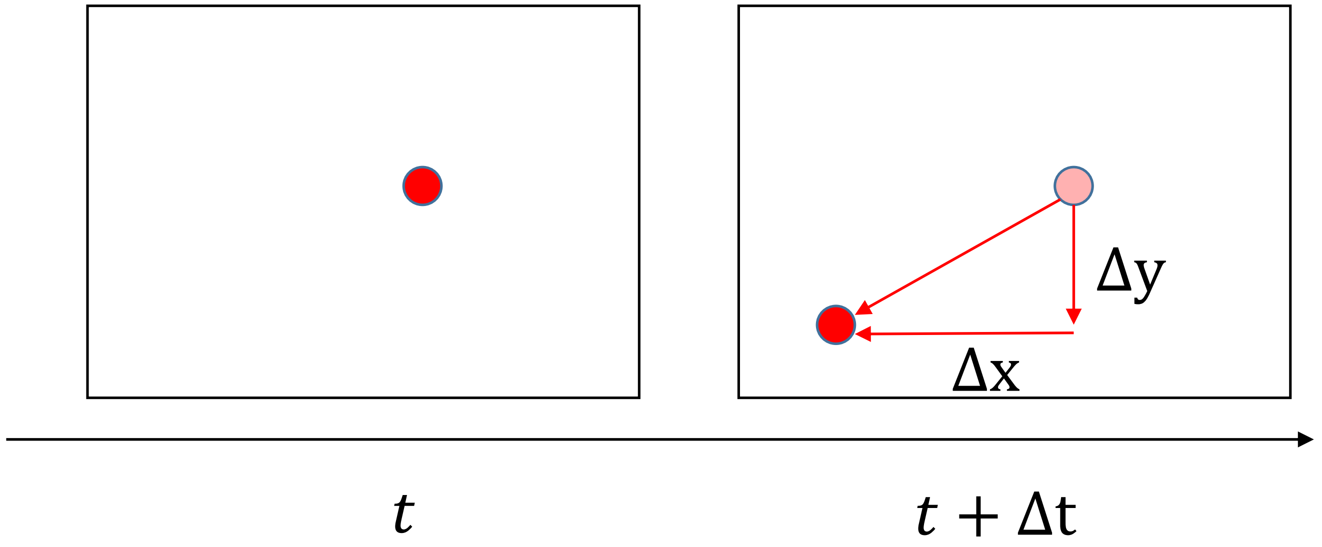
http://www.evl.uic.edu/aej/488/lecture12.html

Motion field vs Apparent motion

- It is important to keep these straight.
- They are “for” different problems.
- We often confuse them.
- But they really need different solutions.

Optical flow – the basics

- $I(x+\Delta x, y+\Delta \mathrm{y}, t+\Delta t) = I(x,y,t)$ (1)
- $I(x+\Delta x, y+\Delta \mathrm{y}, t+\Delta t) = I(x,y,t) + \frac{dI}{dx}\Delta x + \frac{dI}{dy}\Delta \mathrm{y} + \frac{dI}{dt}\Delta \mathrm{t}$ (2)
- $\frac{dI}{dx}\Delta x + \frac{dI}{dy}\Delta \mathrm{y} + \frac{dI}{dt}\Delta \mathrm{t} = 0$
- $\frac{dI}{dx}\frac{\Delta x}{\Delta \mathrm{t}} + \frac{dI}{dy}\frac{\Delta y}{\Delta \mathrm{t}} + \frac{dI}{dt}\frac{\Delta \mathrm{t}}{\Delta \mathrm{t}} = 0$
- $I_x u + I_y v + I_t = 0$
- $\nabla I \cdot (u,v) = -I_t$

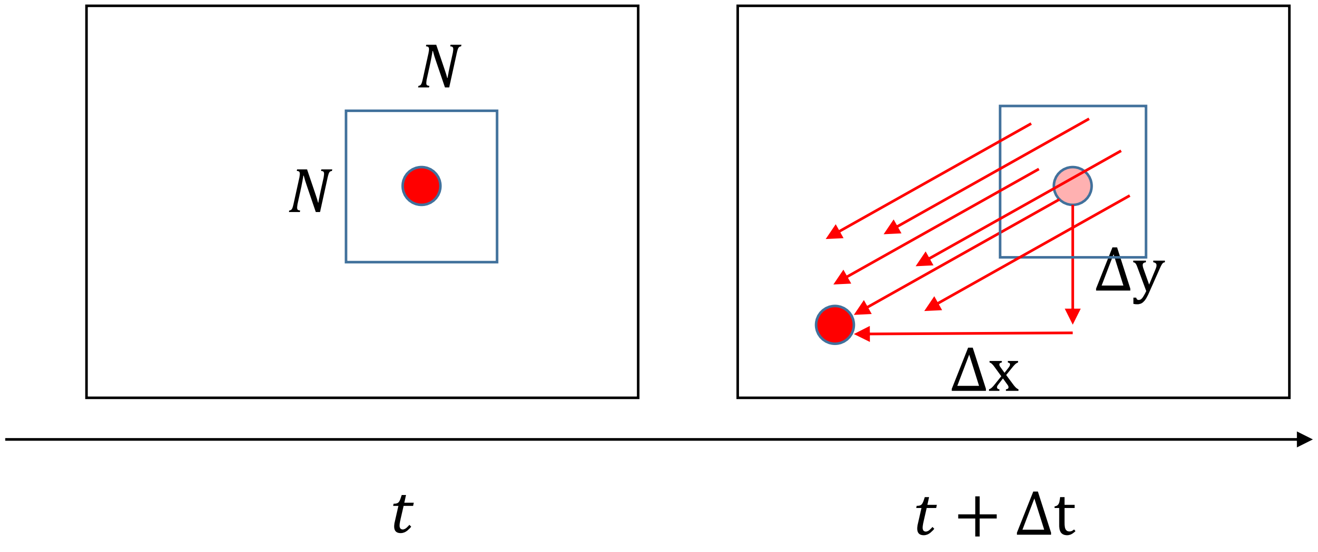


Optical flow – Lucas–Kanade

- $I_x^1 u + I_y^1 v + I_t^1 = 0$
- $I_x^2 u + I_y^2 v + I_t^2 = 0$
- …
- $I_x^{N^2} u + I_y^{N^2} v + I_t^{N^2} = 0$

- $A = \begin{bmatrix} I_x^1 & I_y^1 \\ \cdots & \cdots \\ I_x^{N^2} & I_y^{N^2} \end{bmatrix}, w = [u\ v], b = \begin{bmatrix} -I_t^1 \\ \cdots \\ -I_t^{N^2} \end{bmatrix}$

- $Aw = b$
- $w = (A^T A)^{-1} Ab$



B. D. Lucas and T. Kanade (1981), An iterative image registration technique with an application to stereo vision. Proceedings of Imaging Understanding Workshop, pages 121--130

