ADVANCES IN COMPUTER VISION

MODULE 1: Geometry

CORRESPONDENCE I: OPTICAL FLOW



Prof. Vincent Sitzmann

Fredo Office Hours

- Fridays 1pm in Stata 32-D424 (D side)

Credits

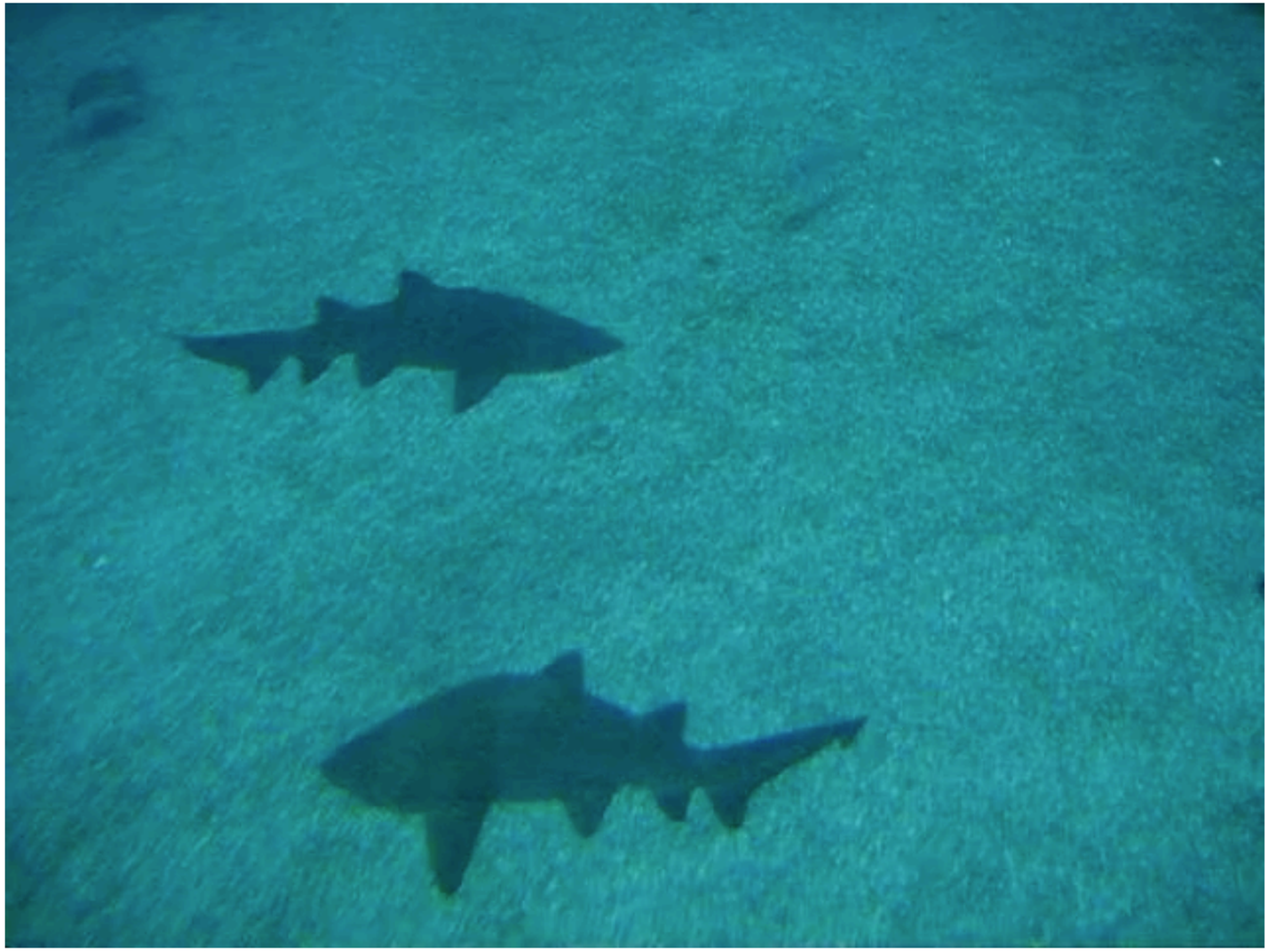
Base of this lecture: Deqing Sun & Charles Herrmann

Some slides, images, and videos from
- Dr. Ce Liu@Microsoft
- Dr. Huaizu Jiang@Northeastern
- Dr. David Fouhey@UMich
- Dr. Justin Johnson@UMich
- Dr. Svetlana Lazebnik@UIUC
- Dr. Shree K. Nayar@Columbia
- Dr. Jia Deng@Princeton
- Dr. Ming-Hsuan Yang@UC Merced
- Dr. Rick Szeliski's book
- Book by Antonio, Phillip, and Bill
- Dr. Christian Rupprecht@Oxford

Suggestions from Dr. Bill Freeman, Dr. Rick Szeliski, Dr. Noah Snavely and Dr. Junhwa Hur

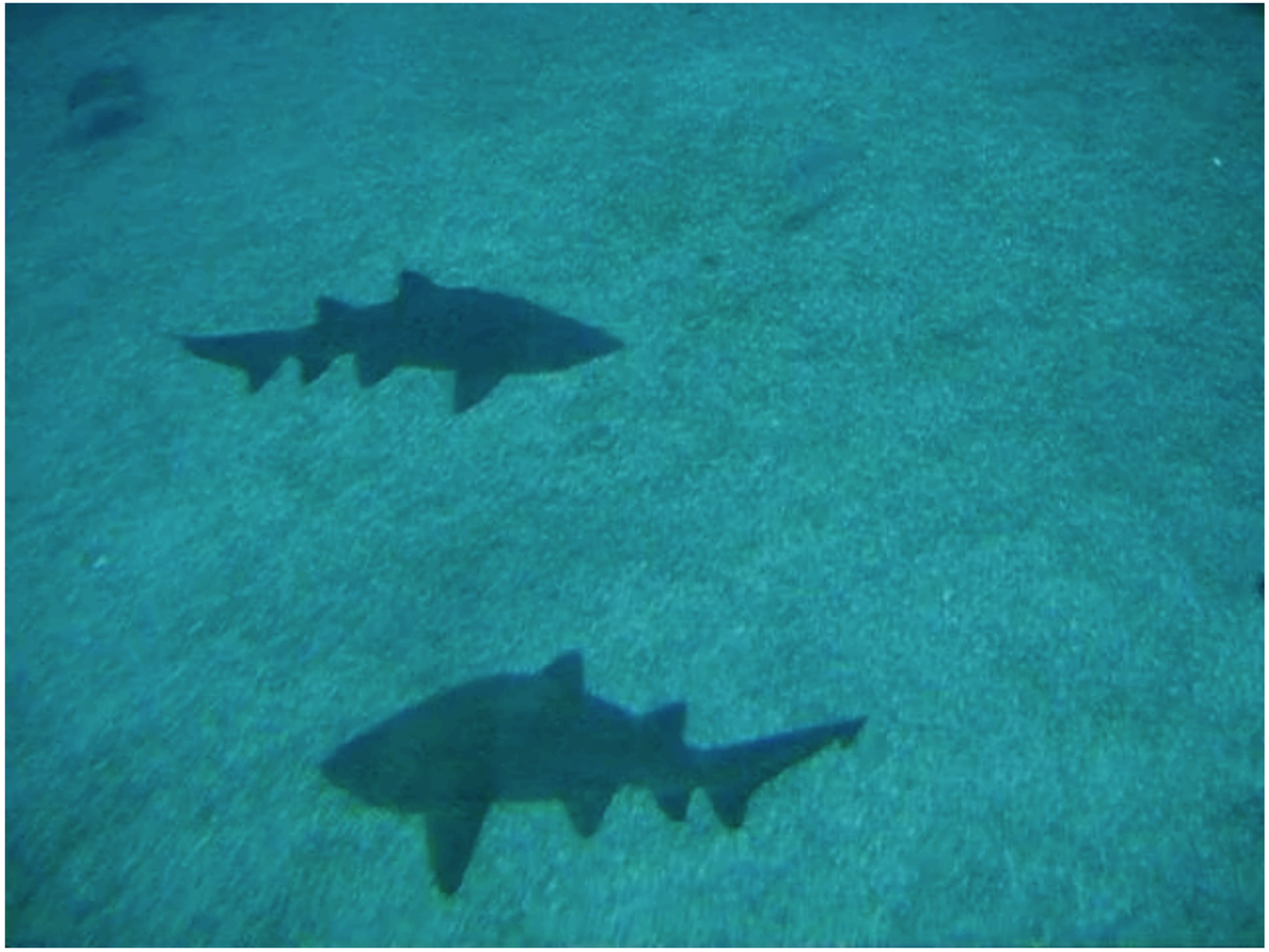
We live in a dynamic world

Perceiving, understanding and predicting motion is an important part of our daily lives



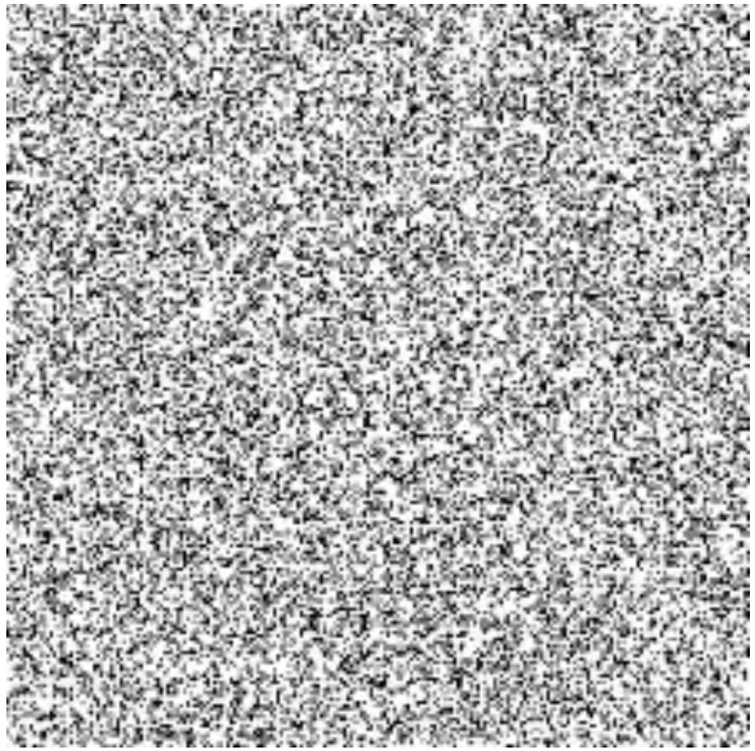
We live in a dynamic world

Perceiving, understanding and predicting motion is an important part of our daily lives



Motion and perceptual organization

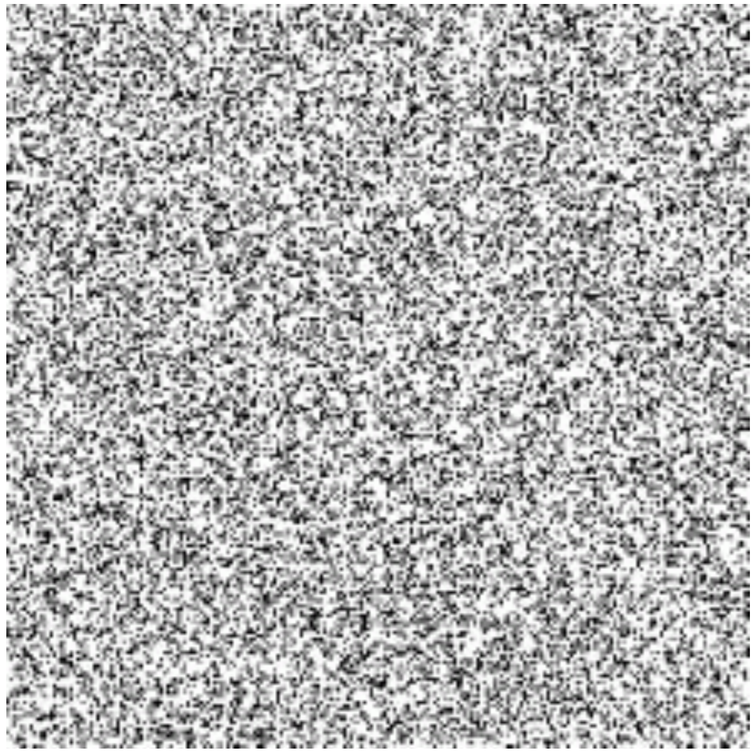
Sometimes motion is the only cue!



Slide Credit: S. Lazebnik, but idea of random dot sterogram is due to B. Julesz

Motion and perceptual organization

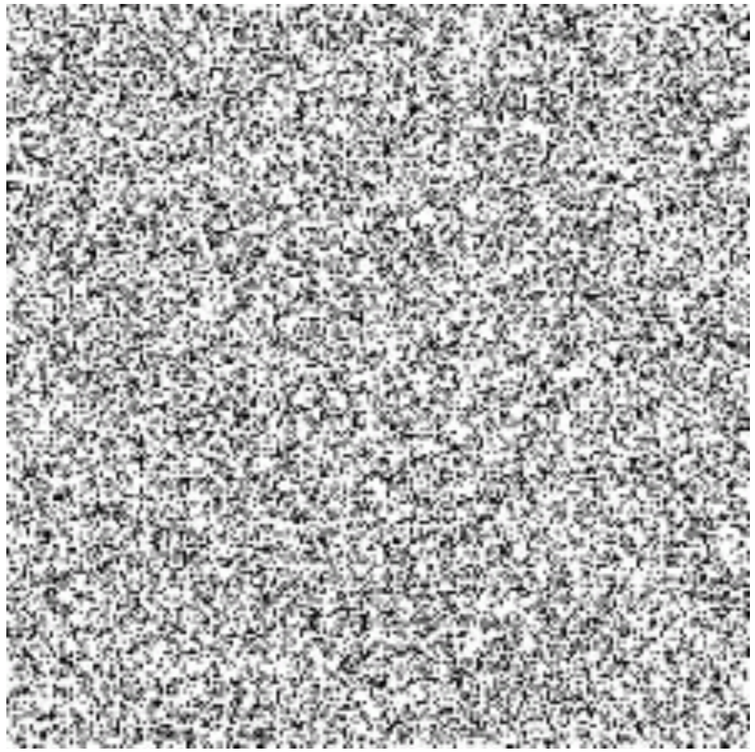
Sometimes motion is the only cue!



Slide Credit: S. Lazebnik, but idea of random dot sterogram is due to B. Julesz

Motion and perceptual organization

Sometimes motion is the only cue!



Slide Credit: S. Lazebnik, but idea of random dot sterogram is due to B. Julesz



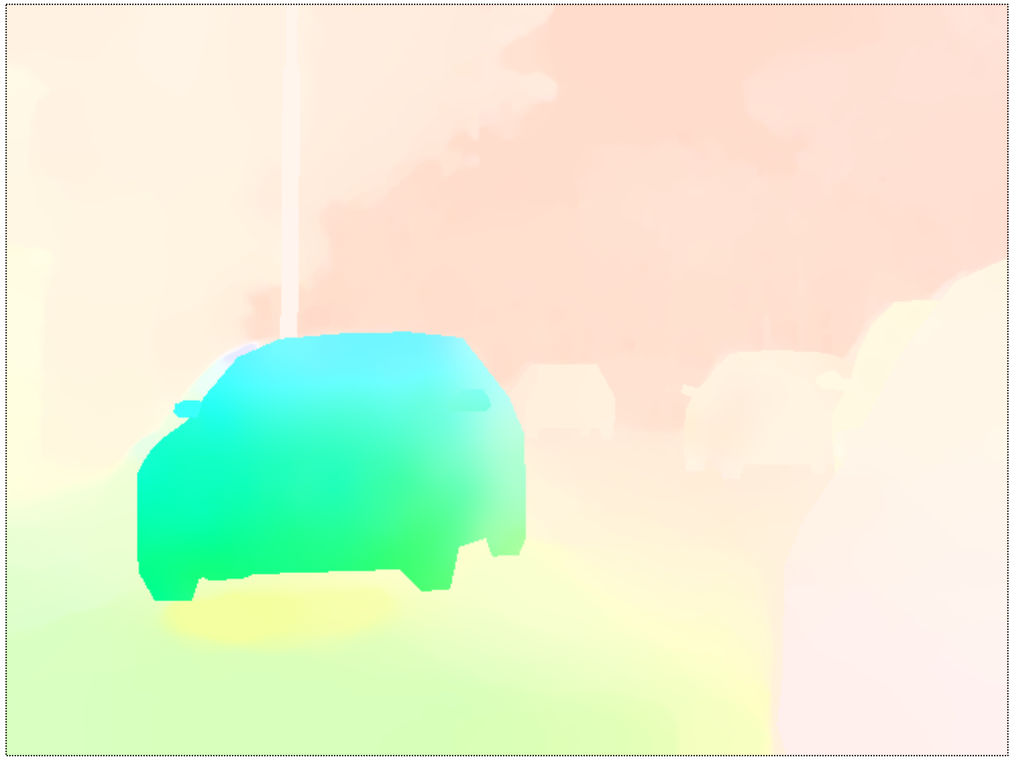


Optical flow: 2D motion of every pixel

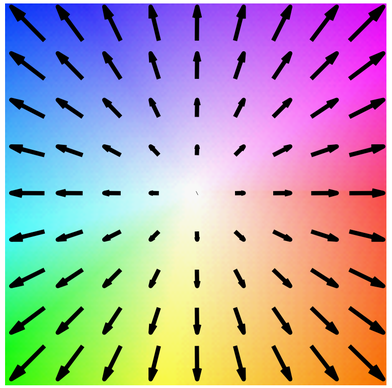
Usually for a pair of successive frames



Input [Liu *et al.* CVPR'08]



Optical flow (2D motion vector)



Color key
[Baker *et al.* IJCV'11]

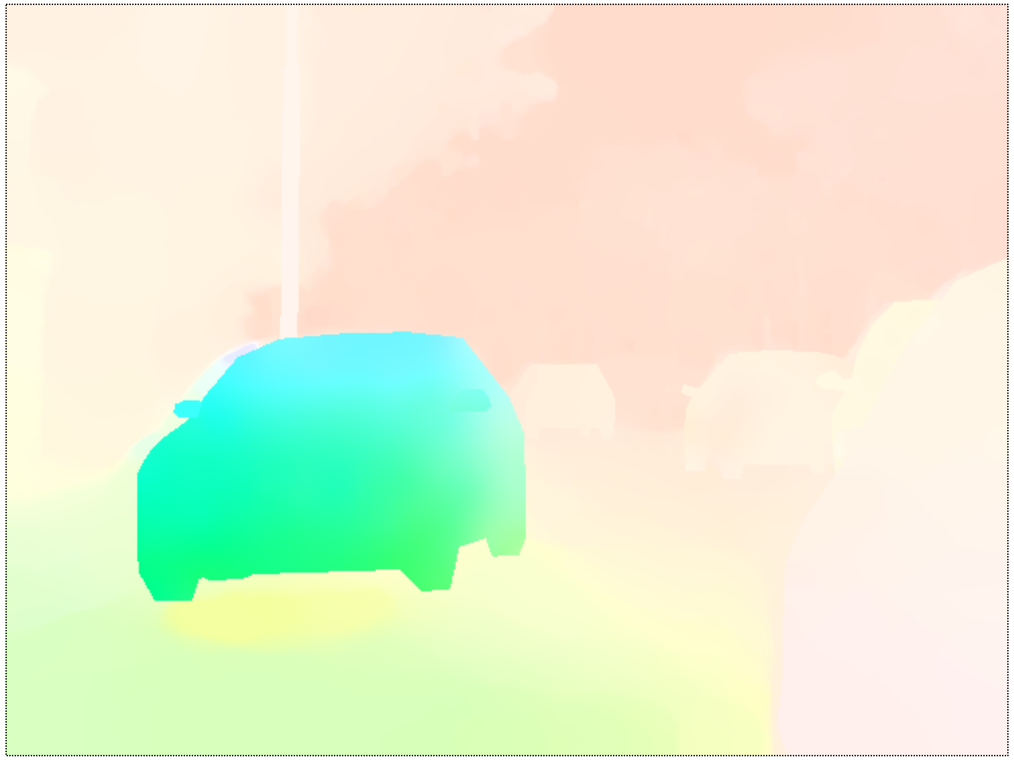
Slide credit: Deqing Sun

Optical flow: 2D motion of every pixel

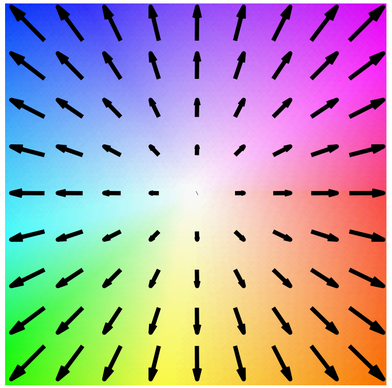
Usually for a pair of successive frames



Input [Liu *et al.* CVPR'08]