Optical flow – PatchMatch

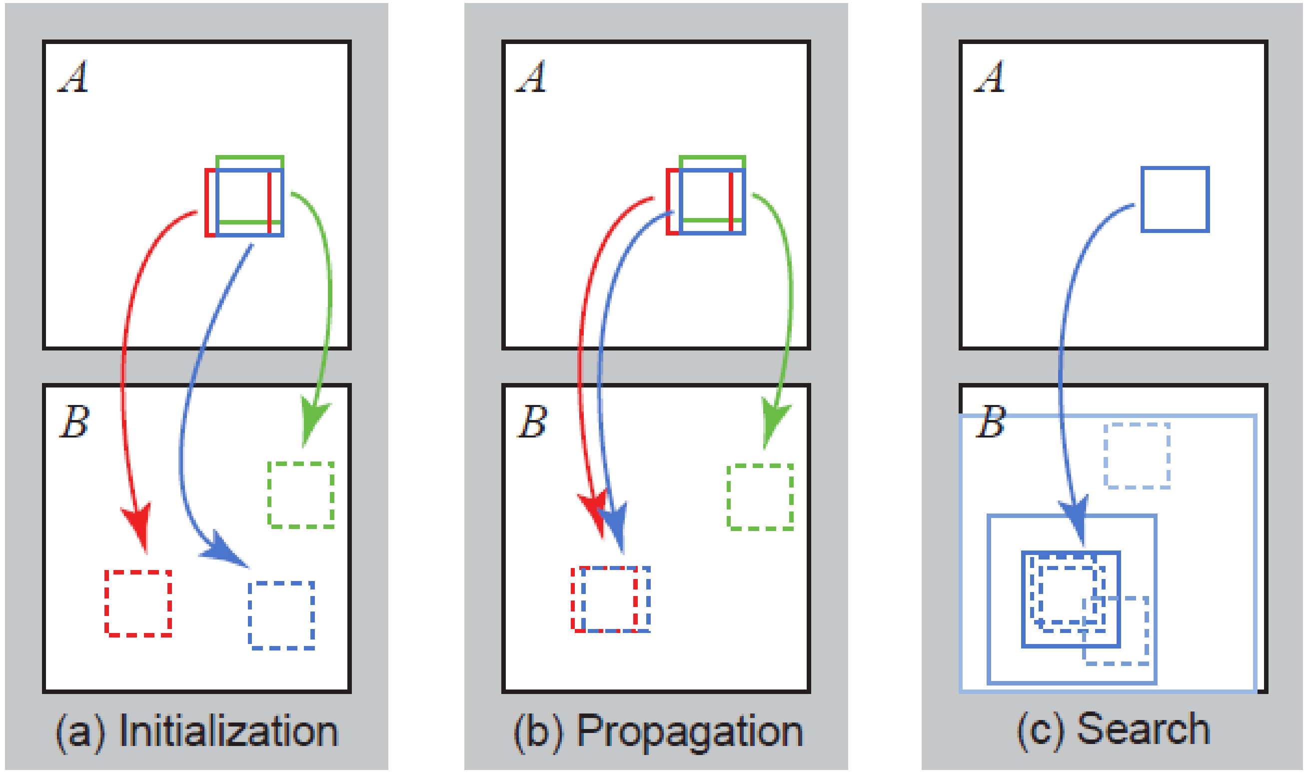
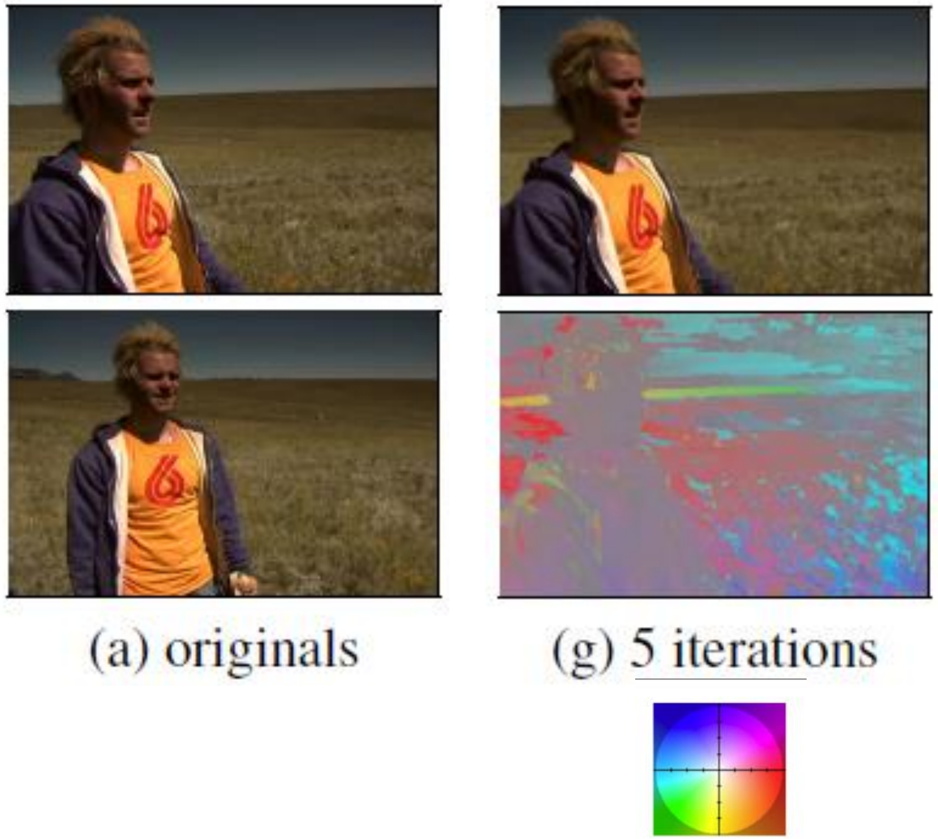


Figure 2: *Phases of the randomized nearest neighbor algorithm: (a) patches initially have random assignments; (b) the blue patch checks above/green and left/red neighbors to see if they will improve the blue mapping, propagating good matches; (c) the patch searches randomly for improvements in concentric neighborhoods.*



Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. "PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing." *ACM Transactions on Graphics (Proc. SIGGRAPH) 28(3)*, August 2009.

Optical flow – Horn and Schunck

- $L = \left(I_x u + I_y v + I_t\right)^2 + \lambda(\|\nabla u\|^2 + \|\nabla v\|^2)$
- $E = \int\int L\, dxdy$

B.K.P. Horn and B.G. Schunck, "Determining optical flow." *Artificial Intelligence*, vol 17, pp 185–203, 1981.

Optical flow – Horn and Schunck

Definition

We consider the optimization problem:

$$\begin{array}{lll} \text{minimize} & f_0(\mathbf{x}) & \\ \text{subject to} & f_i(\mathbf{x}) = 0 & i = 1, \cdots, m \\ & h_j(\mathbf{x}) \le 0 & j = 1, \cdots, p \end{array}$$

with $\mathbf{x} \in \mathbf{R}^d$.

The Lagrangian $\mathscr{L} : \mathbb{R}^d \times \mathbb{R}^m \times \mathbb{R}^p$ associated with the problem is defined as:

$$\mathscr{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{v}) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^{p} v_j h_j(\mathbf{x})$$

The vectors $\boldsymbol{\lambda}$ and $\boldsymbol{v}$ are called the Lagrange multiplier vectors.

B.K.P. Horn and B.G. Schunck, "Determining optical flow." *Artificial Intelligence*, vol 17, pp 185–203, 1981.

Optical flow – Horn and Schunck

- $L = \left(I_x u + I_y v + I_t\right)^2 + \lambda(\|\nabla u\|^2 + \|\nabla v\|^2)$
- $\frac{dL}{du} - \frac{d}{dx}\frac{dL}{du_x} - \frac{d}{dy}\frac{dL}{du_y} = 0\ (1)$
- $\frac{dL}{dv} - \frac{d}{dx}\frac{dL}{dv_x} - \frac{d}{dy}\frac{dL}{dv_y} = 0\ (2)$
- $I_x\left(I_x u + I_y v + I_t\right) - \lambda^2 \Delta u = 0\ (1)$
- $I_y\left(I_x u + I_y v + I_t\right) - \lambda^2 \Delta v = 0\ (2)$

- $\Delta u(x, y) = 4(\bar{u}(x, y) - u(x, y))$

B.K.P. Horn and B.G. Schunck, "Determining optical flow." *Artificial Intelligence*, vol 17, pp 185–203, 1981.

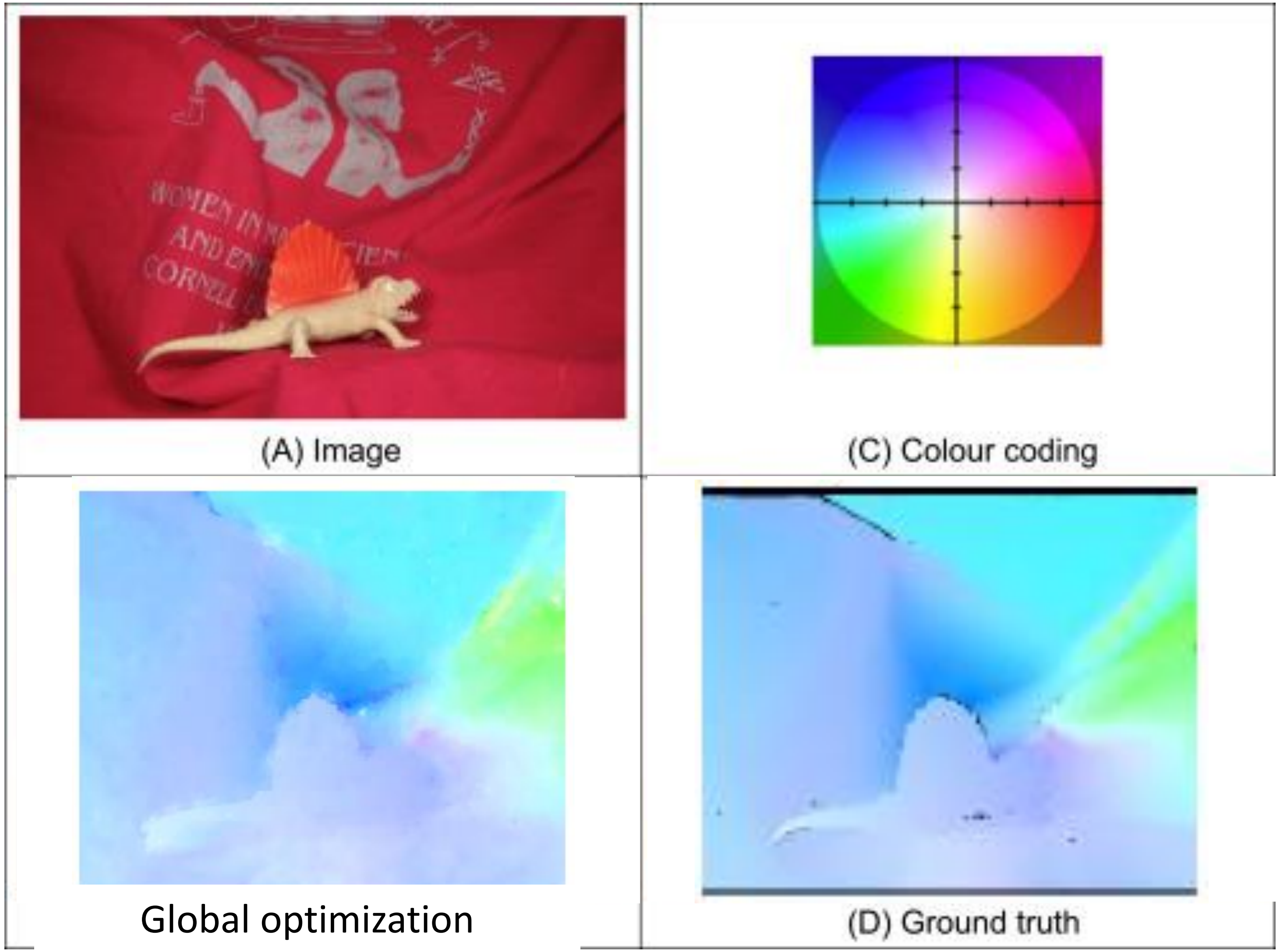
Optical flow – Horn and Schunck

- $(I_x^2 + 4\lambda^2)u + I_x I_y v = 4\lambda^2 \bar{u} - I_x I_t$ (1)
- $I_x I_y u + \left(I_y^2 + 4\lambda^2\right)v = 4\lambda^2 \bar{v} - I_y I_t$(2)

- $u^{k+1} = \bar{u}^k - \frac{I_x(I_x \bar{u}^k + I_y v^k + I_t)}{4\lambda^2 + I_x^2 + I_y^2}$
- $v^{k+1} = \bar{v}^k - \frac{I_y(I_x \bar{u}^k + I_y v^k + I_t)}{4\lambda^2 + I_x^2 + I_y^2}$

B.K.P. Horn and B.G. Schunck, "Determining optical flow." *Artificial Intelligence*, vol 17, pp 185–203, 1981.

Optical flow

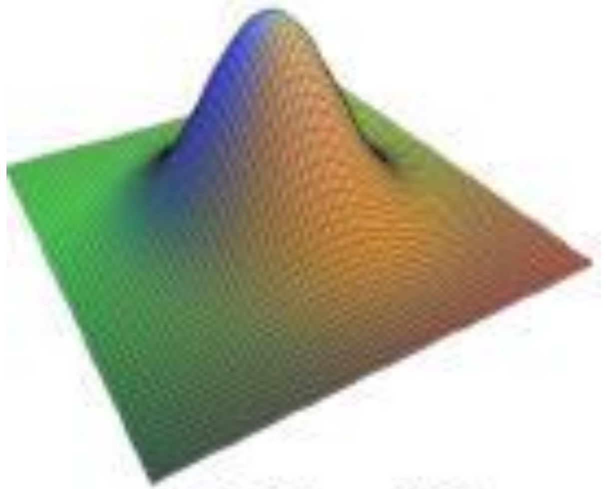


Edge aware filtering

- Reminder on the Gaussian filter: NOT edge-aware

$$J_p = \frac{1}{\sum_{q\in\Omega} f(q-p)} \sum_{q\in\Omega} f(q-p) I_q$$

$$f(q-p) = e^{-|p-q|^2/2\sigma_f^2}$$



spatial kernel f

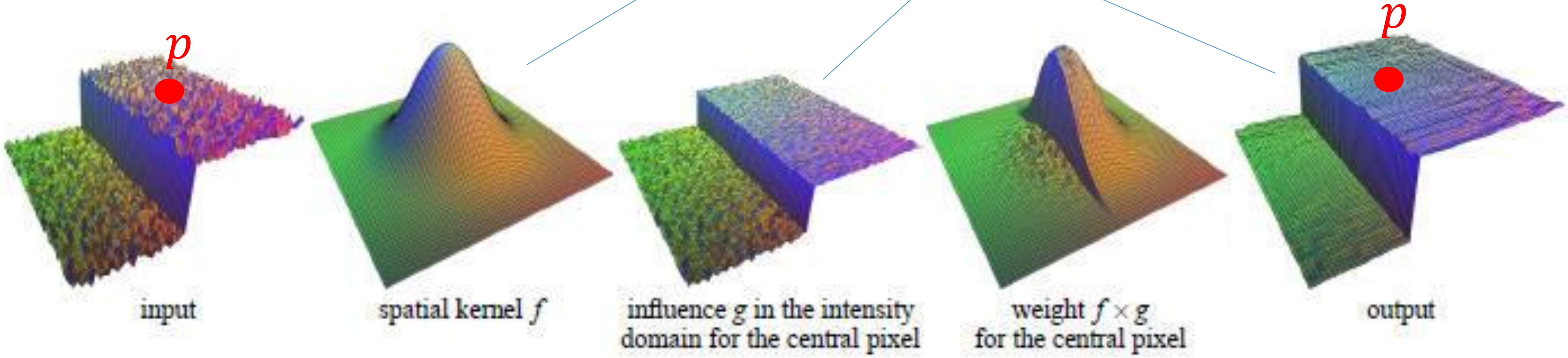
Edge aware filtering – Bilateral filter

$$J_p = \frac{1}{k(p)} \sum_{q \in \Omega} f(q-p) g(I_q - I_p) I_q$$

$$f(q-p) = e^{-|p-q|^2/2\sigma_f^2}$$

$$k(p) = \sum_{q \in \Omega} f(q-p) g(I_q - I_p) I_q$$

$$g(I_q - I_p) = e^{-\|I_p - I_q\|^2/2\sigma_g^2}$$



C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images", *Proceedings of the 1998 IEEE International Conference on Computer Vision*, Bombay, India.