Optical flow (2D motion vector)

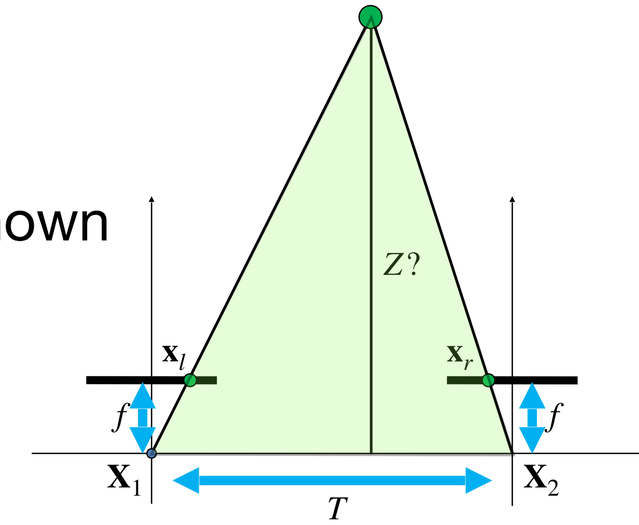


Color key
[Baker *et al.* IJCV'11]

Slide credit: Deqing Sun

Relation to stereo vision

- Disparity ~ optical flow in the case of static scenes and camera baseline
- Stereo is more constrained once cameras are calibrated: 1D vs 2D unknown
  - Only unknown per pixel is depth (or disparity)
  - Epipolar geometry

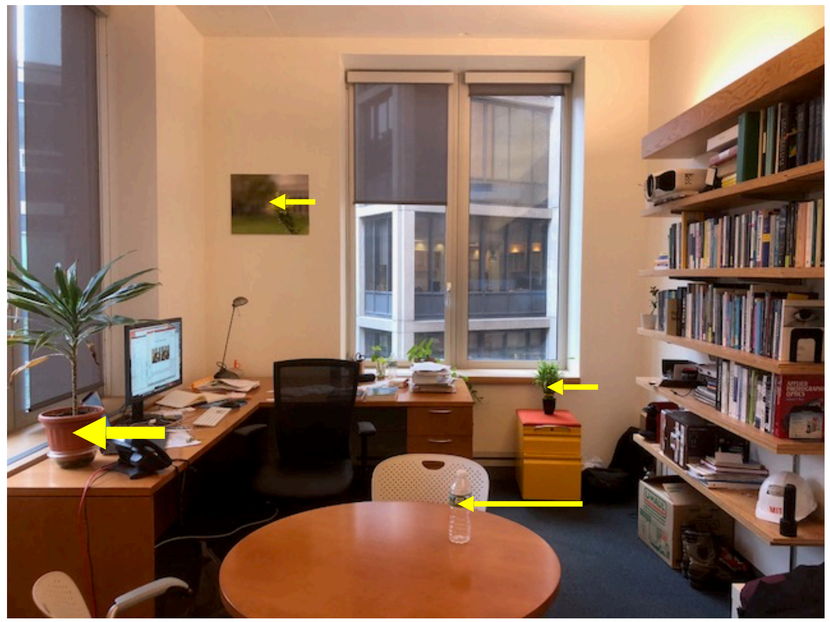


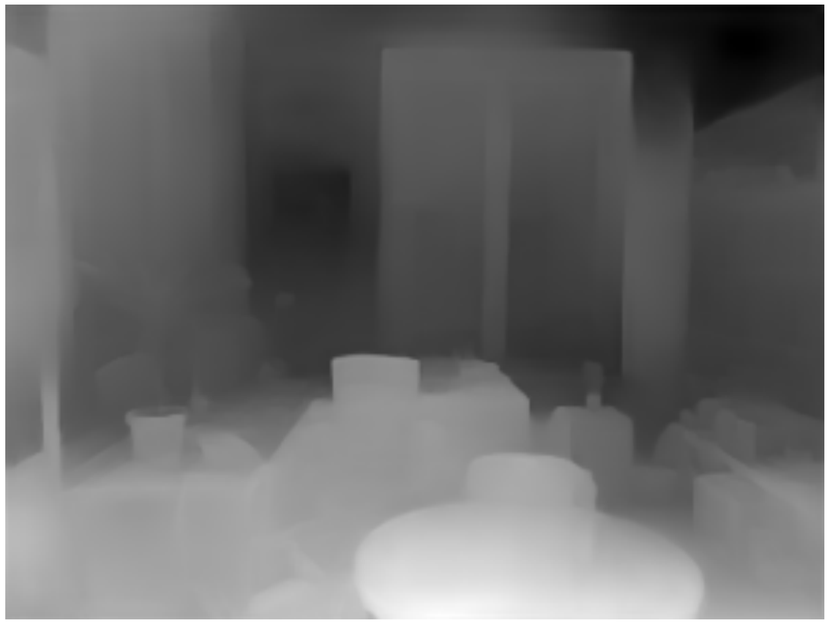
$I(x, y)$

$I'(x, y) = I(x + D(x, y), y)$

$D(x, y)$







(Why) is Optical flow interesting?

- Yes because it’s non-local yet fairly simple
- Yes because traditional method shed much understanding
- Yes because an example of dense (today) vs sparse (next week)
- Yes because not fully solved, especially not by universal convergence models

- No because it may be a fake task
- No because it’s ill-defined

Is optical flow a fake task?

- Downstream tasks:
  - Video editing, slow motion (frame interpolation), video super resolution, in-painting, special effects, video stabilization
  - Action recognition, object tracking and surveillance
  - Egomotion, SLAM
  - Connection to video compression
  - Medical imaging: very similar to dense correspondences
  - Optical mice
  - Non-destructive testing, analysis of machinery, physics (e.g. Fluid flow)
  - Generative video??
  - Biological cognition understanding, time to contact (looming)

Fundamental assumption: Brightness constancy

[Horn & Schunck AI'81]

Assume a pixel at (x, y, t) with intensity $I(x, y, t)$ has moved by $(\Delta x, \Delta y)$ in space during a timestep Δt.

Color constancy assumption (CCA):

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$



Fundamental assumption: Brightness constancy

[Horn & Schunck AI'81]

Assume a pixel at (x, y, t) with intensity $I(x, y, t)$ has moved by $(\Delta x, \Delta y)$ in space during a timestep Δt.

Color constancy assumption (CCA):

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$



Fundamental assumption: Brightness constancy

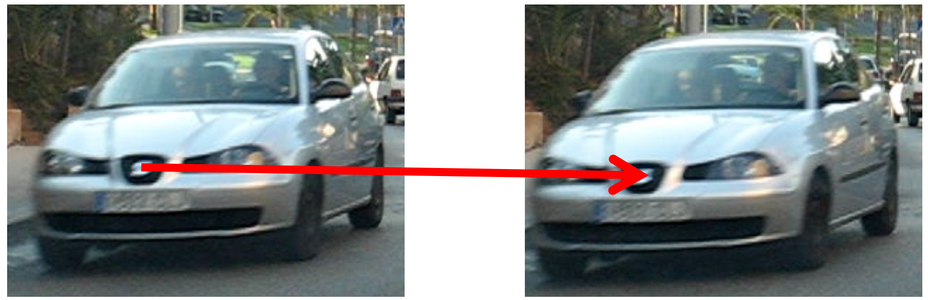
[Horn & Schunck AI'81]

Assume a pixel at (x, y, t) with intensity $I(x, y, t)$ has moved by $(\Delta x, \Delta y)$ in space during a timestep Δt.

Color constancy assumption (CCA):

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$





First image (t)

Second image (t+1)



Berthold Horn

Determining Optical Flow

Berthold K.P. Horn and Brian G. Schunck
Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

ABSTRACT
Optical flow cannot be computed locally, since only one independent measurement is available from the image sequence at a point, while the flow velocity has two components. A second constraint is needed. A method for finding the optical flow pattern is presented which assumes that the apparent velocity of the brightness pattern varies smoothly almost everywhere in the image. An iterative implementation is shown which successfully computes the optical flow for a number of synthetic image sequences. The algorithm is robust in that it can handle image sequences that are quantized rather coarsely in space and time. It is also insensitive to quantization of brightness levels and additive noise. Examples are included where the assumption of smoothness is violated at singular points or along lines in the image.

1. Introduction

Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of objects and the viewer [6, 7]. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement [8]. Discontinuities in the optical flow can help in segmenting images into regions that correspond to different objects [27]. Attempts have been made to perform such segmentation using differences between successive image frames [15, 16, 17, 20, 25]. Several papers address the problem of recovering the motions of objects relative to the viewer from the optical flow [10, 18, 19, 21, 29]. Some recent papers provide a clear exposition of this enterprise [30, 31]. The mathematics can be made rather difficult, by the way, by choosing an inconvenient coordinate system. In some cases information about the shape of an object may also be recovered [3, 18, 19].

These papers begin by assuming that the optical flow has already been determined. Although some reference has been made to schemes for comput-

Artificial Intelligence **17** (1981) 185–203
0004–3702/81/0000–0000/$02.50 © North-Holland

Horn & Schunck Optical Flow

Taylor expansion:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) \approx$$





Horn & Schunck Optical Flow

Taylor expansion:

$$I(x+\Delta x, y+\Delta y, t+\Delta t) \approx I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t$$

With Color Constancy Assumption:

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad \text{or} \quad \frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} = 0$$





Horn & Schunck Optical Flow

Taylor expansion:

$$I(x+\Delta x, y+\Delta y, t+\Delta t) \approx I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t$$

With Color Constancy Assumption:

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad \text{or} \quad \frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} = 0$$

Using $\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right]^T$ and $\mu = \left[\frac{\Delta x}{\Delta t}, \frac{\Delta y}{\Delta t}\right]^T$

yields the **motion constraint equation:**

$$\nabla \mathbf{I}\mu = -\frac{\delta I}{\delta t}$$





Thinking About Horn & Schunck Optical Flow

$$\nabla \mathbf{I}\mu = -\frac{\delta I}{\delta t}$$

To use for optical flow estimation, put into a loss function:

$$E = \min_{\mu} \left[(\nabla^T \mathbf{I}_1 \mu - (\mathbf{I}_2 - \mathbf{I}_1))^2 \right]$$

It’s convex (can be solved via gradient descent or closed-form)!

Thinking About Horn & Schunck Optical Flow

$$\nabla \mathbf{I}\mu = -\frac{\delta I}{\delta t}$$

To use for optical flow estimation, put into a loss function:

$$E = \min_{\mu} \left[(\nabla^T \mathbf{I}_1 \mu - (\mathbf{I}_2 - \mathbf{I}_1))^2 \right]$$

It’s convex (can be solved via gradient descent or closed-form)!

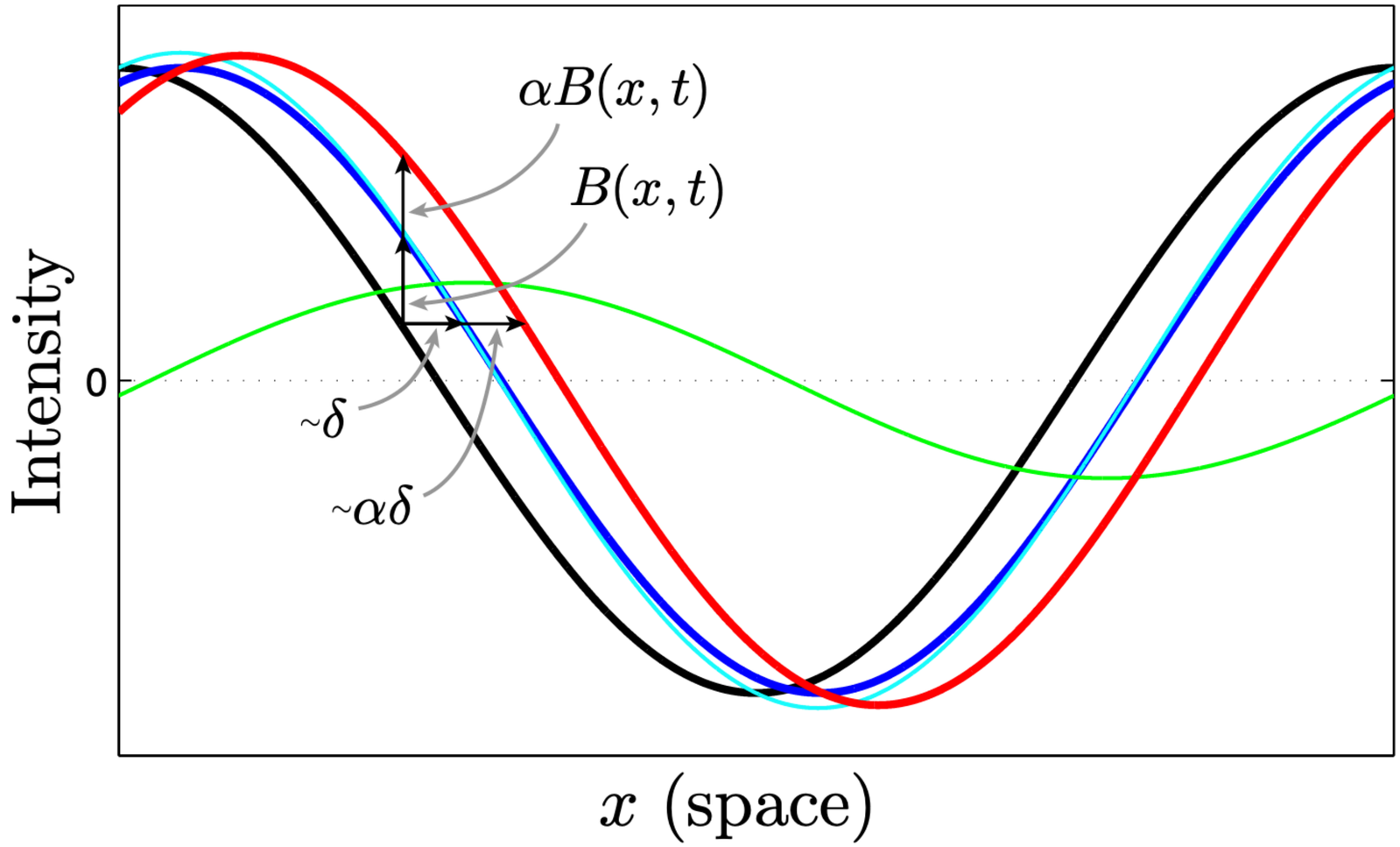
But remember we had to simplify the problem to get there:
Color constancy, linearization with Taylor expansion

Thinking About Horn & Schunck Optical Flow

$$\nabla \mathbf{I}\mu = -\frac{\delta I}{\delta t}$$

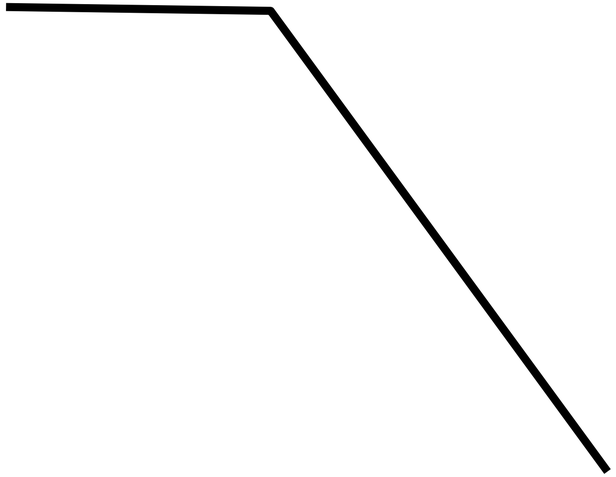
Remarkable: Only the *image gradient* matters for time evolution!
Why? Think about it.

Recall Eulerian Video Magnification



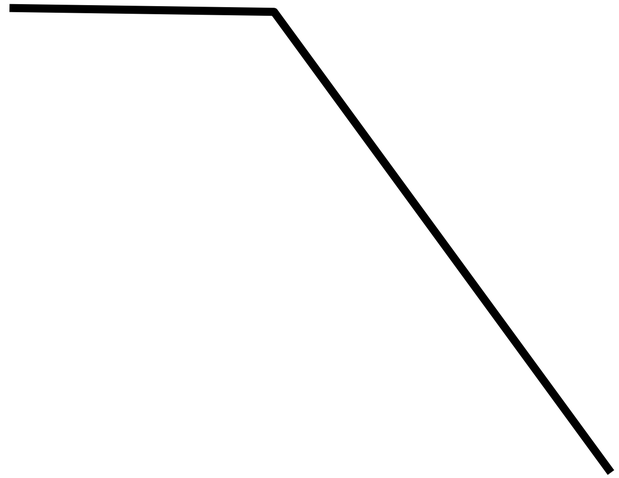
$f(x)$ $f(x+\delta)$ $f(x)+\delta\frac{\partial f(x)}{\partial x}$ $B(x,t)$ $f(x)+(1+\alpha)B(x,t)$

Estimating Motion in Direction of Constant Brightness: The Aperture Problem



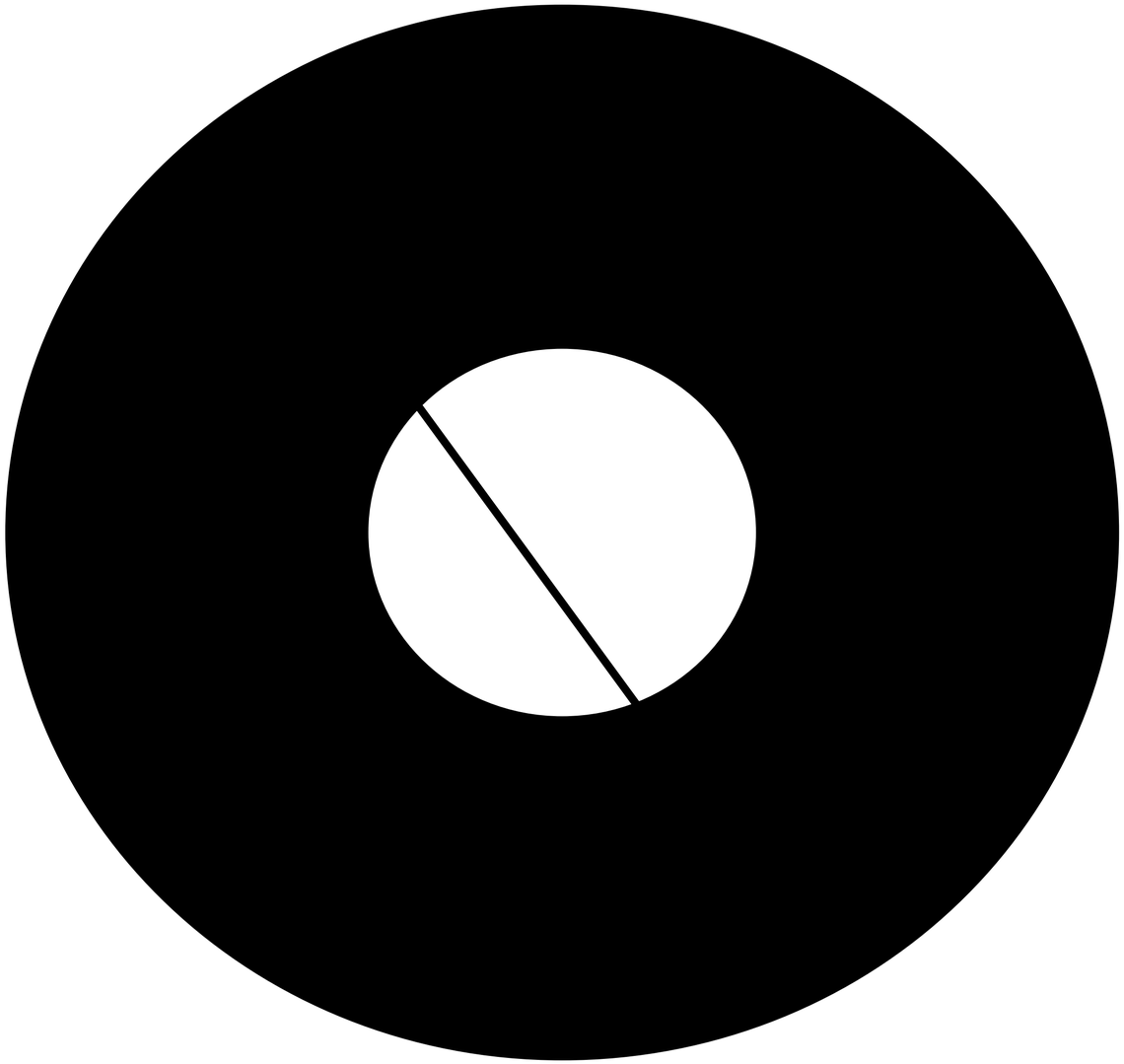
Slide Credit: S. Lazebnik

Estimating Motion in Direction of Constant Brightness: The Aperture Problem



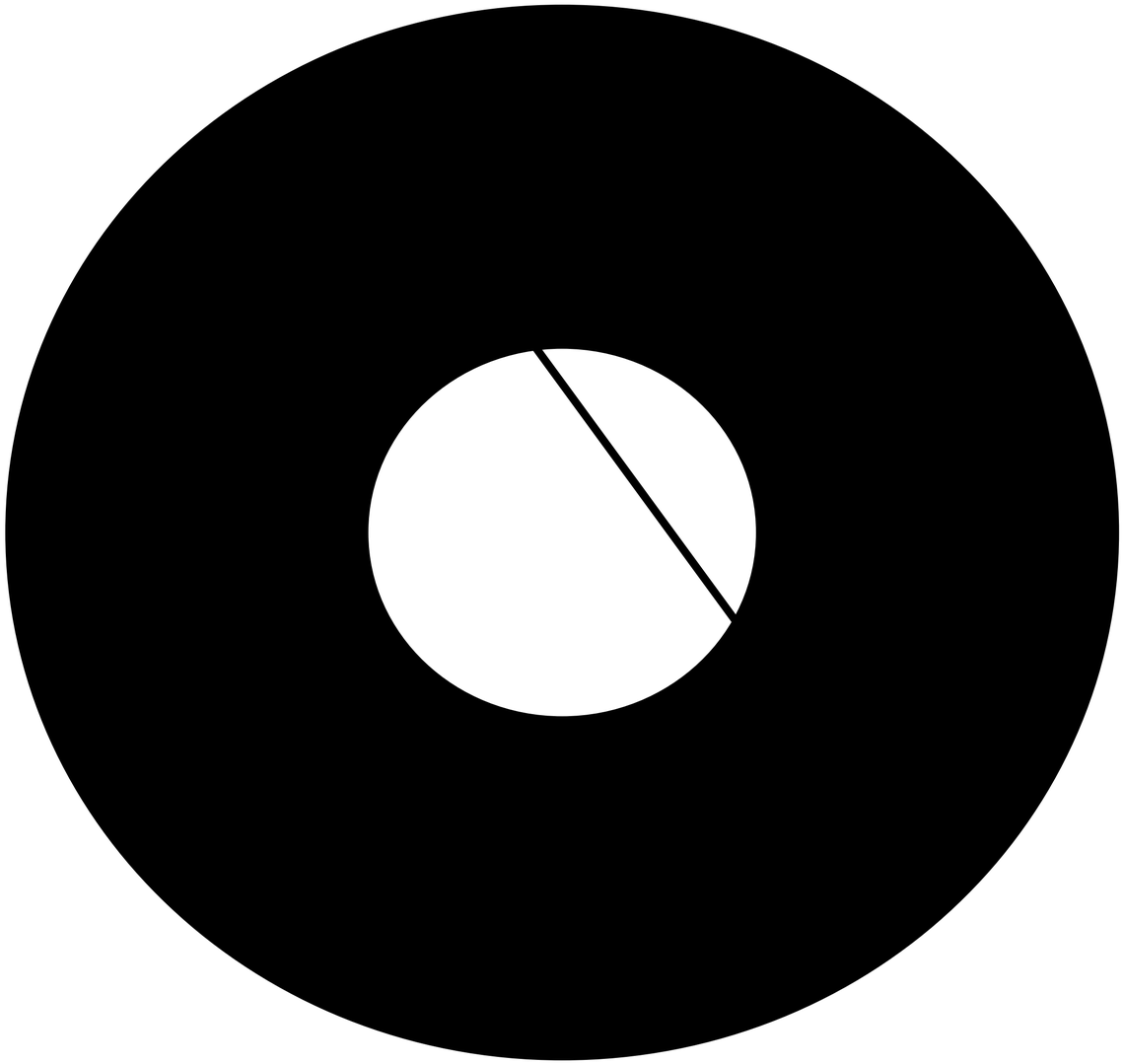
Slide Credit: S. Lazebnik

Estimating Motion in Direction of Constant Brightness: The Aperture Problem



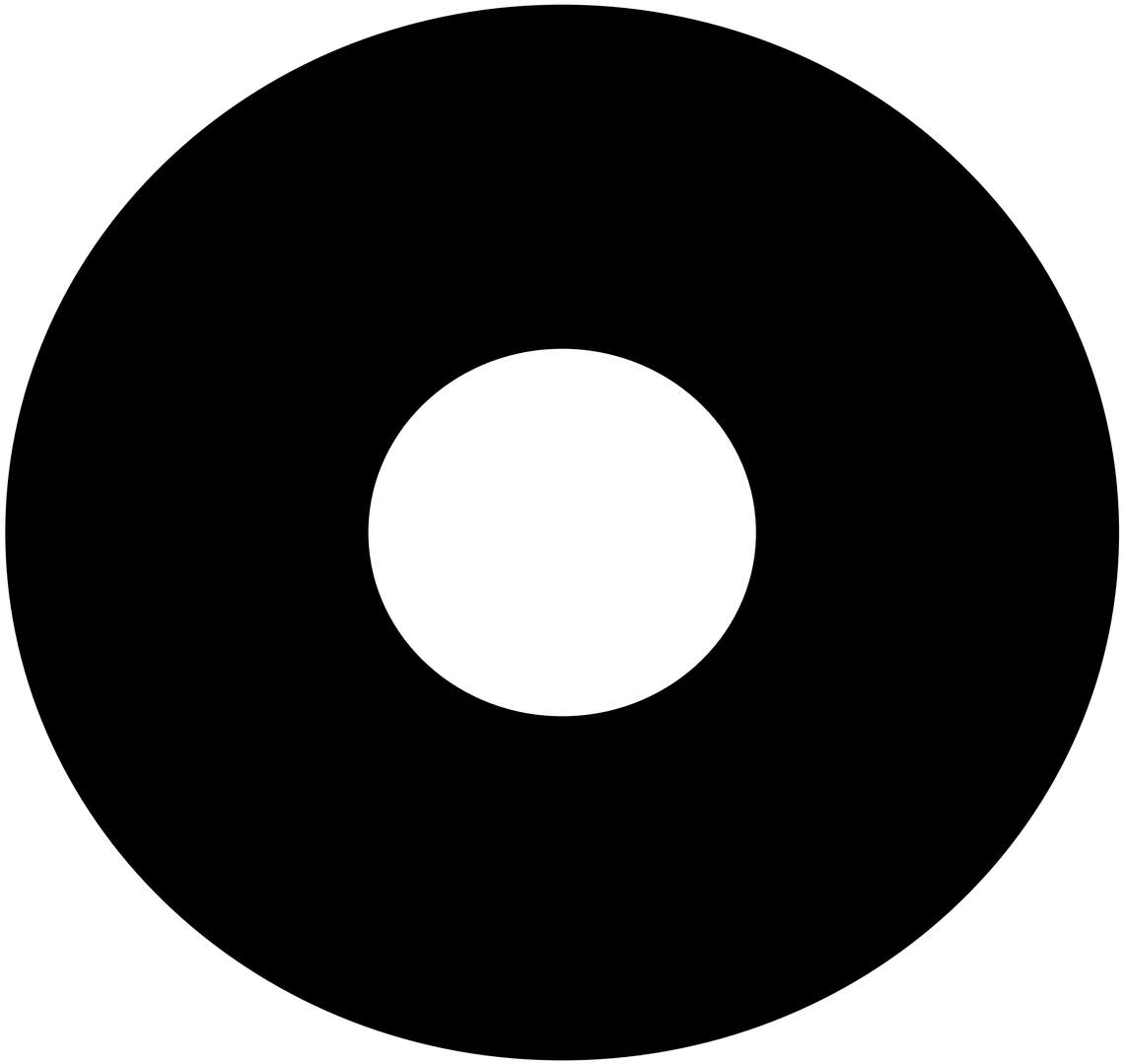
Slide Credit: S. Lazebnik

Estimating Motion in Direction of Constant Brightness: The Aperture Problem



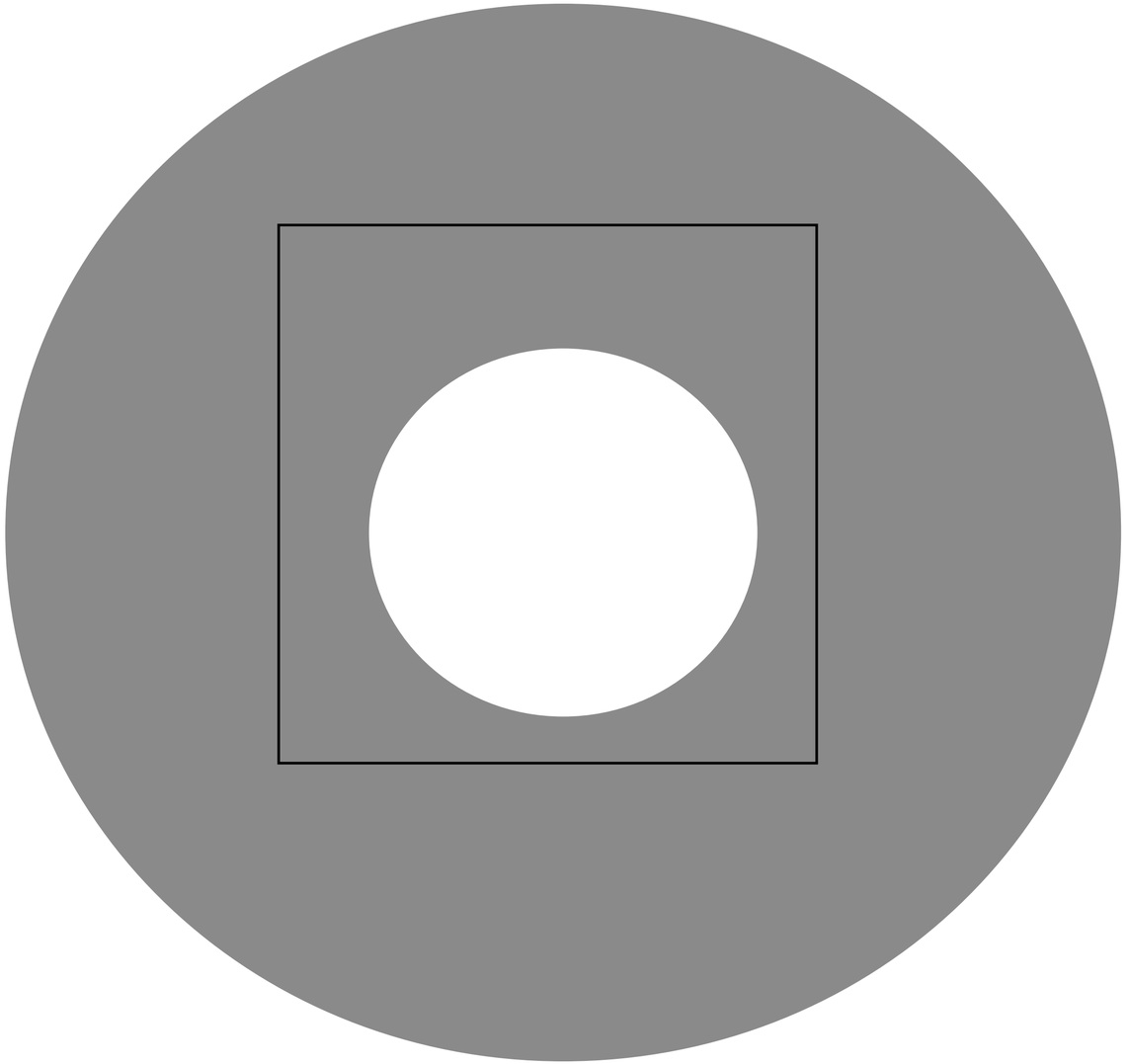
Slide Credit: S. Lazebnik

Other invisible flow



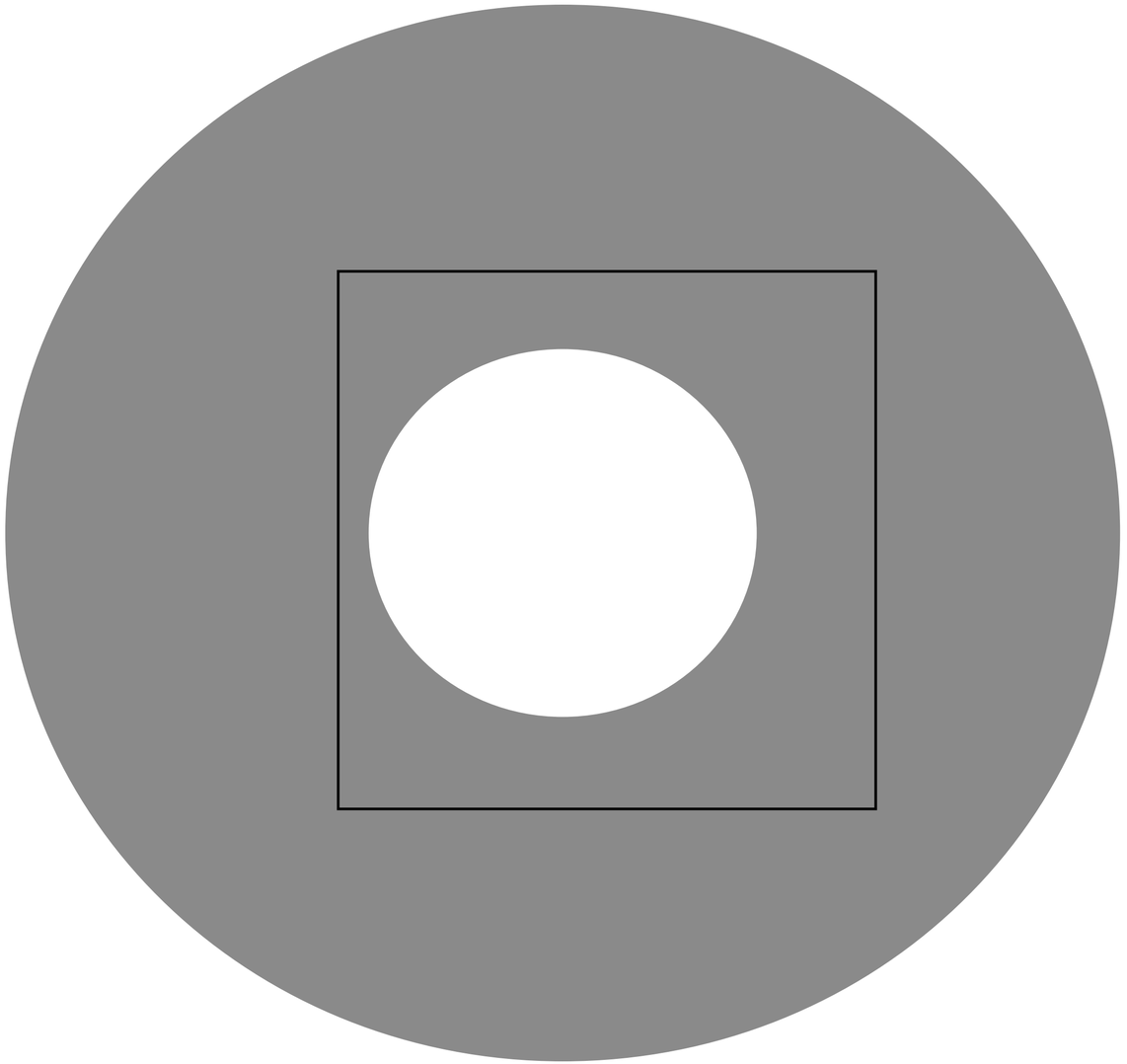
Slide credit: D Fouhey & J Johnson

Other invisible flow



Slide credit: D Fouhey & J Johnson

Other invisible flow



Slide credit: D Fouhey & J Johnson

What do you perceive?

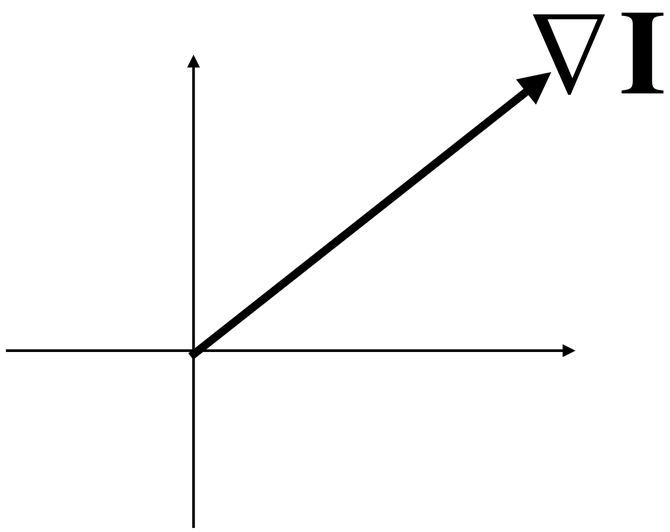


What do you perceive?



The math warned us

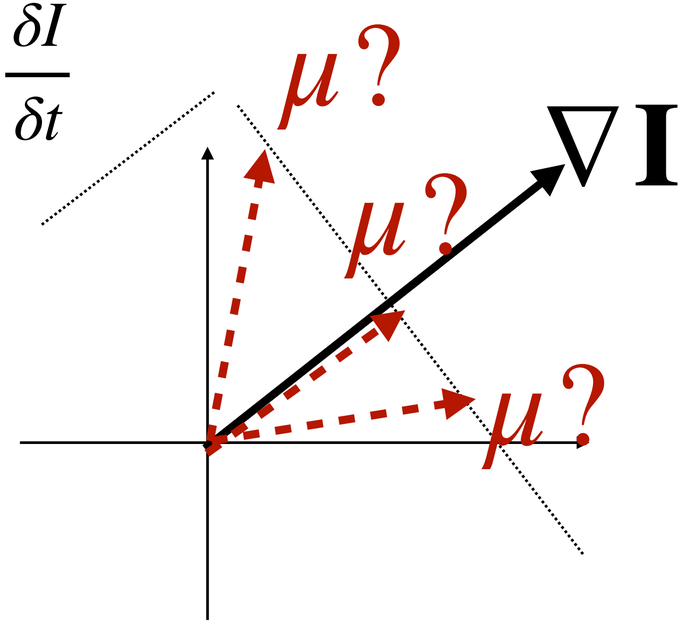
$$\nabla \mathbf{I} \,.\, \mu = -\frac{\delta I}{\delta t}$$



The math warned us

$$\nabla \mathbf{I} \,.\, \mu = -\frac{\delta I}{\delta t}$$

$$-\frac{1}{||\nabla \mathbf{I}||}\frac{\delta I}{\delta t}$$



Optical Flow – The Beginnings

[Horn & Schunck AI'81]



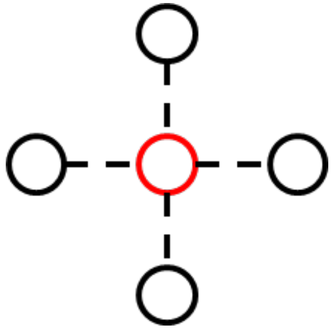
Raw estimate

Optical Flow – The Beginnings

[Horn & Schunck AI'81]



Raw estimate



Prior term (Smoothness)

Optimization/energy minimization

[Horn & Schunck AI'81]



Data term (Constancy)

Prior term (Smoothness)

$$E = \min_{\mu} \left[(\nabla^T \mathbf{I}_1 \mu - (\mathbf{I}_2 - \mathbf{I}_1))^2 + \alpha(|\nabla \mu_x|^2 + |\nabla \mu_y|^2) \right]$$

Optical Flow – The Beginnings

[Horn & Schunck AI'81]



Raw estimate

[Horn & Schunck AI'81]



Smoothed estimate

Horn & Schunck on a more recent scene





Horn & Schunck on a more recent scene





Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$



Image 1



Image 2

Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$



Image 1



Image 2

Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right) \right)^2$$



Image 1



Image 2

Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$

We kept color constancy but dropped linearization

(Becomes more Lagrangian!)