Edge aware filtering – Bilateral filter

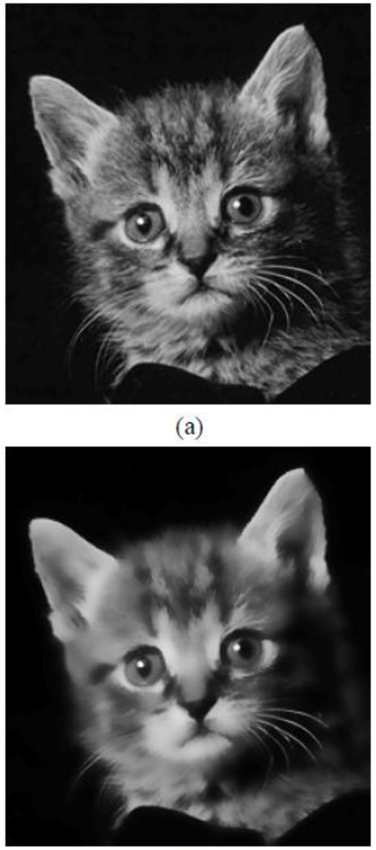
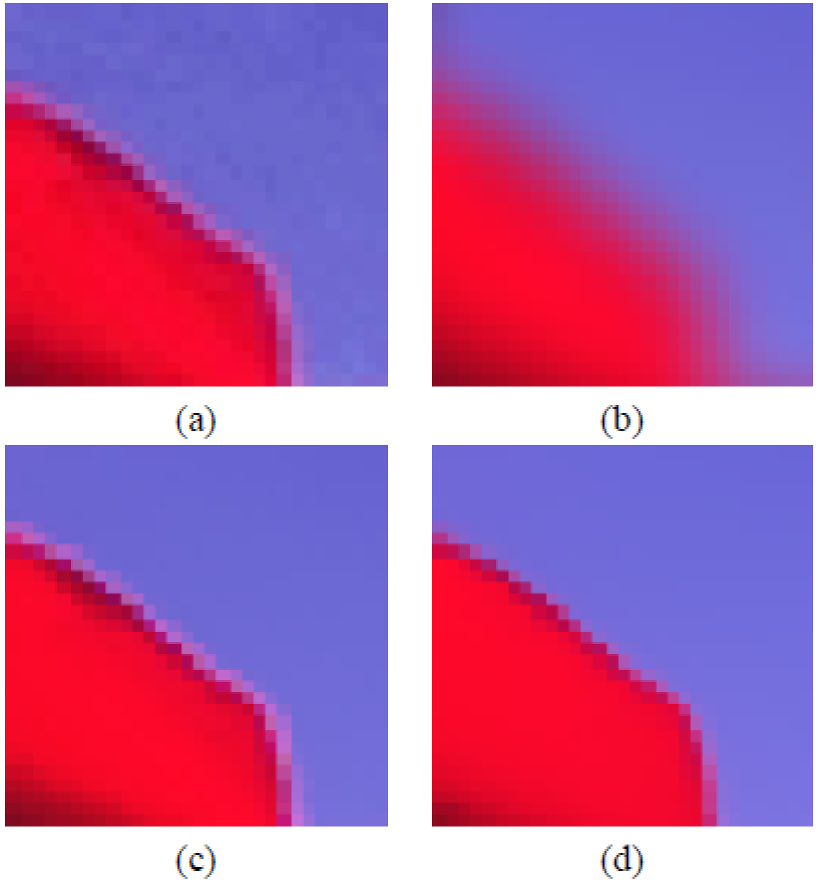


Figure 5: A picture before (a) and after (b) bilateral filtering.



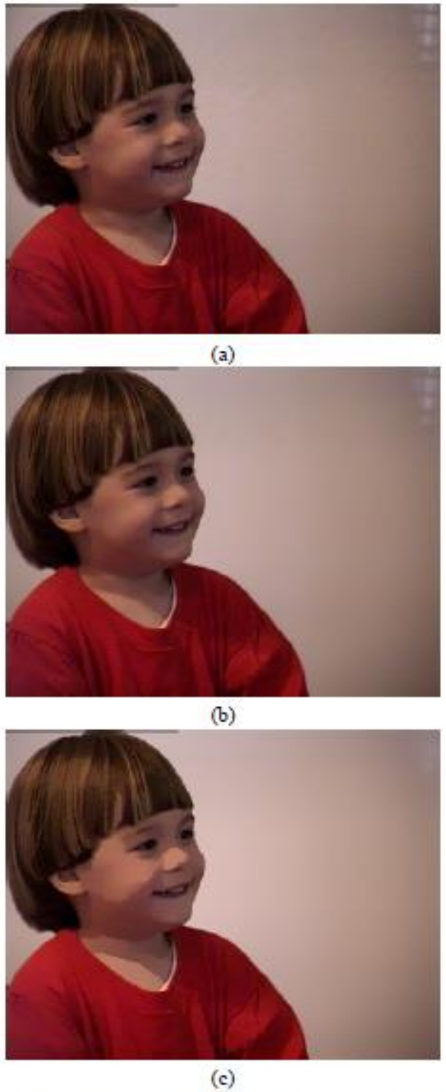


Figure 7: [above] (a) A color image, and its bilaterally smoothed versions after one (b) and five (c) iterations.

C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images", *Proceedings of the 1998 IEEE International Conference on Computer Vision*, Bombay, India.

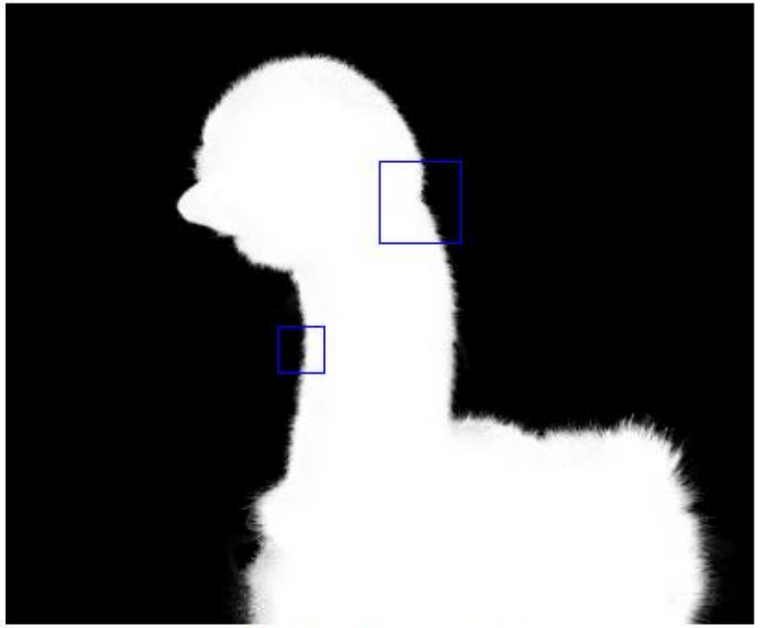
Edge aware filtering - Domain Transform

Eduardo S. L. Gastal and Manuel M. Oliveira. *"Domain Transform for Edge-Aware Image and Video Processing"*. **ACM Transactions on Graphics.** Volume 30 (2011), Number 4, Proceedings of SIGGRAPH 2011, Article 69.