Image 1



Image 2

Comparing pixel colors directly



Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$



Image 1



Image 2

Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$



Image 1



Image 2

Matching-based motion estimation

Generally: Energy to measure similarity between a pixel in image 1 with pixels in image 2:

$$\min_{\mathbf{w_p}} \left(I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w_p}\right)\right)^2$$



Image 1



Image 2

Resolving ambiguities: Lucas-Kanade

Insight: Pixels in the same patch generally share the same motion

$$\min_{\mathbf{w_p}} \sum_{\mathbf{q} \in N_p} \left(I_t(\mathbf{q}) - I_{t+1}\left(\mathbf{q} + \mathbf{w_p}\right) \right)^2$$



Image 1



Image 2

Resolving ambiguities: Lucas-Kanade

Insight: Pixels in the same patch generally share the same motion

$$\min_{\mathbf{w_p}} \sum_{\mathbf{q} \in N_p} \left(I_t(\mathbf{q}) - I_{t+1}\left(\mathbf{q} + \mathbf{w_p}\right) \right)^2$$



Image 1



Image 2

Resolving ambiguities: Lucas-Kanade

Insight: Pixels in the same patch generally share the same motion

$$\min_{\mathbf{w_p}} \sum_{\mathbf{q} \in N_p} \left(I_t(\mathbf{q}) - I_{t+1}\left(\mathbf{q} + \mathbf{w_p}\right) \right)^2$$



Image 1



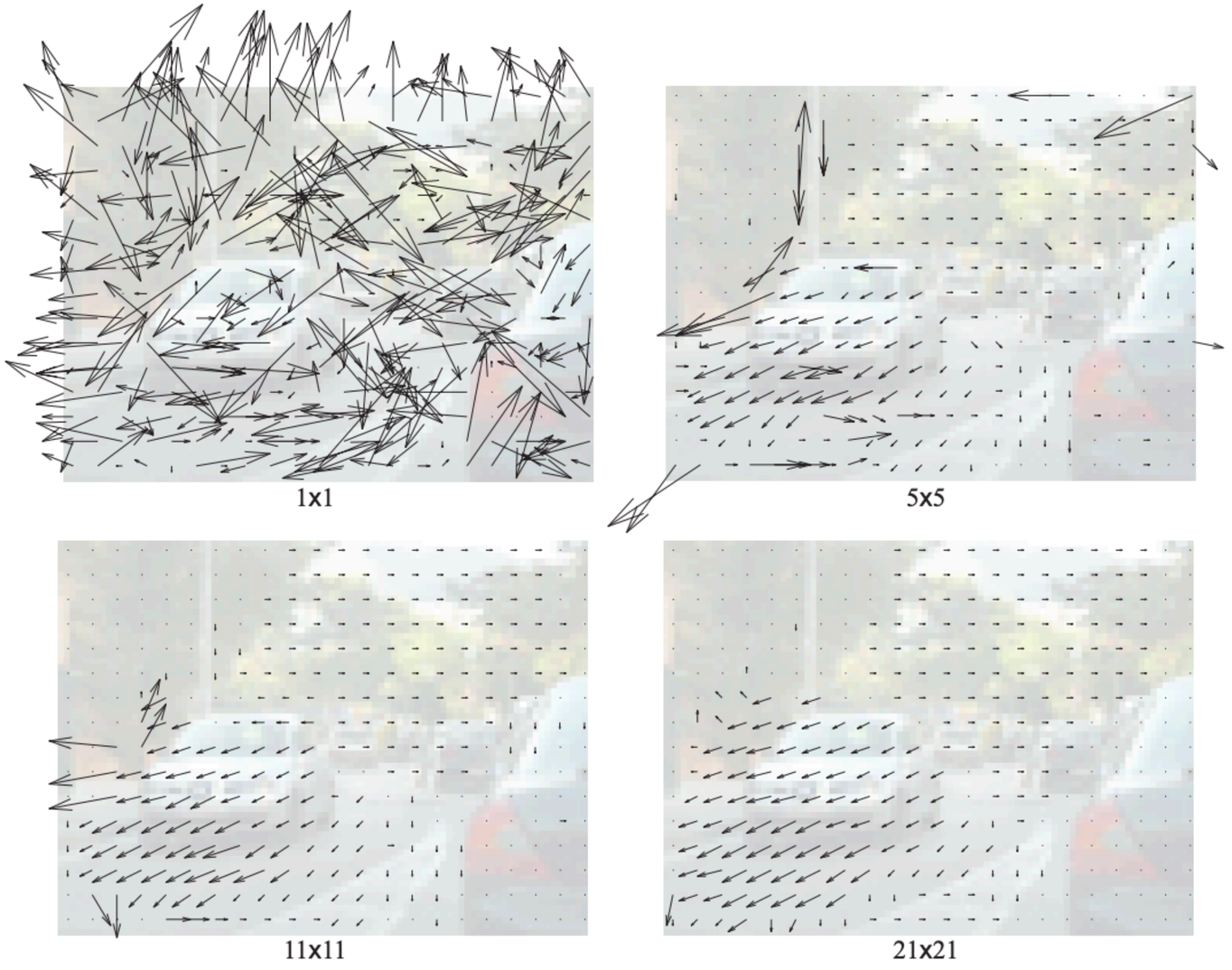
Image 2

Lukas Kanade

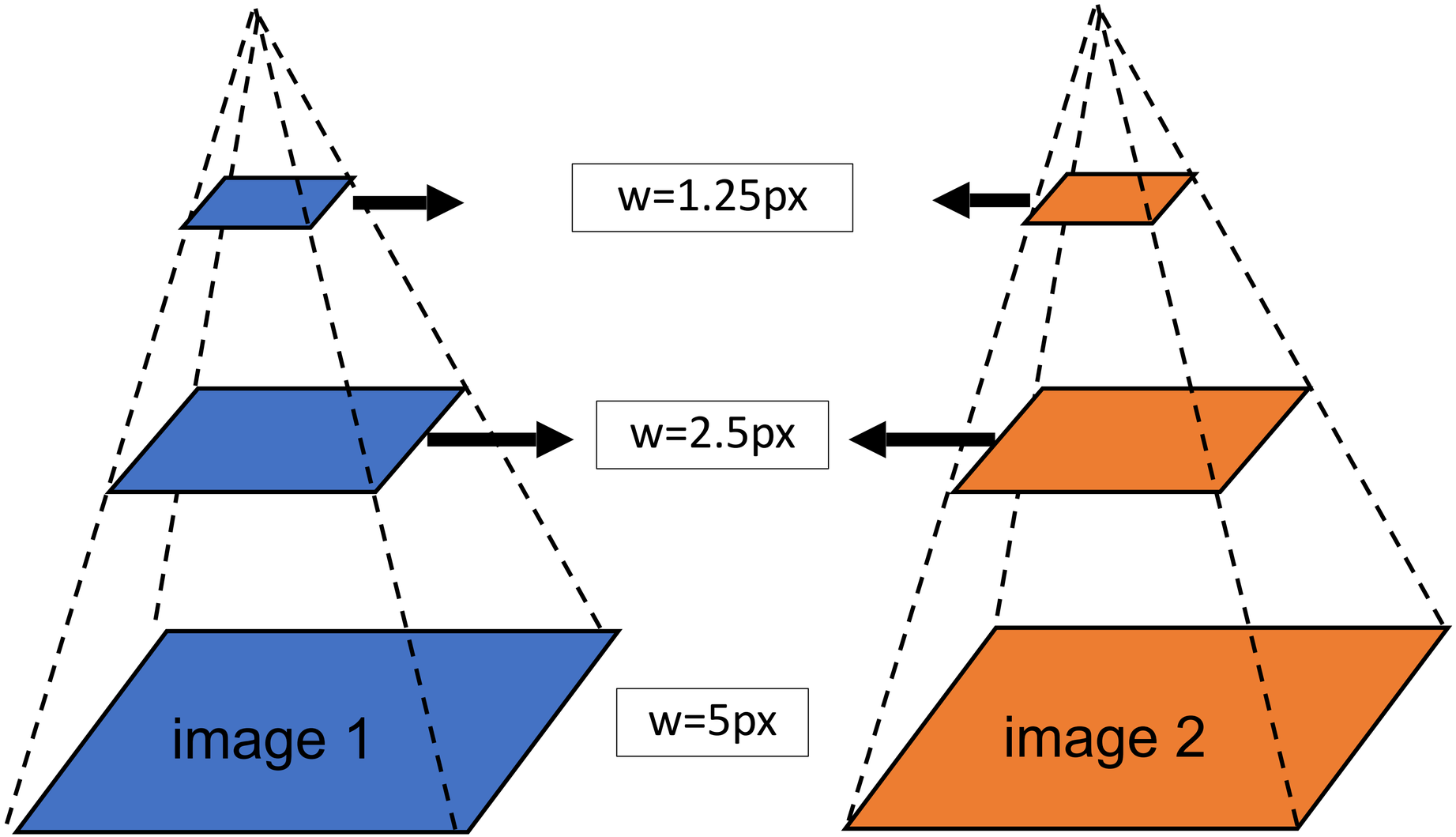
- Per-patch linear system
  - Essentially aggregate info from multiple pixels to 1/ resolve aperture problem 2/ regularize
  - Can be seen as per-patch 1-step Horn-Schunk with strong constant smoothness
- Assumes small motion (~1 pixel max)

- Later extended to use coarse-to-fine pyramids

Effect of patch size



Coarse-to-fine iterative estimation



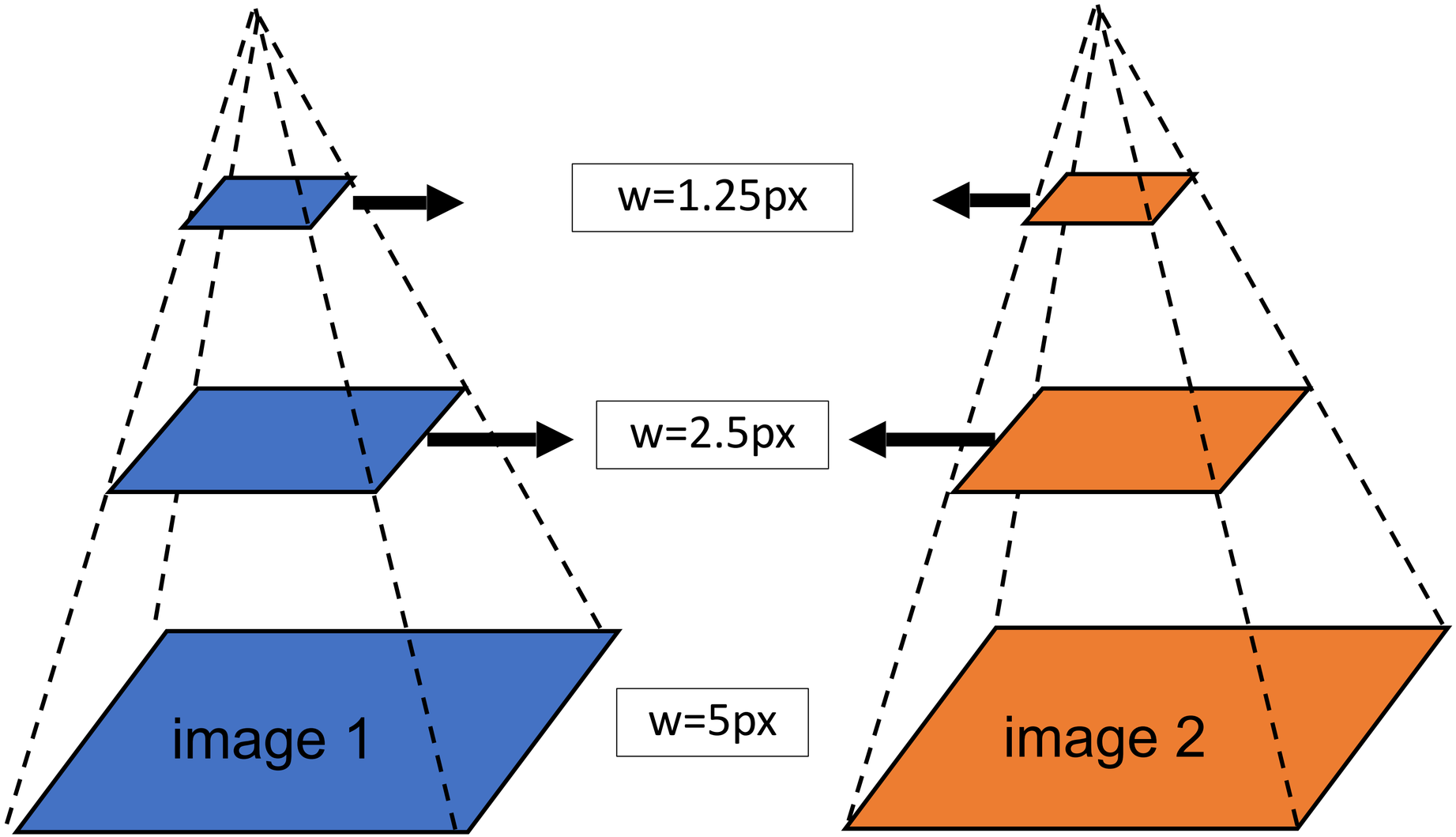






Slide credit: S. Lazebnik

Coarse-to-fine iterative estimation



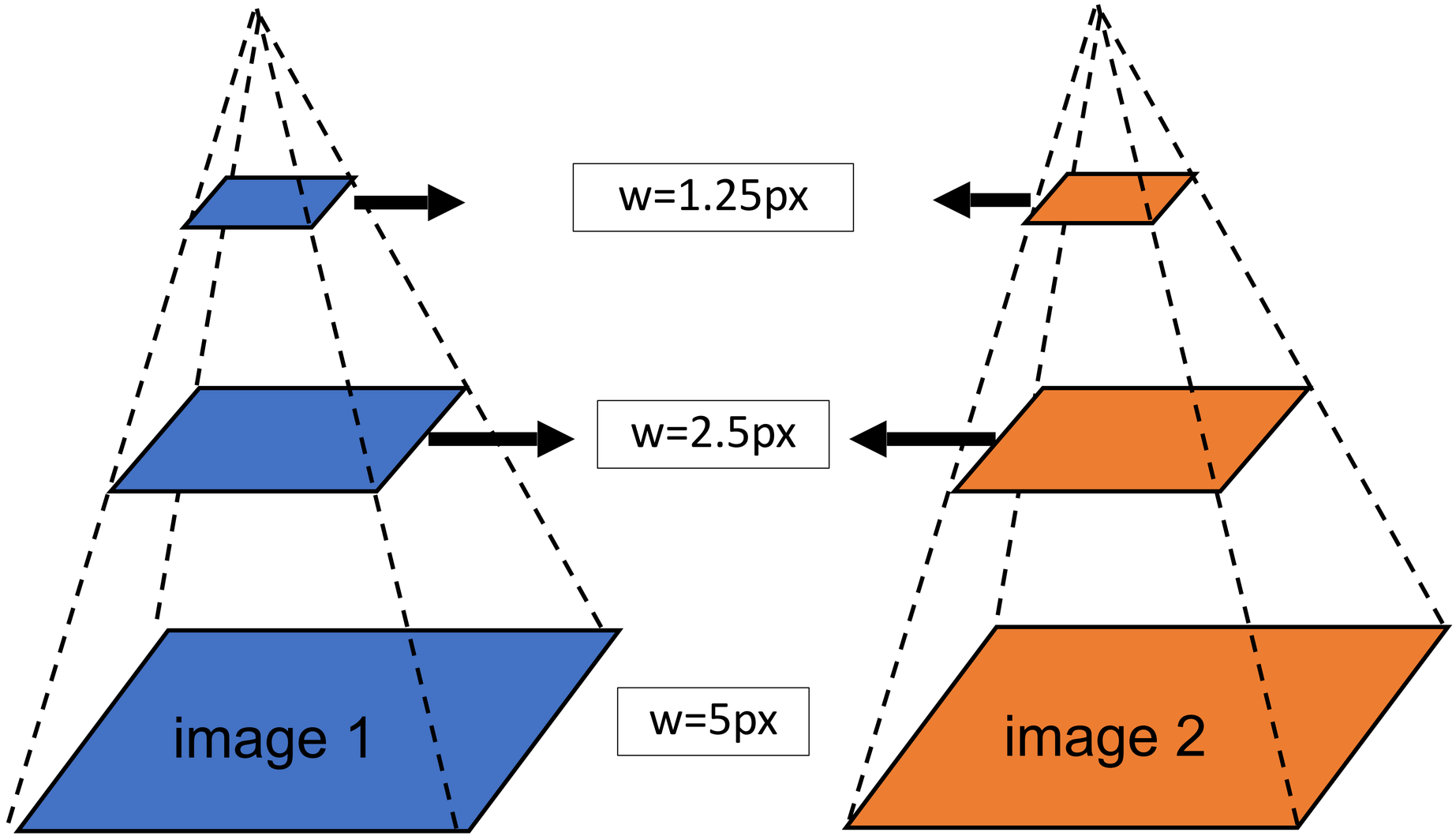






Slide credit: S. Lazebnik

Coarse-to-fine iterative estimation





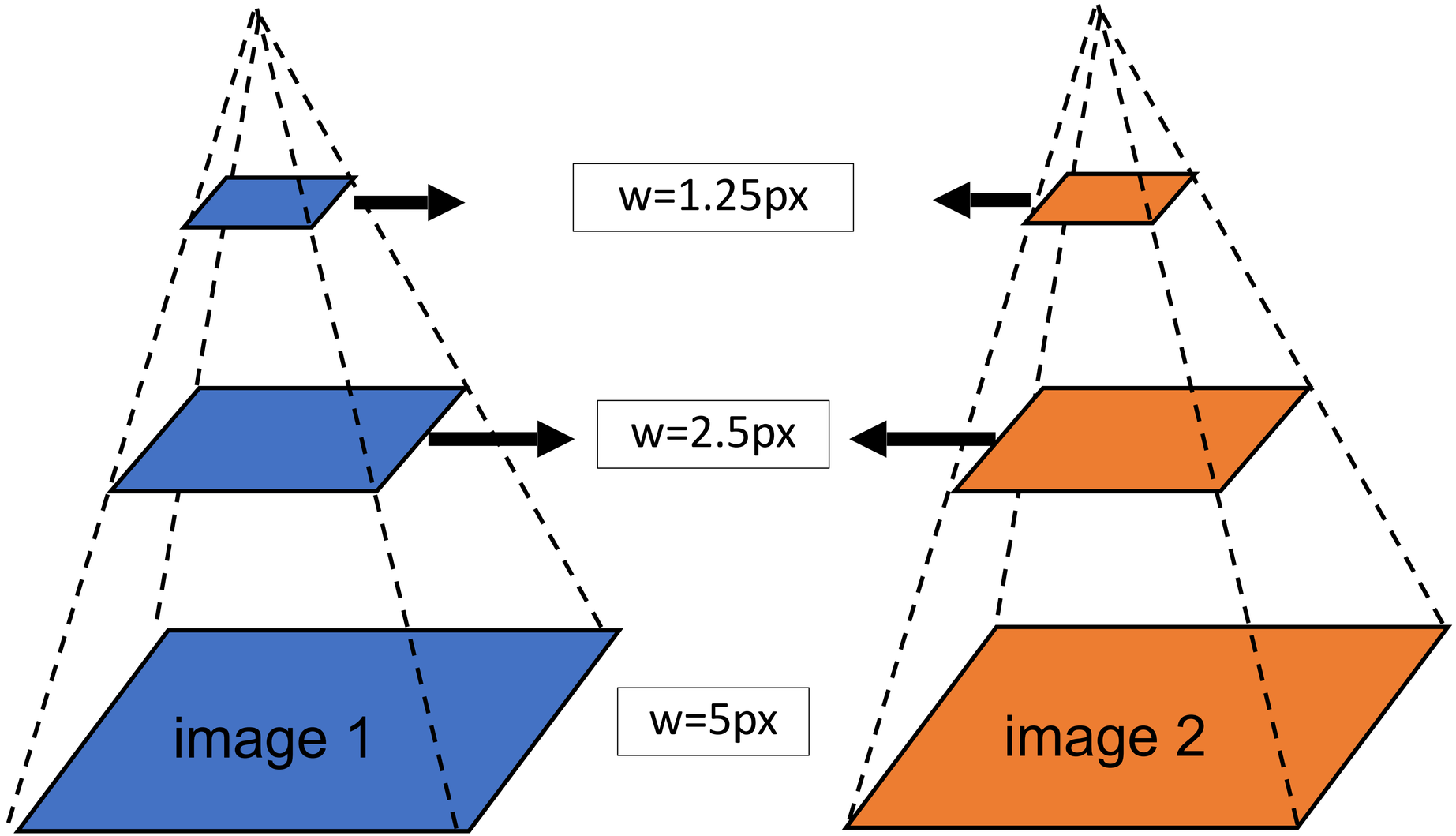




Start from top or bottom?