Edge aware filtering – Guided image filter

Guidance I

Binary Mask p

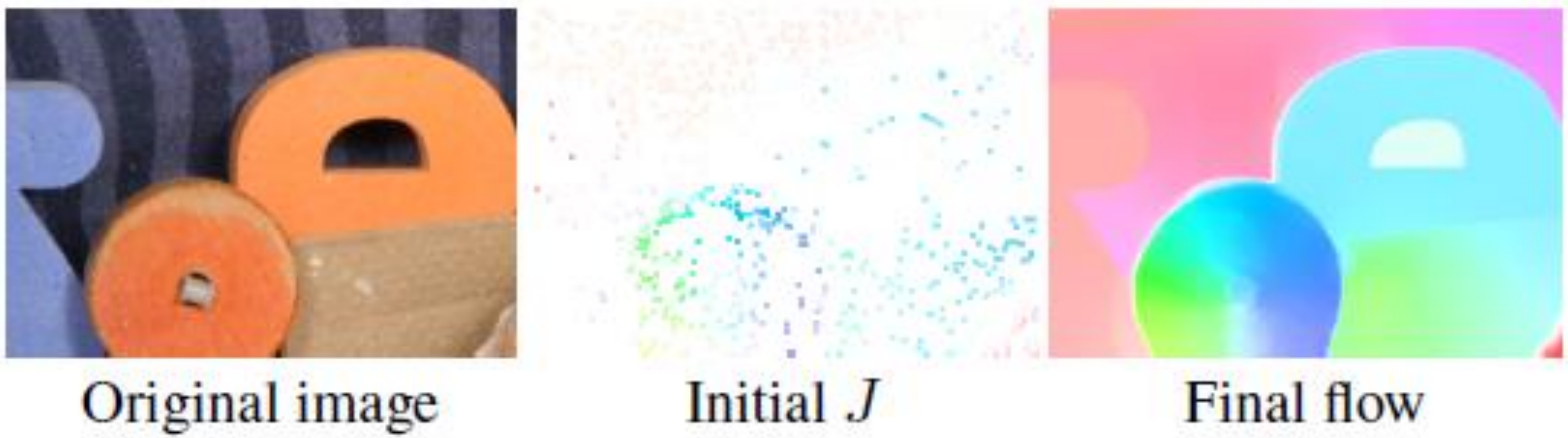


Guided Filter Output q

[Guided Image Filtering](), by Kaiming He, Jian Sun, and Xiaoou Tang, in **TPAMI** 2013

Edge-aware filtering for optical flow

- $L = \left(I_x u + I_y v + I_t\right)^2 + \lambda(\|\nabla u\|^2 + \|\nabla v\|^2)$
- $E = \int\int L\, dxdy$
- Replace global optimization of $\|\nabla u\|^2 + \|\nabla v\|^2$ by edge aware filtering



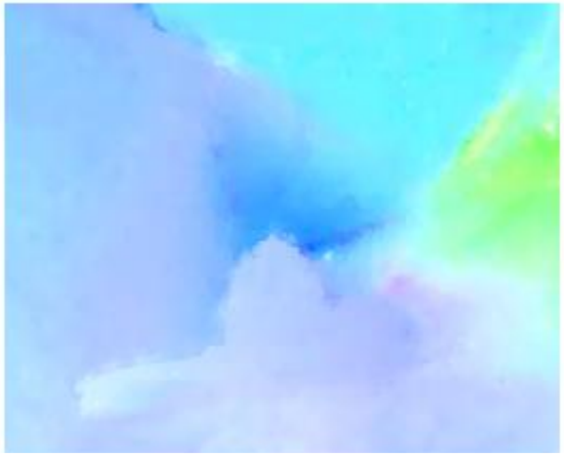
Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.

Edge-aware filtering for optical flow

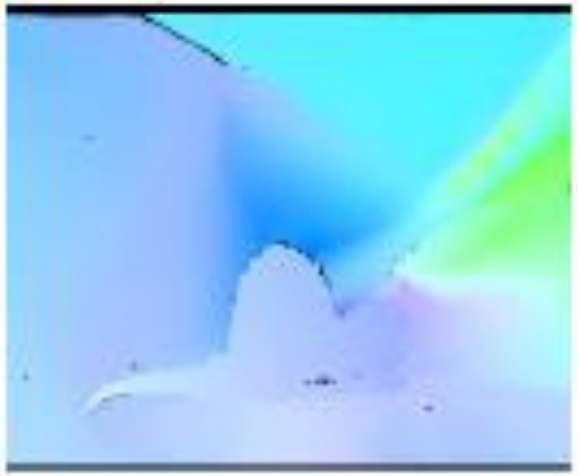
(A) Image

(C) Colour coding

Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.



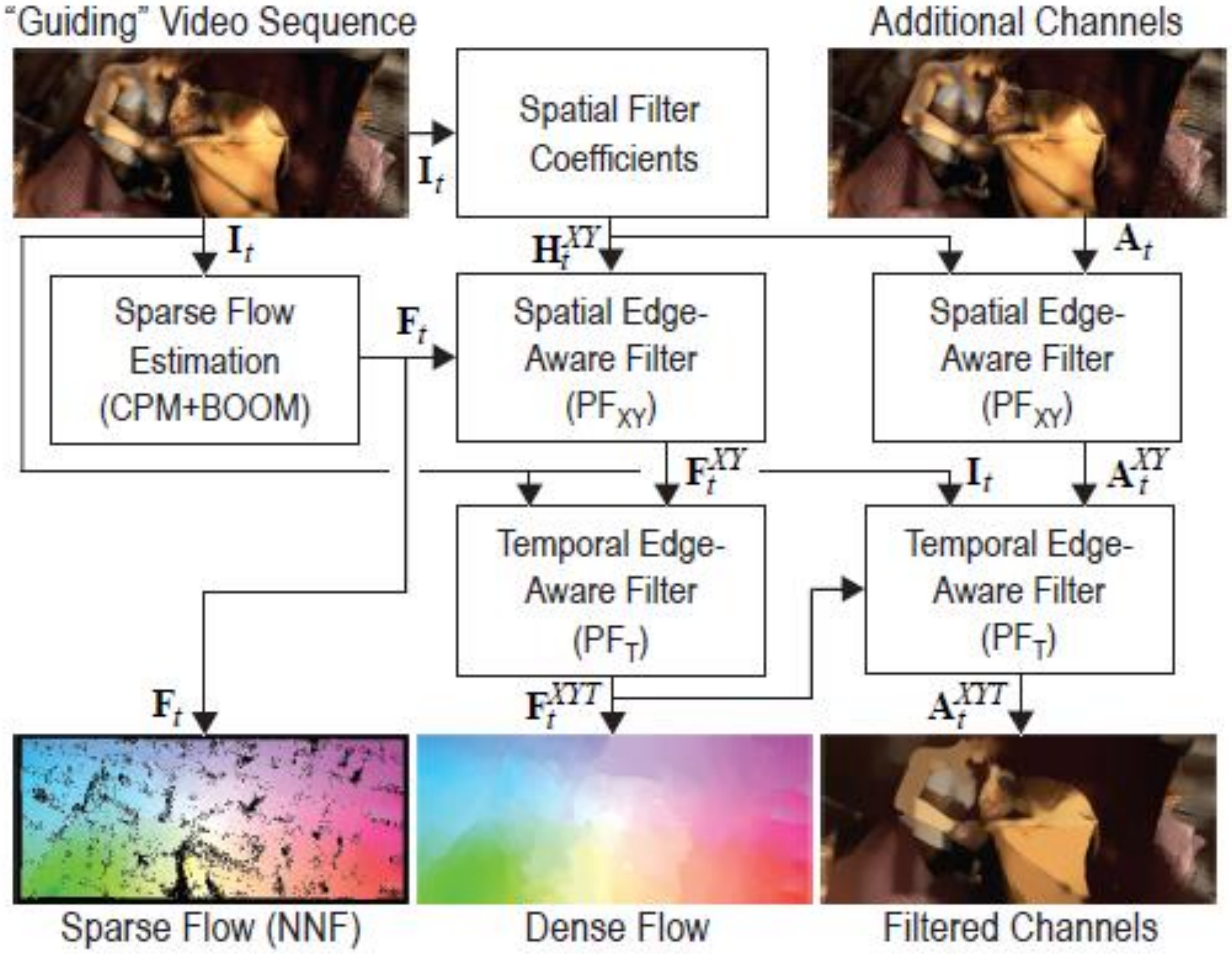
Global optimization



(D) Ground truth

Edge aware filter

Edge-aware filtering for optical flow



Towards Edge-Aware Spatio-Temporal Filtering in Real-Time.