Slide credit: S. Lazebnik

Coarse-to-fine iterative estimation



How to use estimates from the upper level?

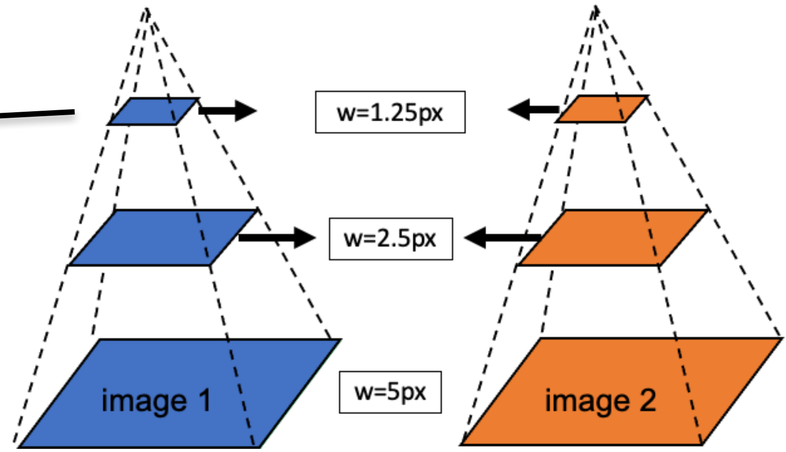
Slide credit: S. Lazebnik

Coarse-to-fine iterative estimation

Current estimate

$$I_{t+1}\left(\mathbf{q}+\mathbf{w_p}\right) = I_{t+1}\left(\mathbf{q}+\mathbf{w_p}^{k}+\delta\mathbf{w_p}\right) = \mathbf{I_w}(\mathbf{q}+\delta\mathbf{w_p})$$

Small increment

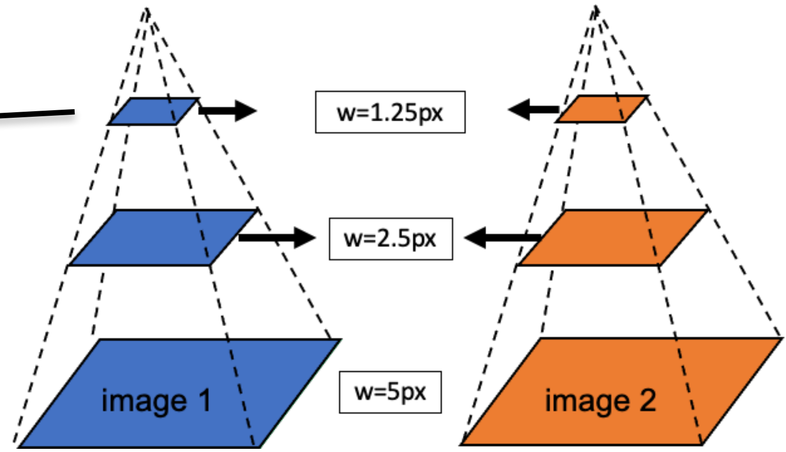


Coarse-to-fine iterative estimation

Current estimate

$$I_{t+1}\left(\mathbf{q}+\mathbf{w_p}\right) = I_{t+1}\left(\mathbf{q}+\mathbf{w_p}^k+\delta\mathbf{w_p}\right) = \mathbf{I_w}(\mathbf{q}+\delta\mathbf{w_p})$$

Small increment



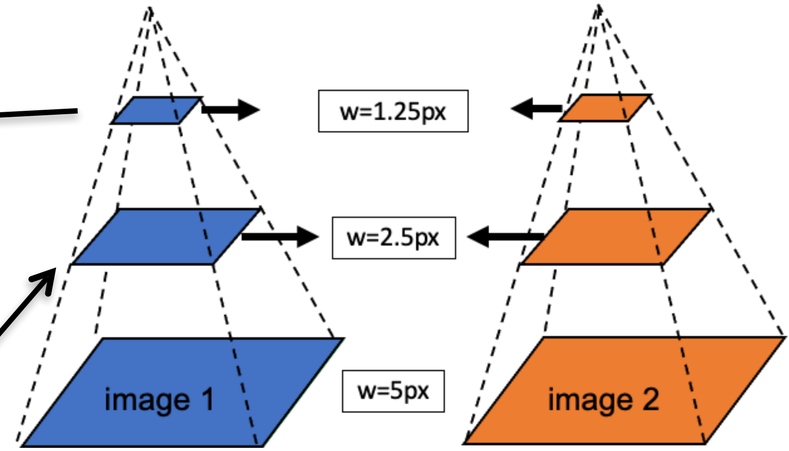
Warped image:

Coarse-to-fine iterative estimation

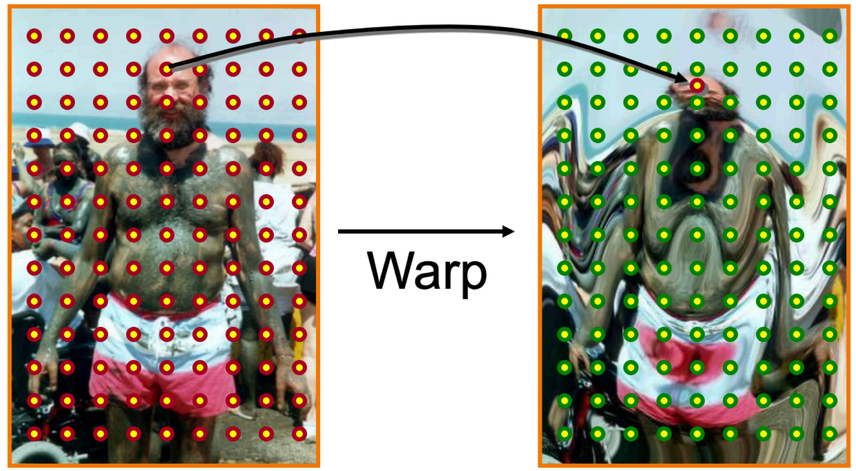
Current estimate

$$I_{t+1}\left(\mathbf{q}+\mathbf{w}_{\mathbf{p}}\right)=I_{t+1}\left(\mathbf{q}+\mathbf{w}_{\mathbf{p}}{}^{k}+\delta\mathbf{w}_{\mathbf{p}}\right)=\mathbf{I}_{\mathbf{w}}(\mathbf{q}+\delta\mathbf{w}_{\mathbf{p}})$$

Small increment



Warped image: $I_{\mathrm{w}}(\mathbf{q})=I_{t+1}\left(\mathbf{q}+\mathbf{w}_{\mathbf{p}}^{k}\right)$



Source image

Destination image

Warping operation

$$I_{\mathrm{w}}(\mathbf{q}) = I_{t+1}\left(\mathbf{q} + \mathbf{w}_{\mathbf{p}}^{k}\right)$$



Input images *t* and *t*+1



Image *t* and warped image

Warping operation

$$I_{\mathrm{w}}(\mathbf{q}) = I_{t+1}\left(\mathbf{q} + \mathbf{w}_{\mathbf{p}}^{k}\right)$$



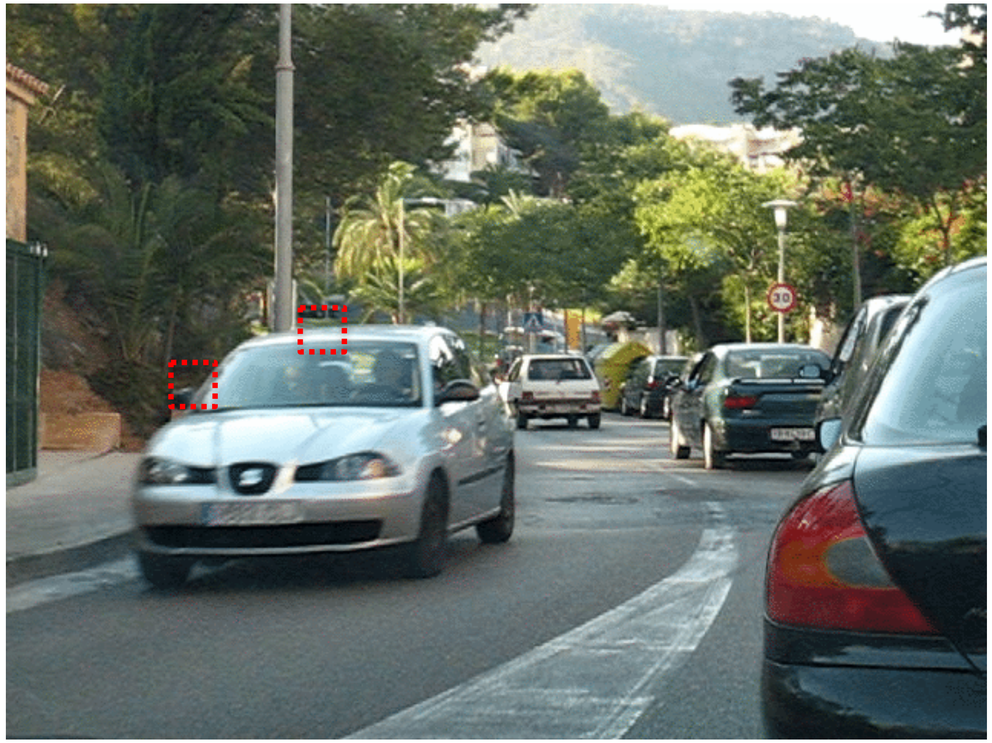
Input images *t* and *t*+1

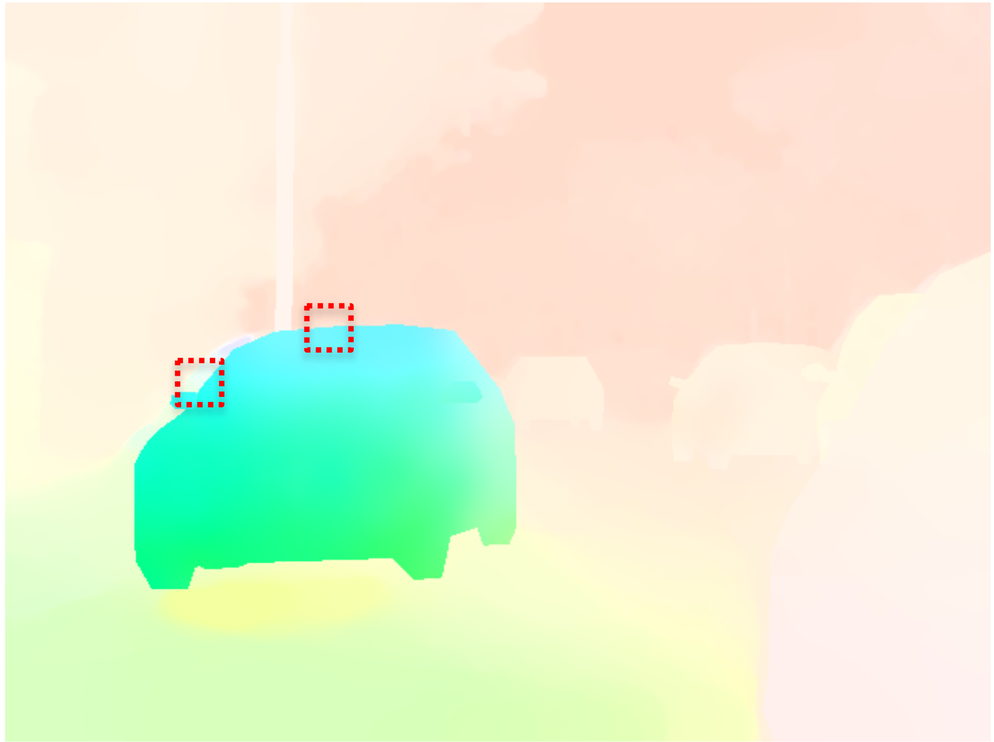


Image *t* and warped image

Issue: Motion Boundaries

No matches can be found at motion boundaries…

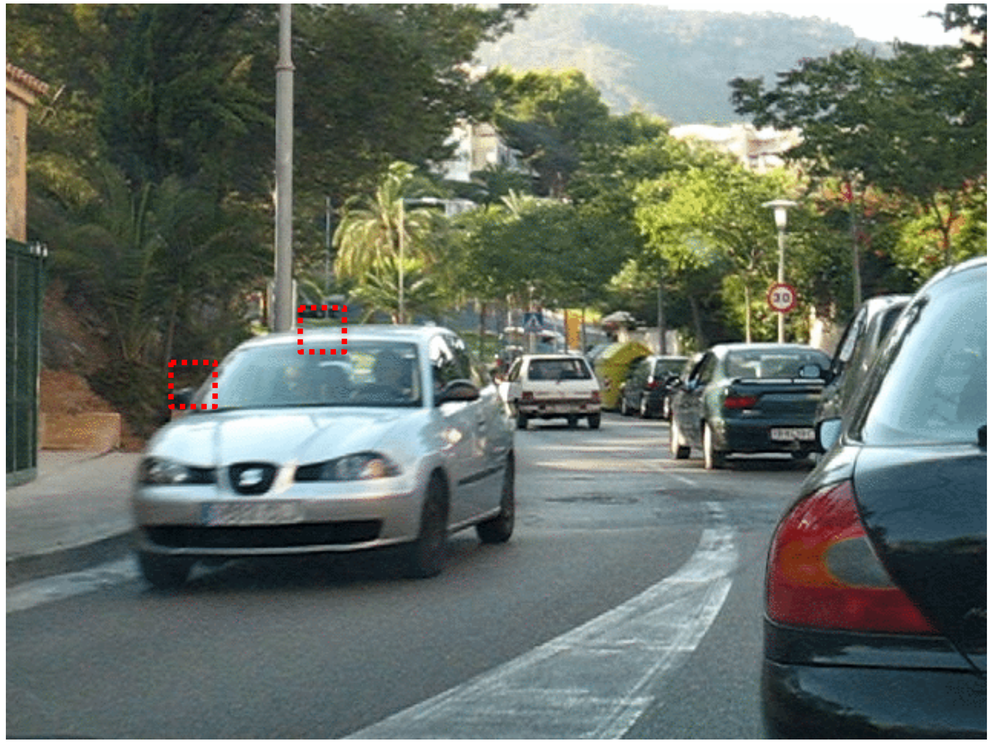


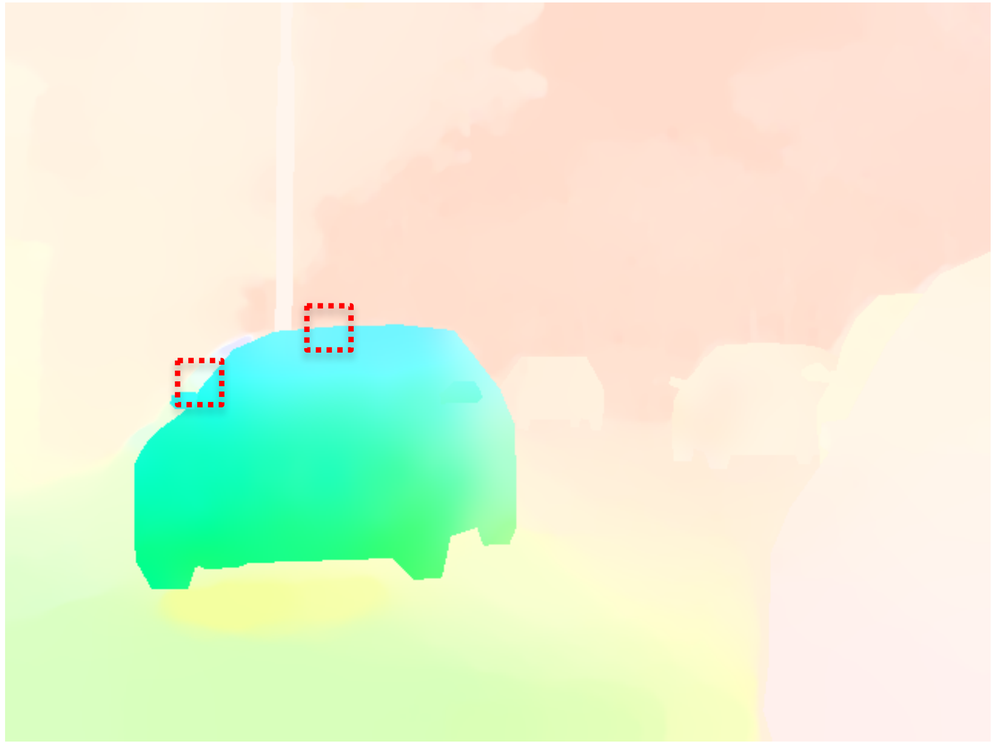


Also at frame boundary

Issue: Motion Boundaries

No matches can be found at motion boundaries…





Also at frame boundary

Issue: Ambiguous patch details

Flat regions of the image are underconstrained





Issue: Ambiguous patch details

Flat regions of the image are underconstrained





Secrets paper

- https://cs.brown.edu/people/dqsun/pubs/cvpr_2010_flow.pdf

Secrets of Optical Flow Estimation and Their Principles

Deqing Sun
Brown University

Stefan Roth
TU Darmstadt

Michael J. Black
Brown University

Improving Horn & Schunck

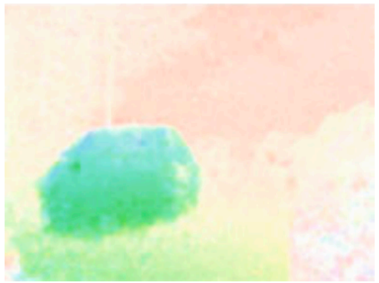
[Sun *et al.* CVPR'10, IJCV'14]

$$E(\mathbf{w}) = \sum_{\mathbf{p}} \left| I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w}_\mathbf{p}\right) \right|^2 + \lambda \sum_{\mathbf{q} \in N_\mathbf{p}} \left| \mathbf{w}_\mathbf{p} - \mathbf{w}_\mathbf{q} \right|^2$$





"Old"



Multi-Scale Pyramid



Ground truth

Improving Horn & Schunck

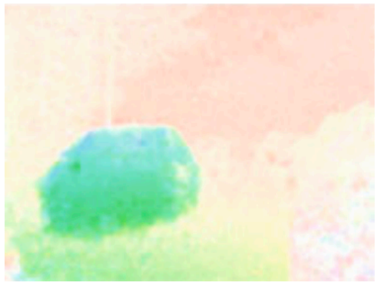
[Sun *et al.* CVPR'10, IJCV'14]

$$E(\mathbf{w}) = \sum_{\mathbf{p}} \left| I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w}_\mathbf{p}\right) \right|^2 + \lambda \sum_{\mathbf{q} \in N_\mathbf{p}} \left| \mathbf{w}_\mathbf{p} - \mathbf{w}_\mathbf{q} \right|^2$$

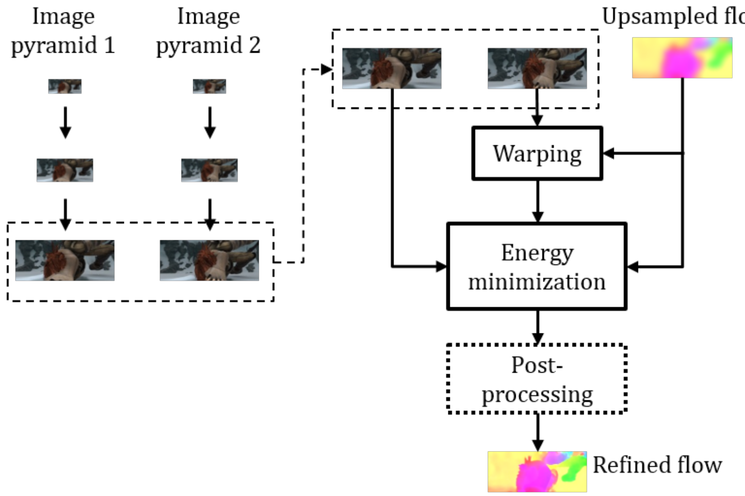




"Old"



Multi-Scale Pyramid





Ground truth

Improving Horn & Schunck

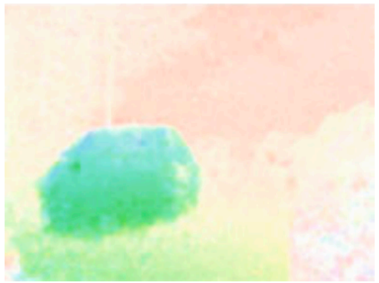
[Sun *et al.* CVPR'10, IJCV'14]

$$E(\mathbf{w}) = \sum_{\mathbf{p}} \left| I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w}_\mathbf{p}\right) \right|^2 + \lambda \sum_{\mathbf{q} \in N_\mathbf{p}} \left| \mathbf{w}_\mathbf{p} - \mathbf{w}_\mathbf{q} \right|^2$$





"Old"



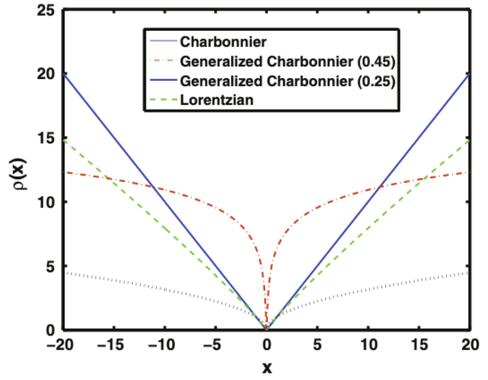
Multi-Scale Pyramid



Robust function



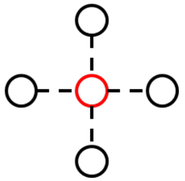
Ground truth



Improving Horn & Schunck

[Sun *et al.* CVPR'10, IJCV'14]

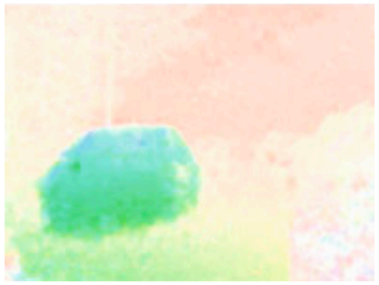
$$E(\mathbf{w}) = \sum_{\mathbf{p}} \left| I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w}_\mathbf{p}\right) \right|^2 + \lambda \sum_{\mathbf{q} \in N_\mathbf{p}} \left| \mathbf{w}_\mathbf{p} - \mathbf{w}_\mathbf{q} \right|^2$$







"Old"



Multi-Scale Pyramid



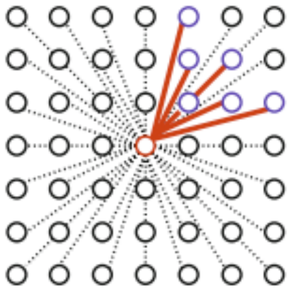
Robust function



Non-local term

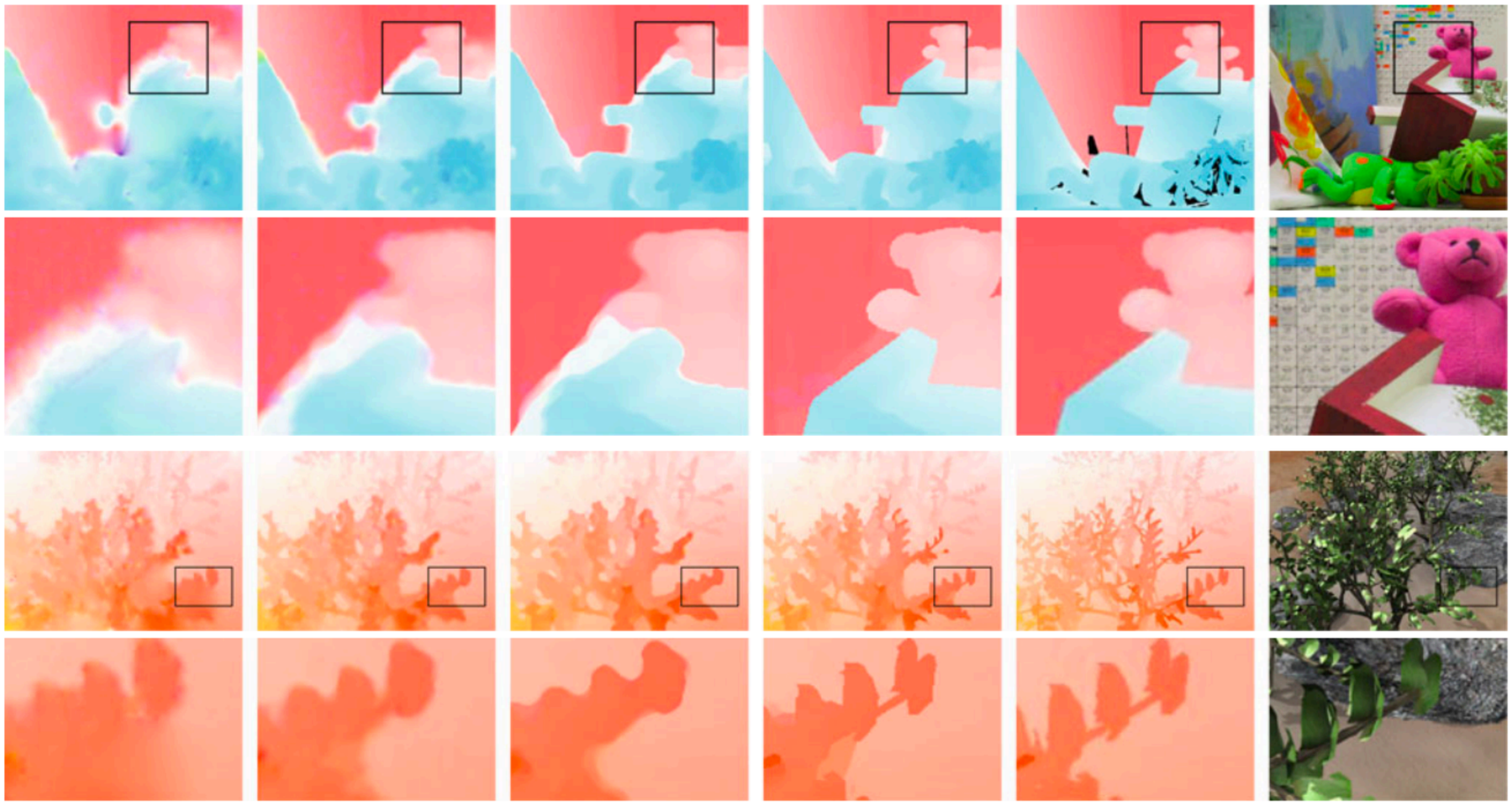


Ground truth



Improving Horn & Schunck

[Sun *et al.* CVPR'10, IJCV'14]



(a) "Old" **HS** [58] **(b)** "New" **HS** **(c)** **Classic++** **(d)** **Classic+NL** **(e)** Ground truth **(f)** First frame

Implementation Robust function Non-local smoothness term

Challenges for classical methods

Large motion
Motion blur
Occlusions
Lighting changes
Noise…

Small objects

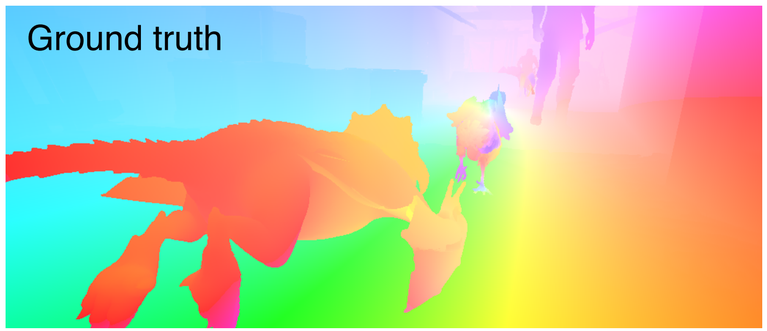
Hard to modify objective function and even harder to optimize it

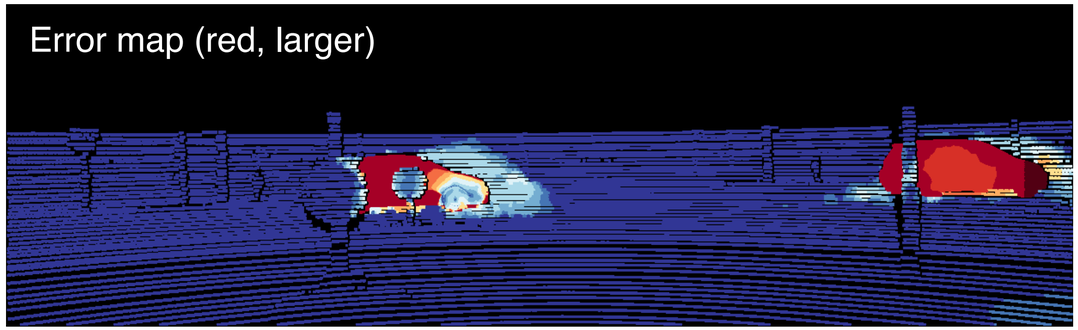












Challenges for classical methods

Large motion
Motion blur
Occlusions
Lighting changes
Noise…

Small objects

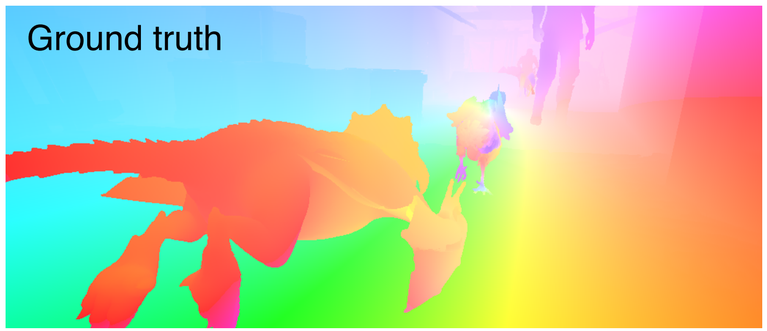
Hard to modify objective function and even harder to optimize it

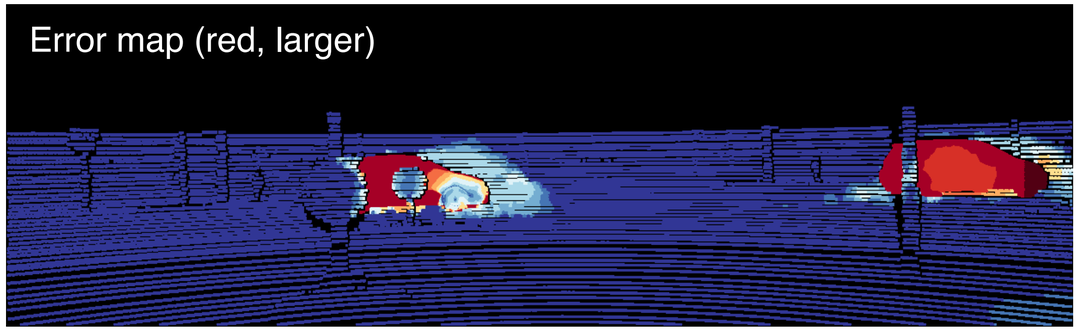












SIFT flow and Large Displacement Optical Flow (LDOF)

SIFT Flow: Dense Correspondence across Scenes and its Applications

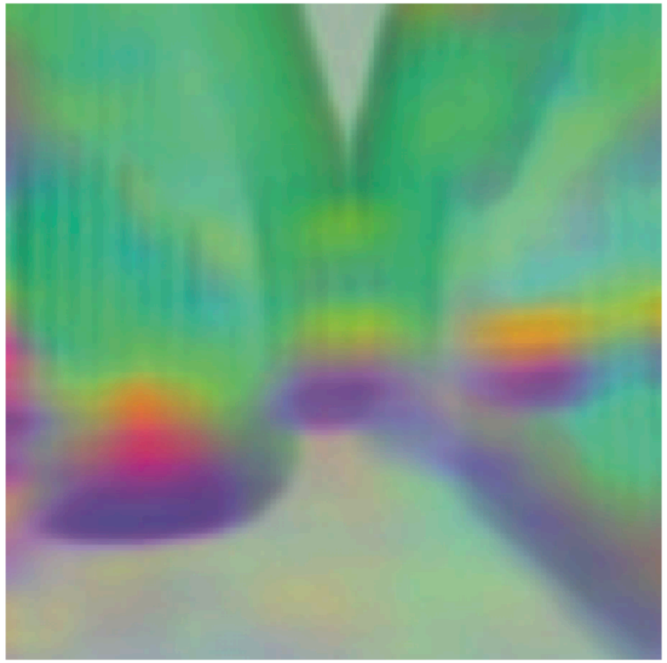
Ce Liu, *Member, IEEE,* Jenny Yuen, *Student Member, IEEE,* and Antonio Torralba, *Member, IEEE*

Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation

Thomas Brox, Jitendra Malik, *Fellow, IEEE*



(c) A RGB image



(d) Visualization of the SIFT image of (c)

Recap

- Flow is a dense 2D vector field between two frames
- Energy minimization (+regularization+iterative solver)
  vs. 1-step prediction (maybe with coarse to fine)
- Edge aware regularization
- RGB vs features

Supervised optical flow

[Dosovitskiy *et al.* ICCV'15]

FlowNet: Learning Optical Flow with Convolutional Networks

Alexey Dosovitskiy[*], Philipp Fischer[†,*], Eddy Ilg[*], Philip Häusser, Caner Hazırbaş, Vladimir Golkov[†]
University of Freiburg — Technical University of Munich
{fischer,dosovits,ilg}@cs.uni-freiburg.de, {haeusser,hazirbas,golkov}@cs.tum.edu

Patrick van der Smagt
Technical University of Munich
smagt@brml.org

Daniel Cremers
Technical University of Munich
cremers@tum.de

Thomas Brox
University of Freiburg
brox@cs.uni-freiburg.de

Abstract

Convolutional neural networks (CNNs) have recently been very successful in a variety of computer vision tasks, especially on those linked to recognition. Optical flow estimation has not been among the tasks CNNs succeeded at. In this paper we construct CNNs which are capable of solving the optical flow estimation problem as a supervised learning task. We propose and compare two architectures: a generic architecture and another one including a layer that correlates feature vectors at different image locations. Since existing ground truth data sets are not sufficiently large to train a CNN, we generate a large synthetic Flying Chairs dataset. We show that networks trained on this unrealistic data still generalize very well to existing datasets such as Sintel and KITTI, achieving competitive accuracy at frame rates of 5 to 10 fps.

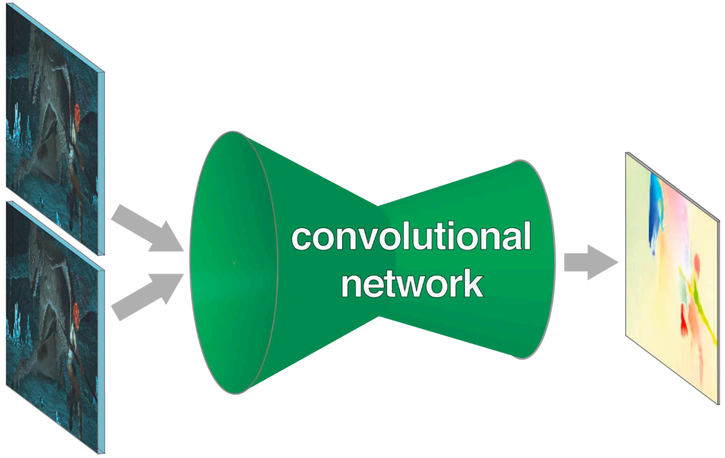
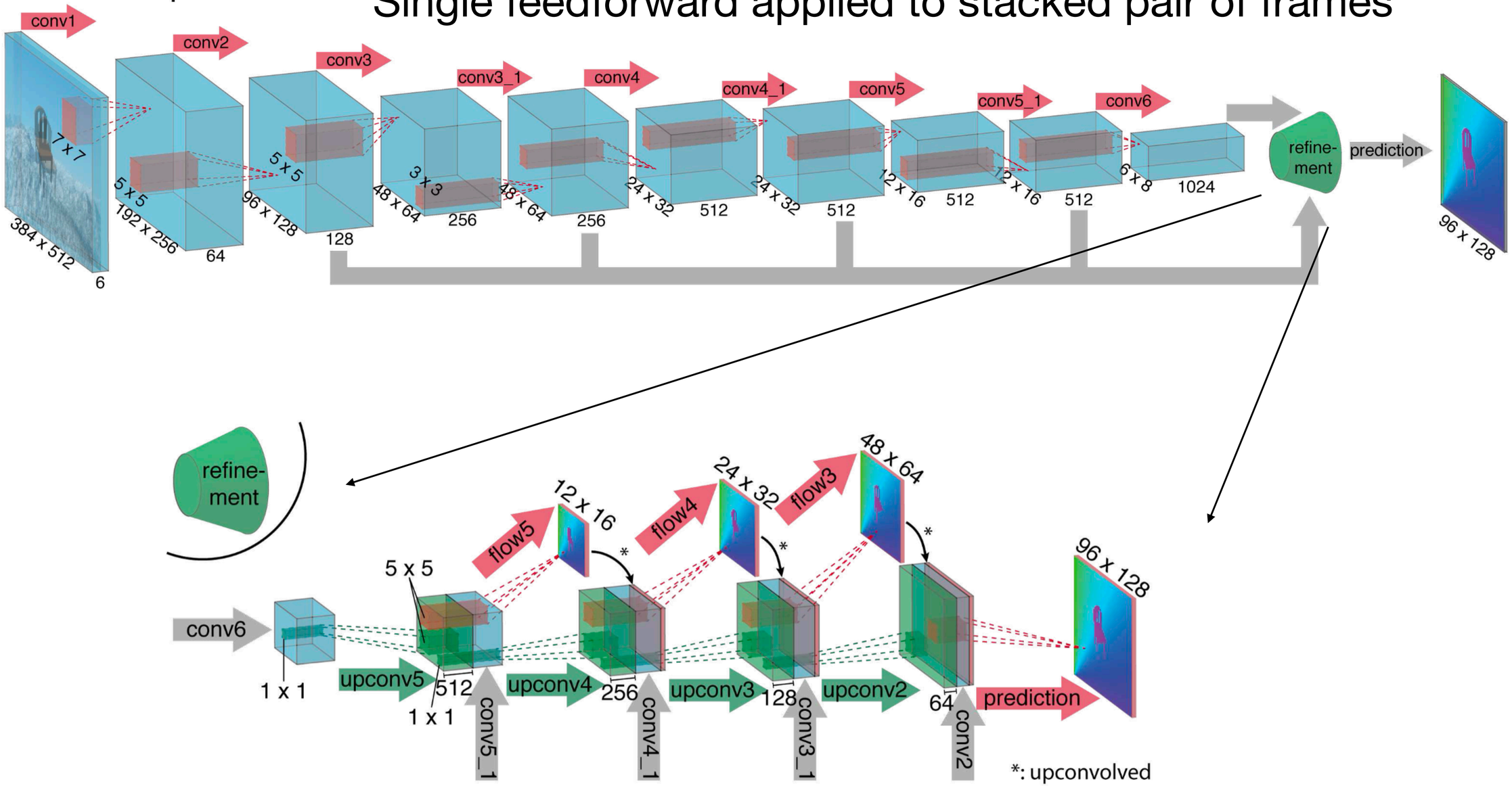


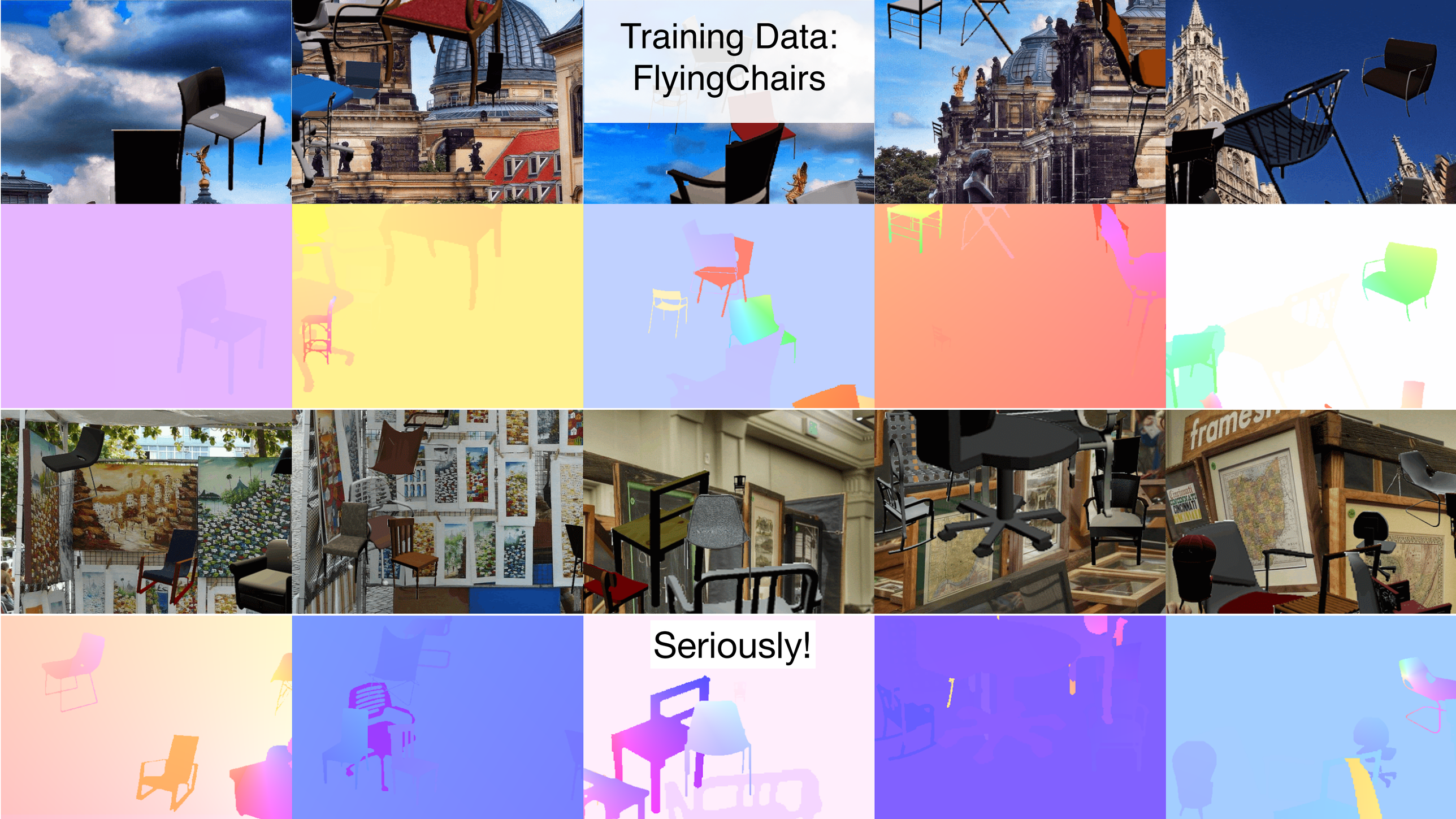
Figure 1. We present neural networks which learn to estimate optical flow, being trained end-to-end. The information is first spatially compressed in a contractive part of the network and then refined in an expanding part.