M. Schaffner, F. Scheidegger, L. Cavigelli, H. Kaeslin, L. Benini and A. Smolic, *Trans. on Image Processing (TIP),* 2017.

Applications

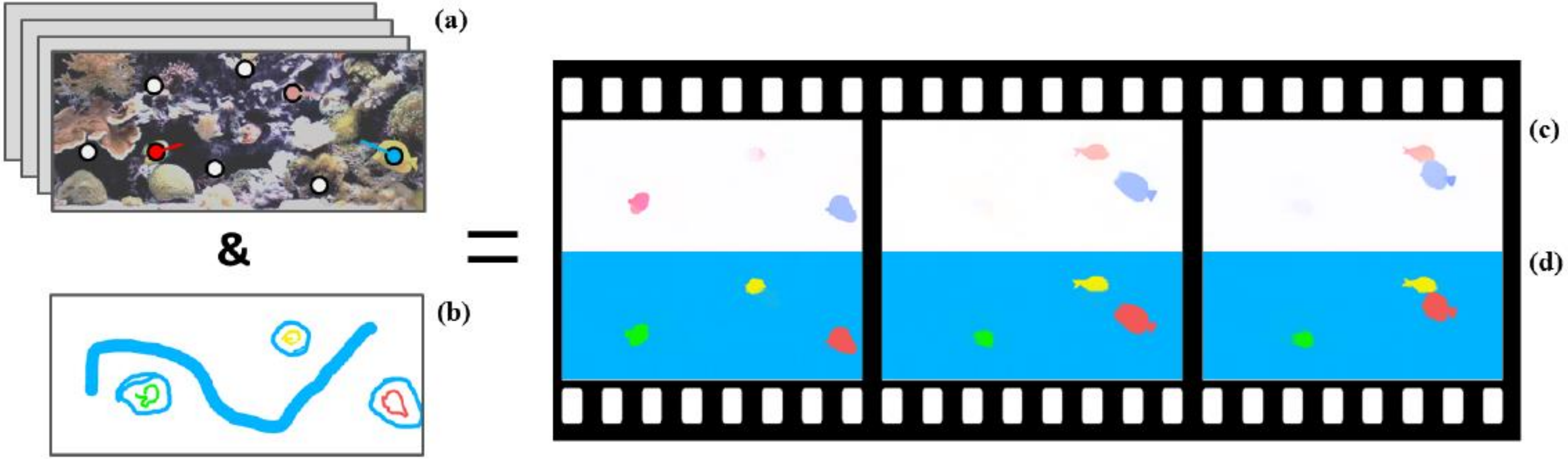
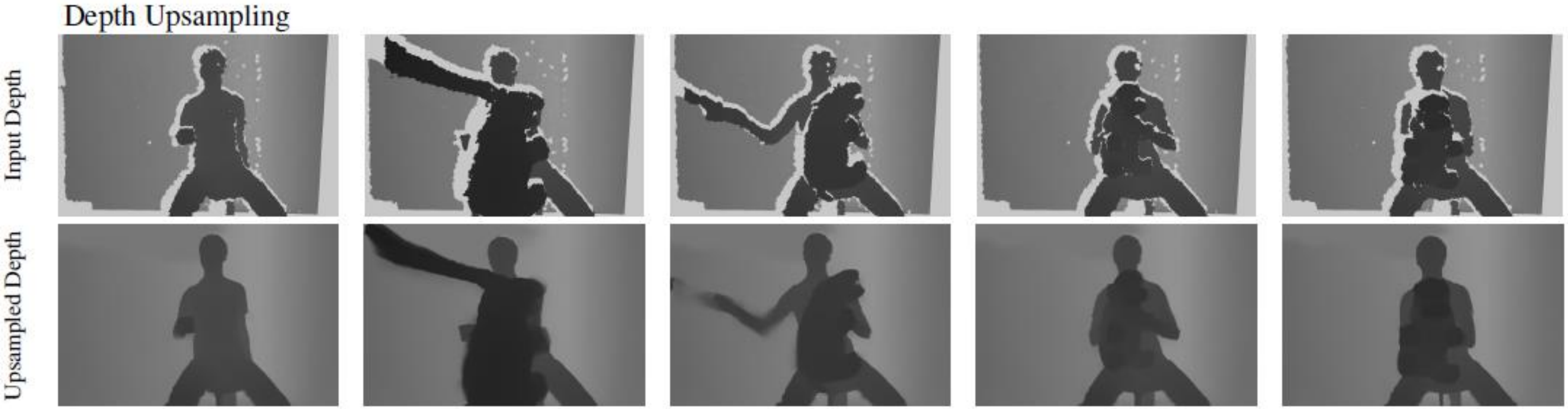


Figure 1: *One application of our method is the temporally consistent propagation of scribbles through video volumes. Sparse feature correspondences from an input video (a) are used to compute optical flow (c). Then, color scribbles (b) are spread in space and time to compute the final coherent output (d).*

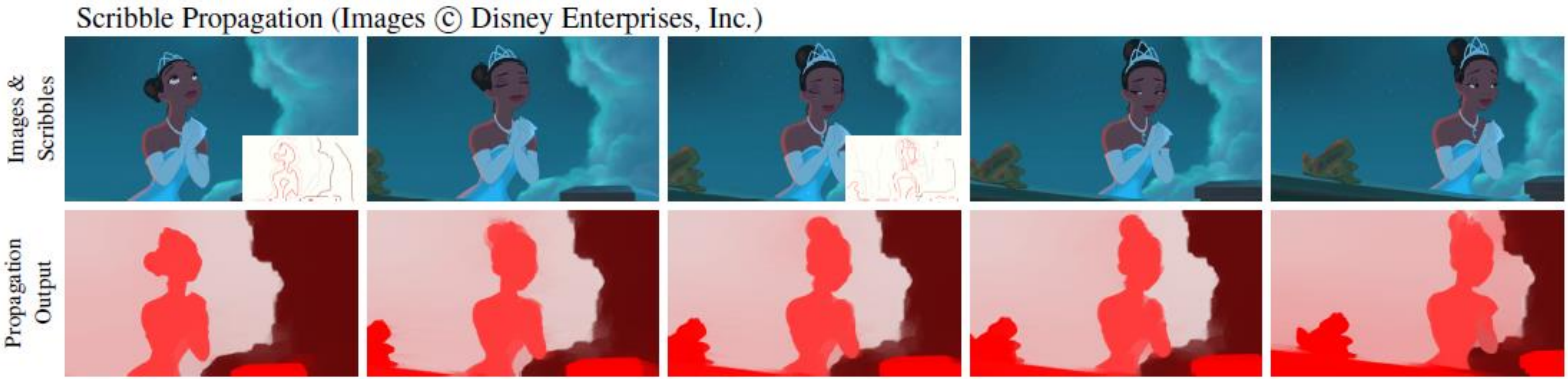
Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.