FlowNetSimple

Single feedforward applied to stacked pair of frames



Refinement of features and flow field


Training Data:
FlyingChairs
Seriously!

Two widely-used benchmarks for optical flow

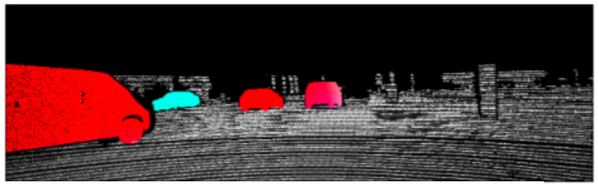
Sintel (Blender movie)



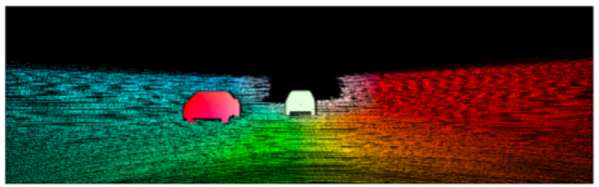


KITTI (driving)









One big issue with optical flow : limited ground truth data

Two widely-used benchmarks for optical flow

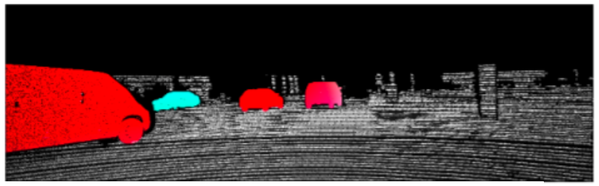
Sintel (Blender movie)



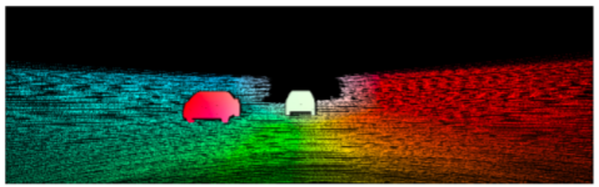


KITTI (driving)

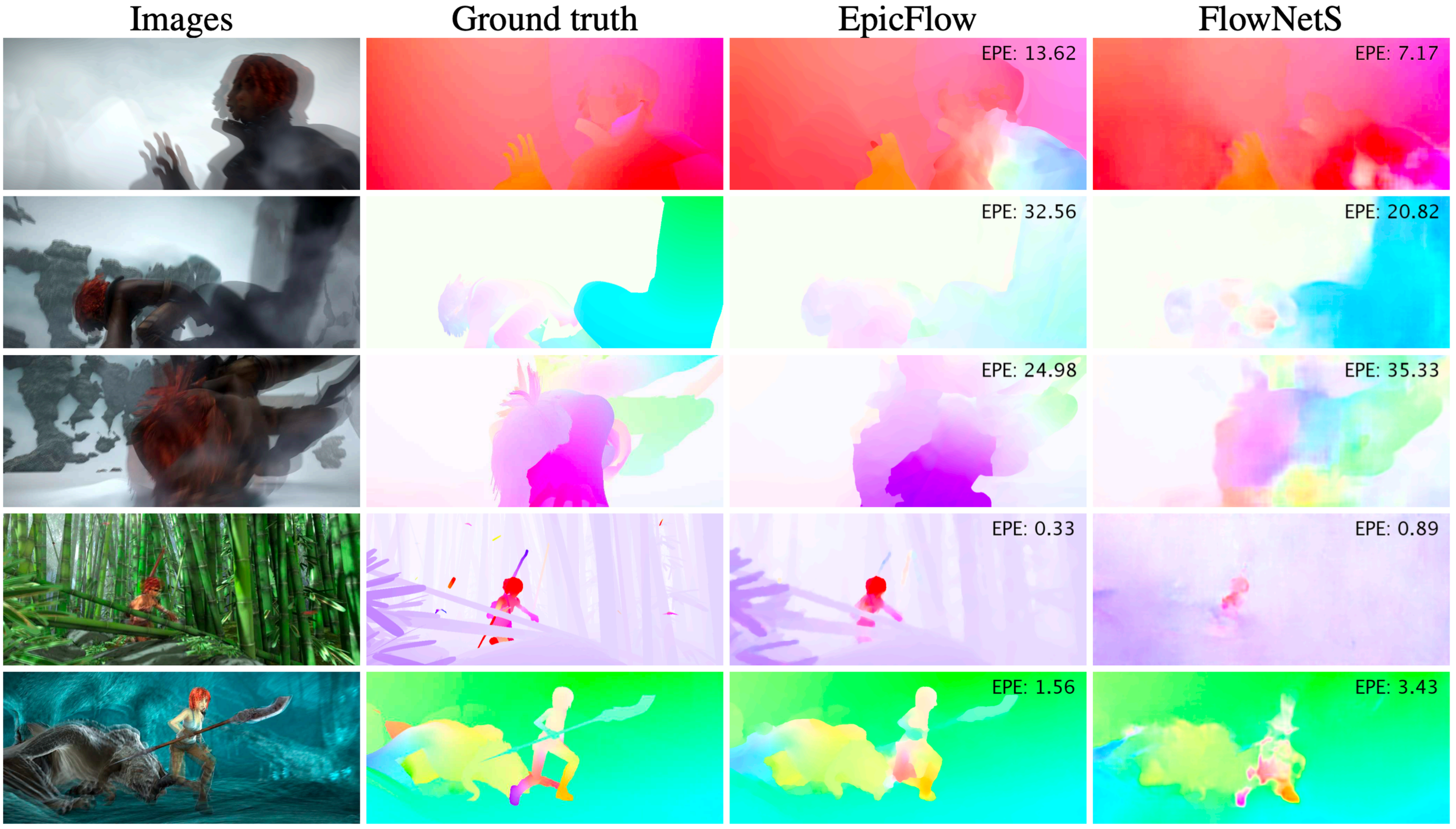






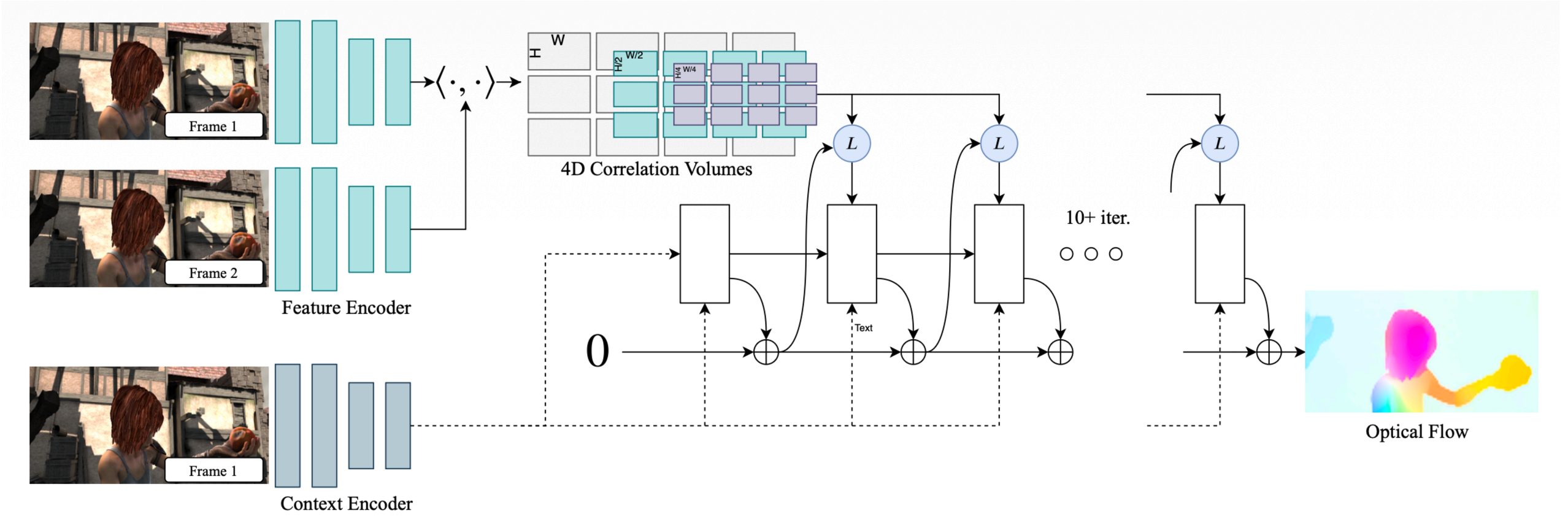


One big issue with optical flow : limited ground truth data


Images
Ground truth
EpicFlow
FlowNetS
EPE: 13.62
EPE: 7.17
EPE: 32.56
EPE: 20.82
EPE: 24.98
EPE: 35.33
EPE: 0.33
EPE: 0.89
EPE: 1.56
EPE: 3.43

[Teed and Deng ECCV 2020 Best paper]

RAFT: Recurrent All-pairs Field Transforms



RAFT idea : Neuralize old approach

Traditional approach:

- Data term: similarity of RGB
- Iterative minimization :
  - Current estimate of flow
    (2D vector field)
  - At each step, update flow estimate
    based on data similarity and
    neighbors (regularization)

RAFT idea : Neuralize old approach

- Data term: similarity of RGB

Replace RGB by learned features

+ precompute all similarities (2Dx2D -> 4D)

- Iterative minimization :

Iterations with serialized recurrent network (shared weights)

  - Current estimate of flow (2D vector field)

Unchanged!

  - At each step, update flow estimate based on data similarity and neighbors (regularization)

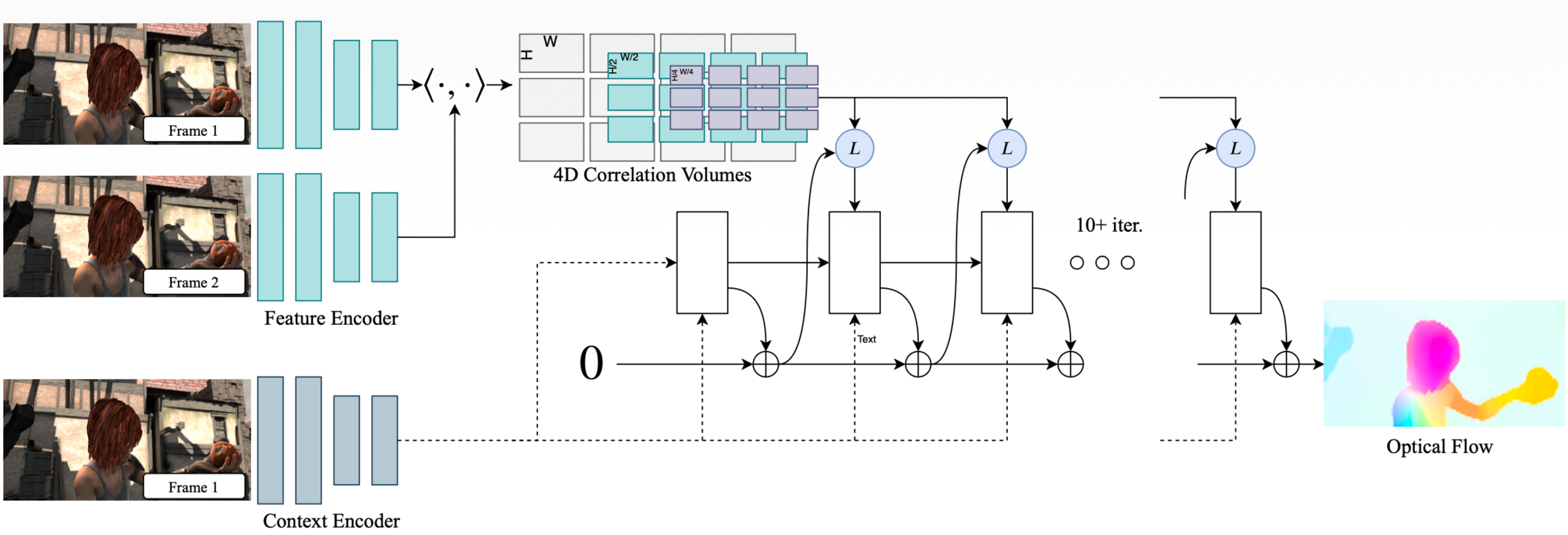
Learn update

Look up precomputed correlations

Provide image context so that update can regularize based on edges and the such

[Teed and Deng ECCV 2020 Best paper]

RAFT: Recurrent All-pairs Field Transforms



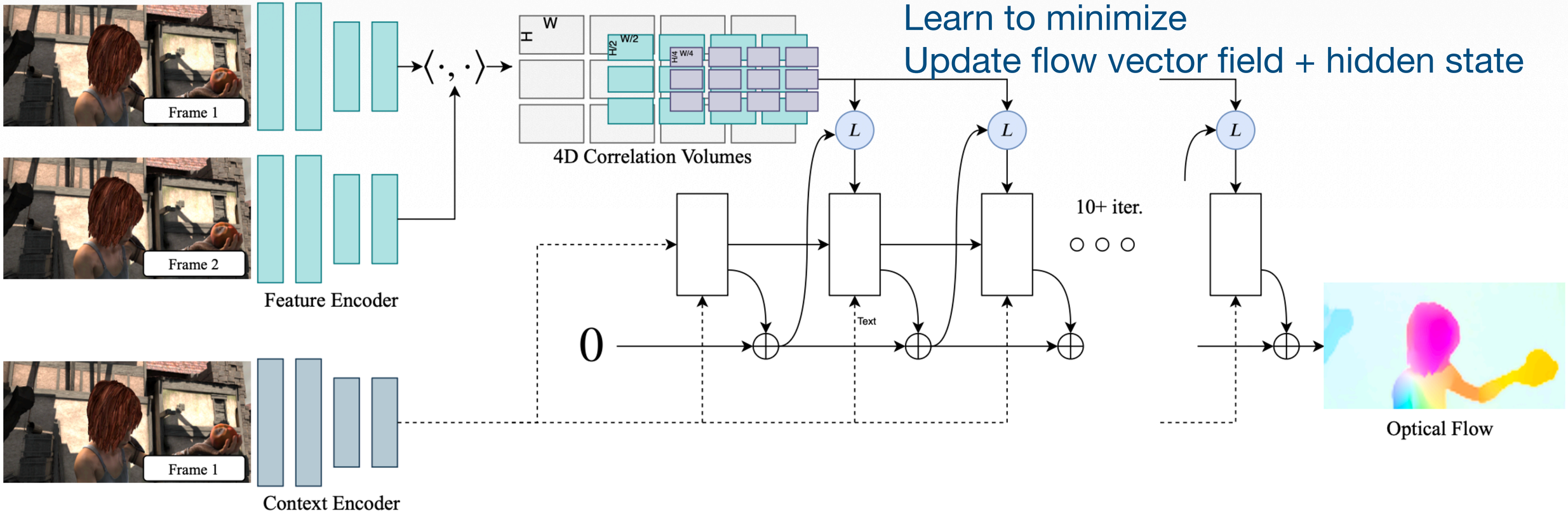
[Teed and Deng ECCV 2020 Best paper]

RAFT: Recurrent All-pairs Field Transforms

Learn features

Correlation 4D Volume / pyramids (~precompute similarity for performance)

Update operator:
Learn to minimize
Update flow vector field + hidden state

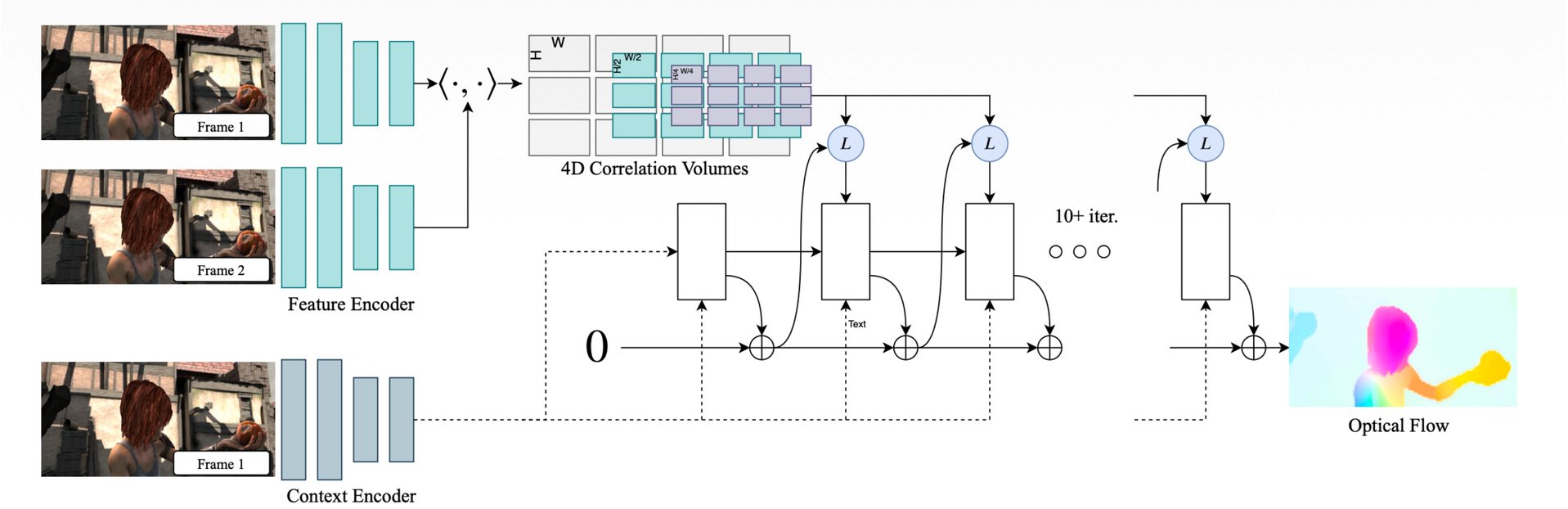


Context (~ for smart regularization)