Applications



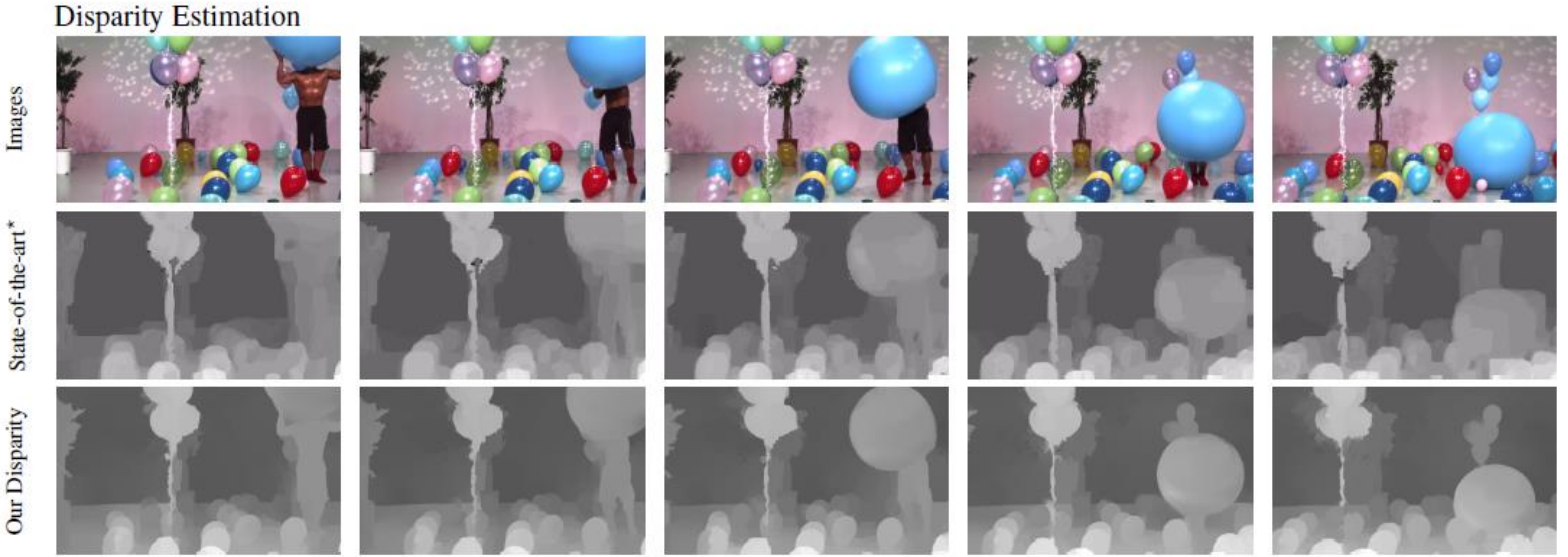
Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.

Applications



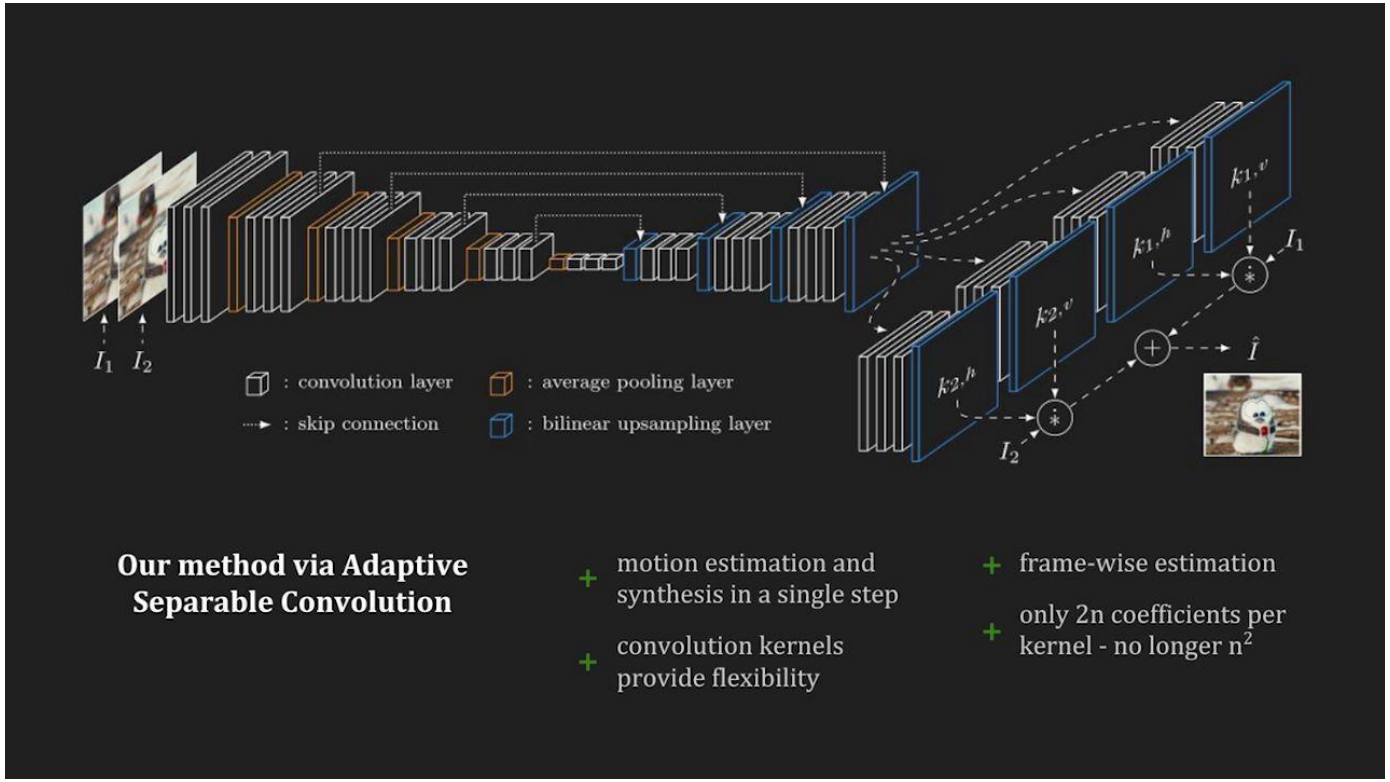
Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.

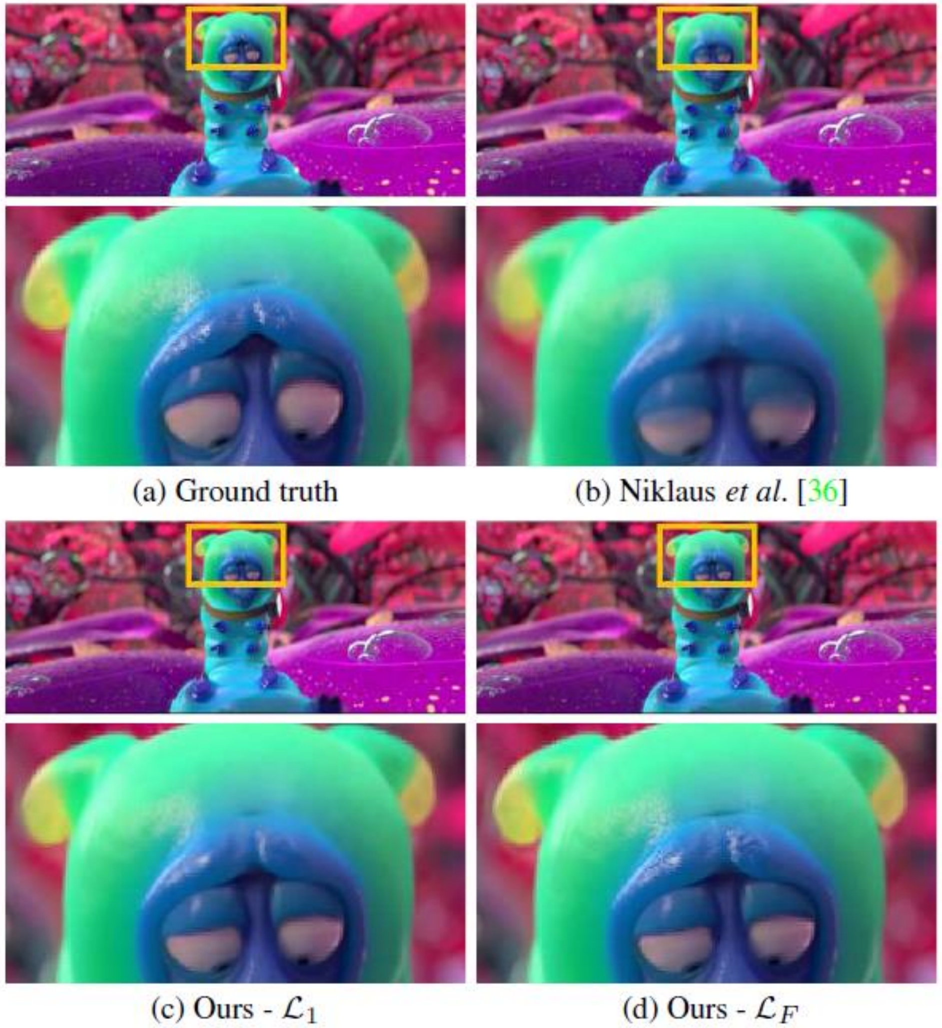
Applications



Lang, M., Wang, O., Aydin, T.O., Smolic, A., & Gross, M.H. (2012). Practical temporal consistency for image-based graphics applications. ACM Transactions on Graphics (TOG), 31, 1 - 8.

Applications – slow motion



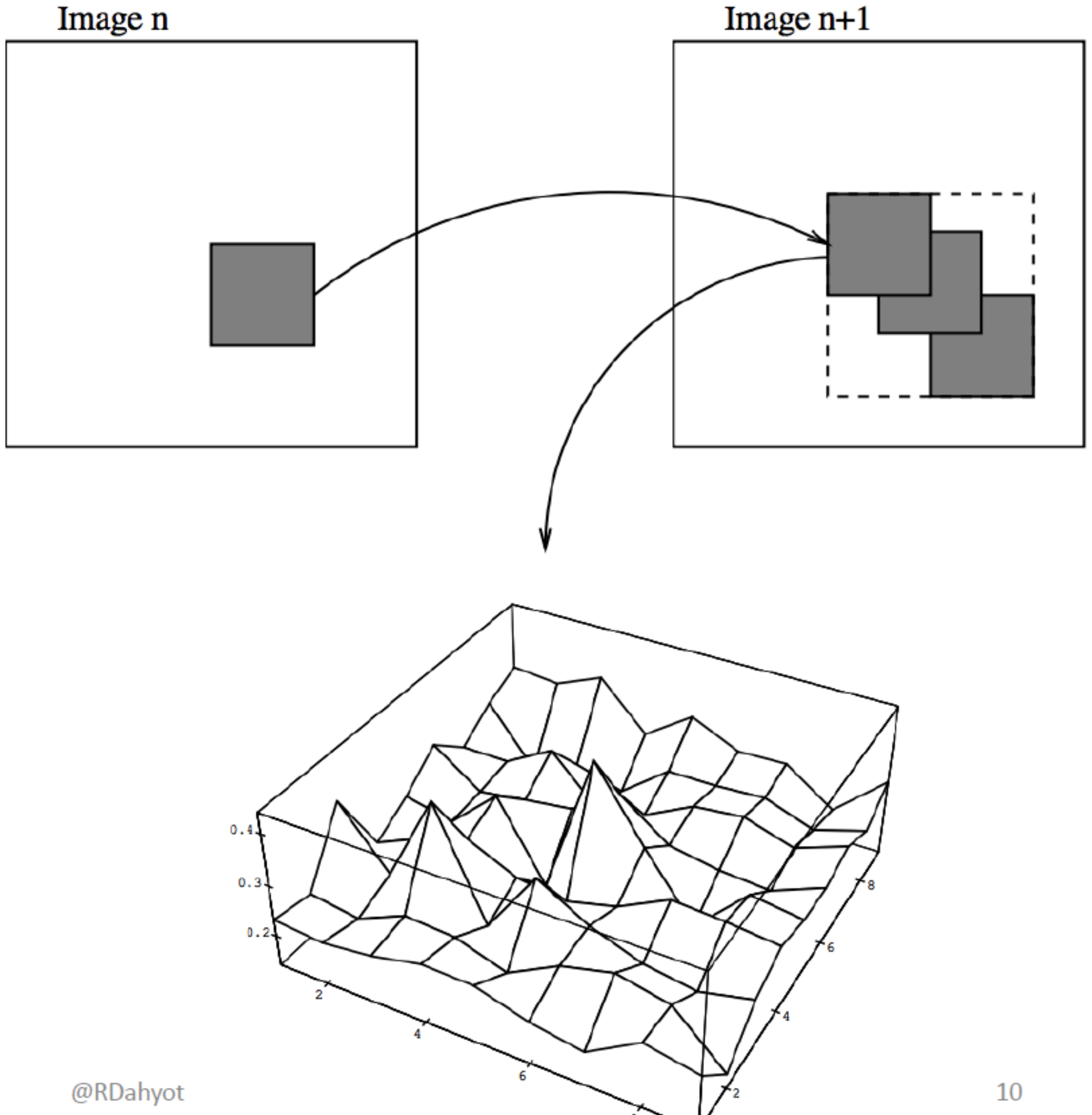


Simon Niklaus, Long Mai, and Feng Liu. **Video Frame Interpolation via Adaptive Separable Convolution**. IEEE ICCV 2017.

Applications – video compression

MPEG

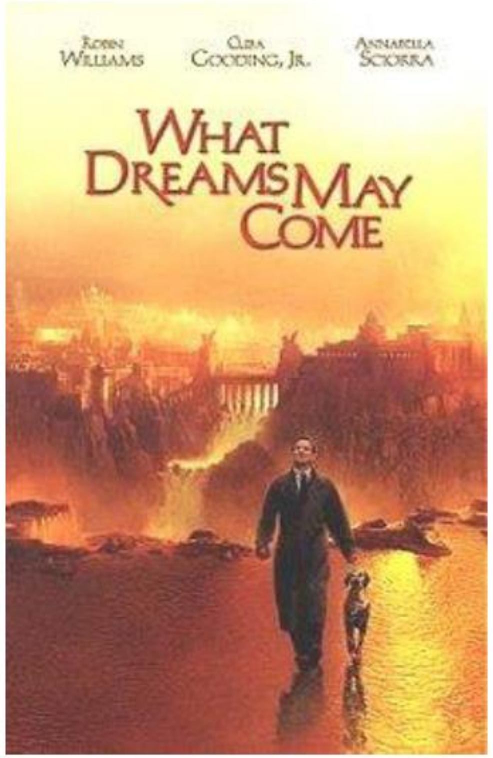
- Motion coding based on block matching



Applications – movie production

https://www.fxguide.com/fxfeatured/art_of_optical_flow/

Painterly effect



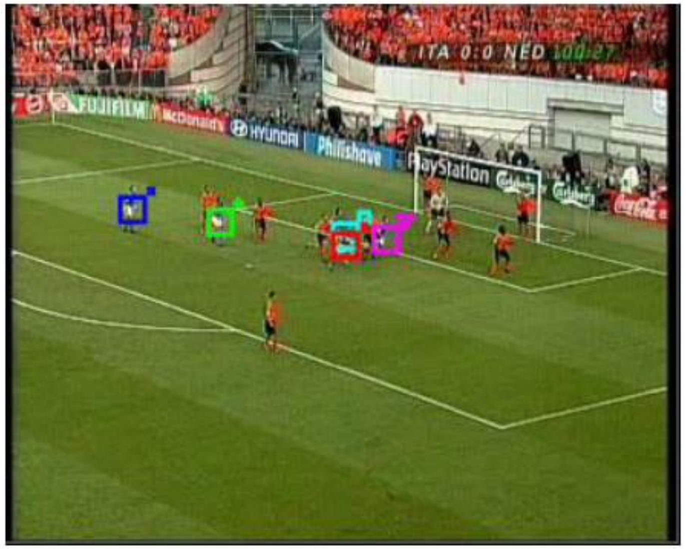
Applications – movie production

Bullet Time

Use optical flow to compute correspondence between different camera views. Allows smooth interpolation between views.

Applications – Object tracking

- Kalman Filter (online)
- Particle Filter (online)
- Camshift (online)
- Hidden Markov Models (offline)



[Off-line multiple object tracking using candidate selection and the Viterbi algorithm]

F. Pitie, S-A. Berrani, R. Dahyot and A. Kokaram, in IEEE International Conference on Image Processing (ICIP'05), Genoa, Italy.
DOI:10.1109/ICIP.2005.1530340

Summary

- Optical flow is a fundamental task in many computer vision systems
- Edge aware filters can replace costly global optimization
- Fast and accurate methods are critical
- Recent progress are based on deep learning and CNNs