All differentiable and learned end-to-end

[Teed and Deng ECCV 2020 Best paper]

RAFT: Recurrent All-pairs Field Transforms

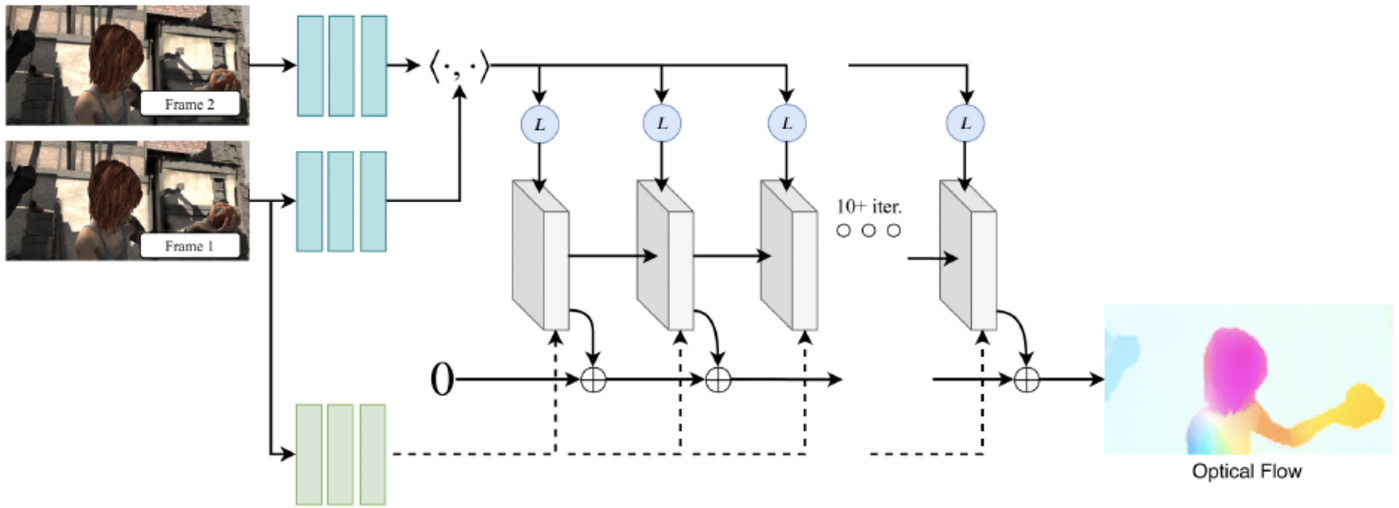


In the authors' words:

Fig. 1: RAFT consists of 3 main components: (1) A feature encoder that extracts per-pixel features from both input images, along with a context encoder that extracts features from only I_1. (2) A correlation layer which constructs a 4D $W \times H \times W \times H$ correlation volume by taking the inner product of all pairs of feature vectors. The last 2-dimensions of the 4D volume are pooled at multiple scales to construct a set of multi-scale volumes. (3) An *update operator* which recurrently updates optical flow by using the current estimate to look up values from the set of correlation volumes.

RAFT: Recurrent All-pairs Field Transforms

[Teed and Deng ECCV 2020 Best paper]



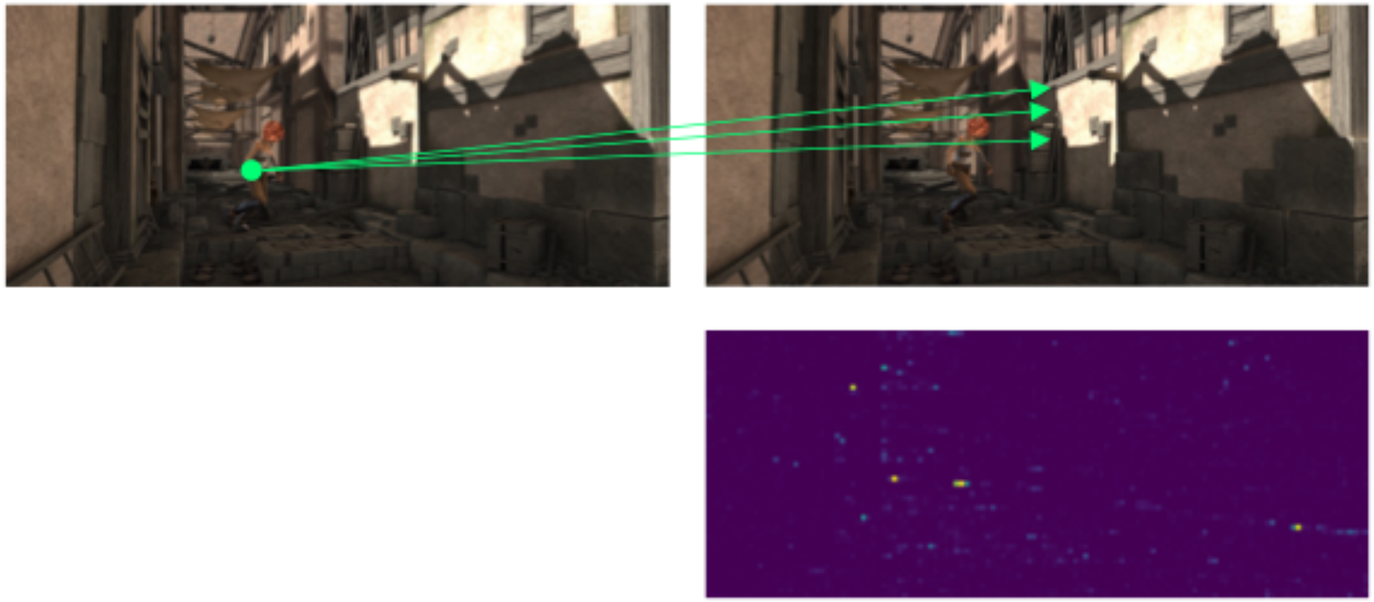
Key ideas

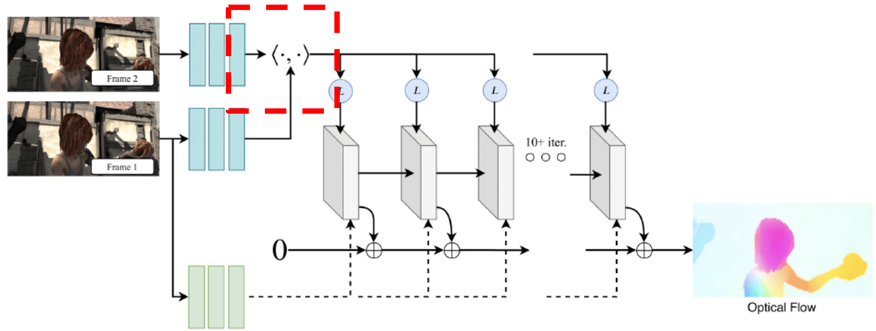
- Learned features
- All-pairs cost volume (4D) across scales
- Recurrent module for iterative refinement of flow
- Better training schemes

Slide credit: Teed and Deng

All-pairs visual similarity (cost volume)

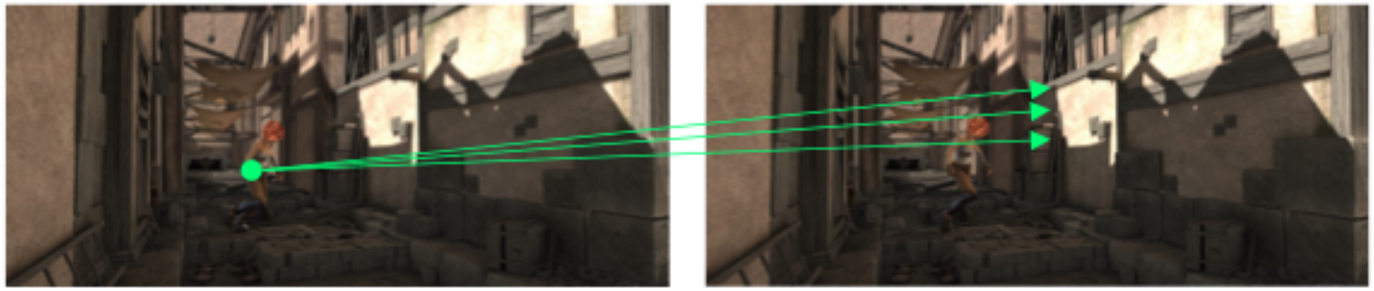
Inner product/correlation between features





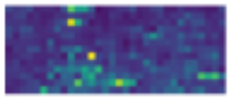
Cost volume pyramid

Spatial pooling

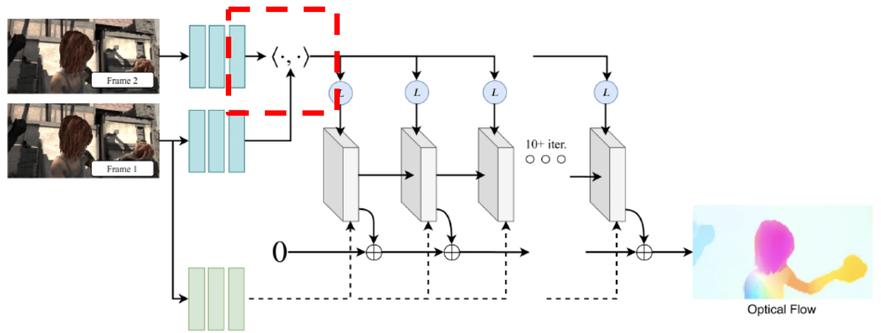








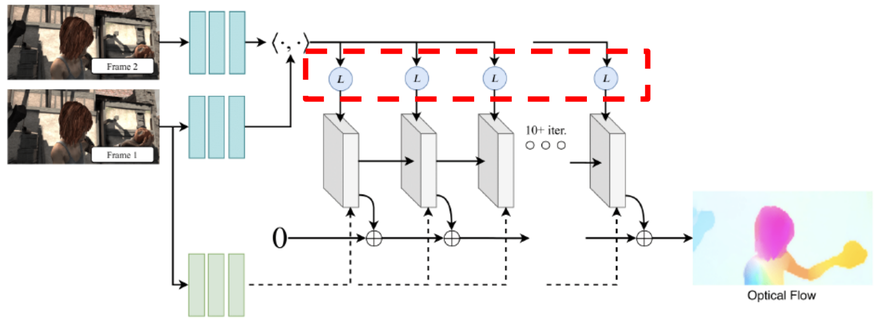


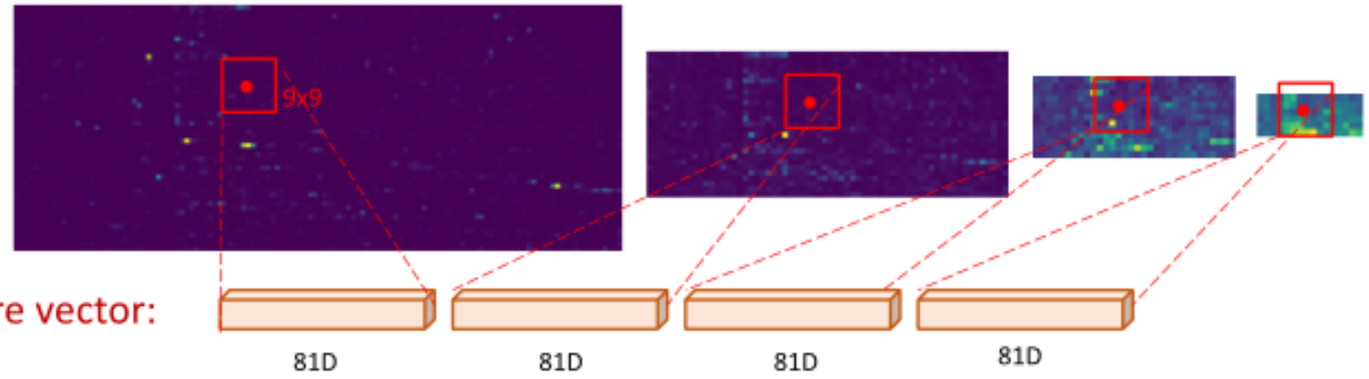


Look up cost volume

Retrieve using current motion estimates



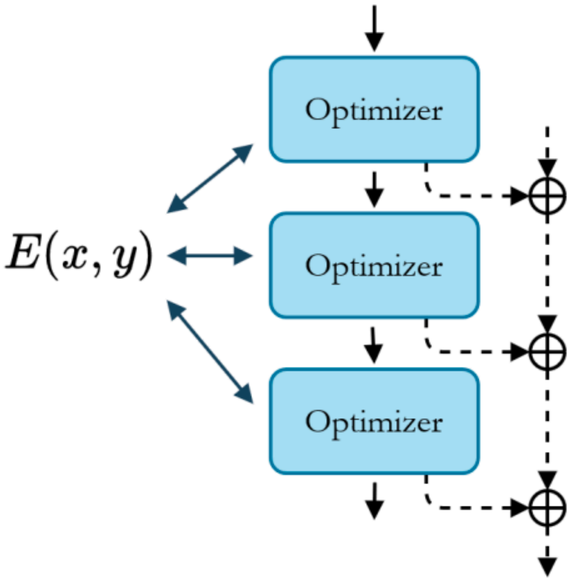


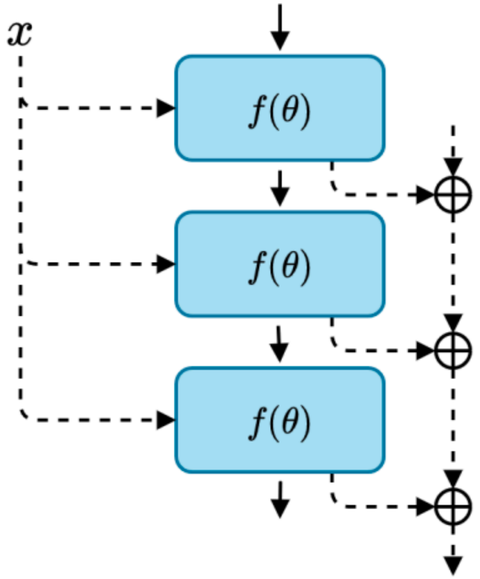


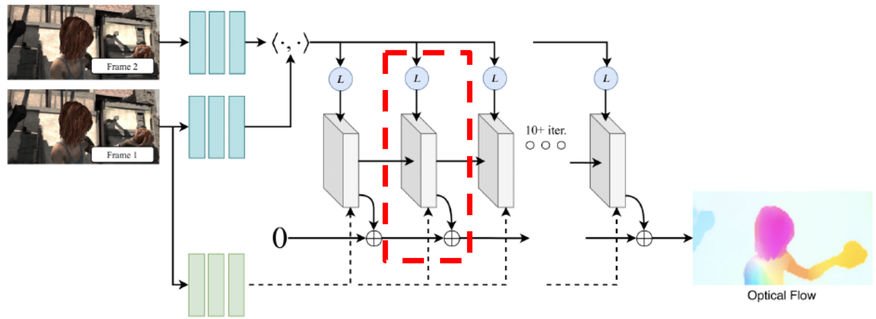
cues on how good the current flow is and where are better similarities

Recurrent update

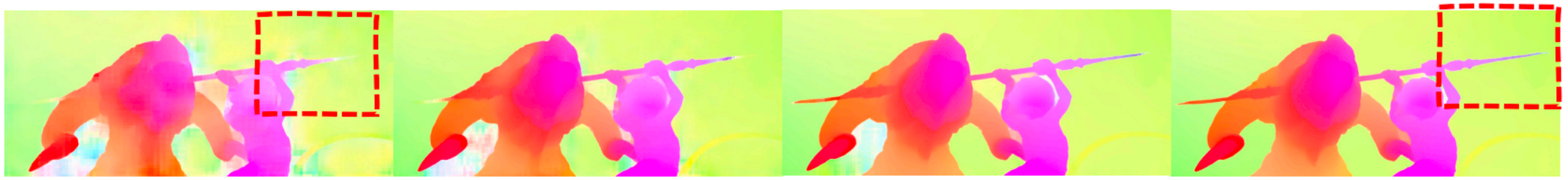
Like classical optimization algorithms







Recurrent update



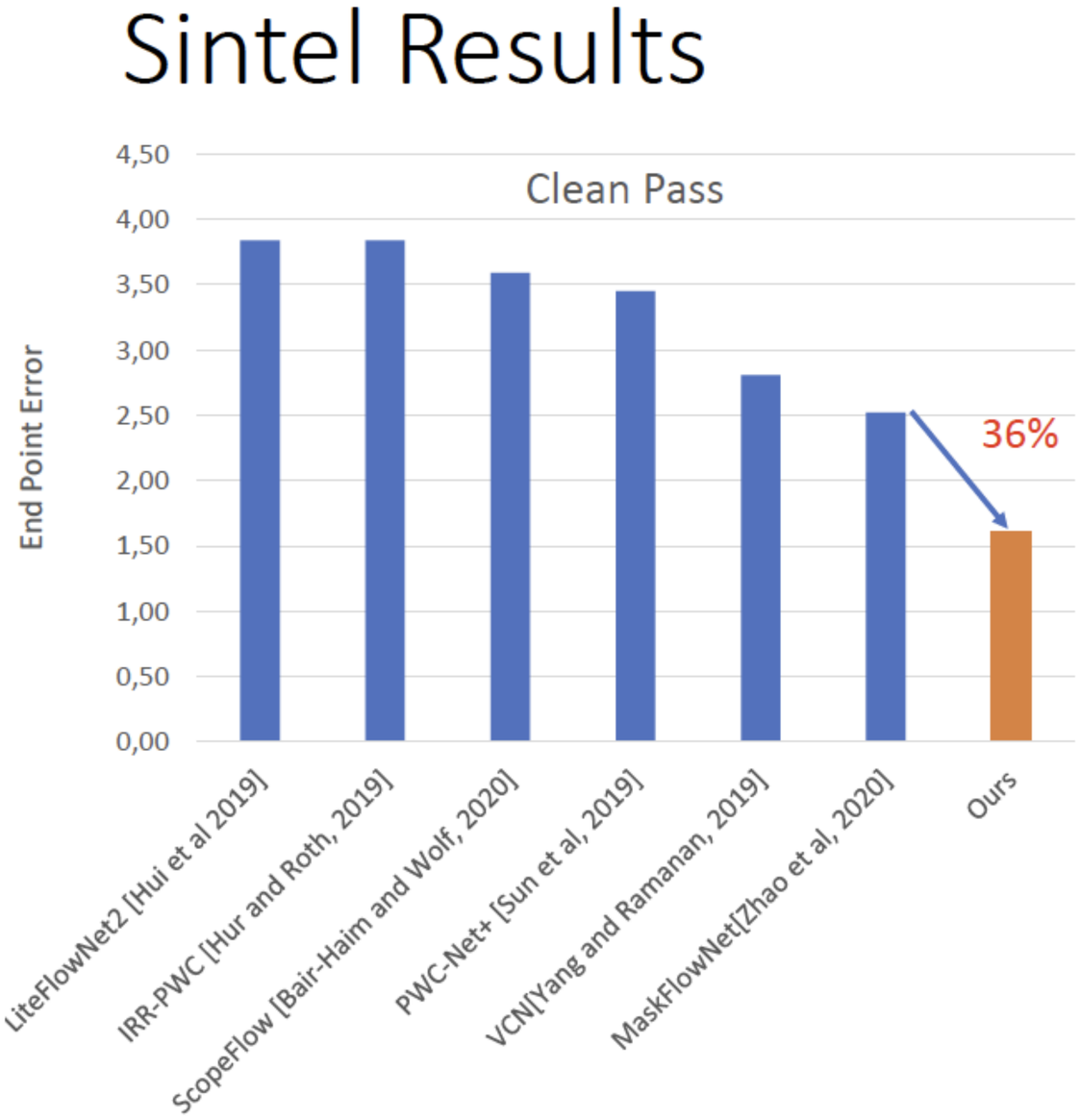
1 Iteration | 2 Iterations | 5 Iterations | 32 Iterations

From the paper

- “The RAFT architecture
  is **motivated by traditional optimization-based approaches**.
- The feature encoder extracts per-pixel features.
- The correlation layer computes visual similarity between pixels.
- The update operator mimics the steps of an iterative optimization algorithm. But unlike traditional approaches, features and motion priors are not handcrafted but learned—learned by the feature encoder and the update operator respectively.”

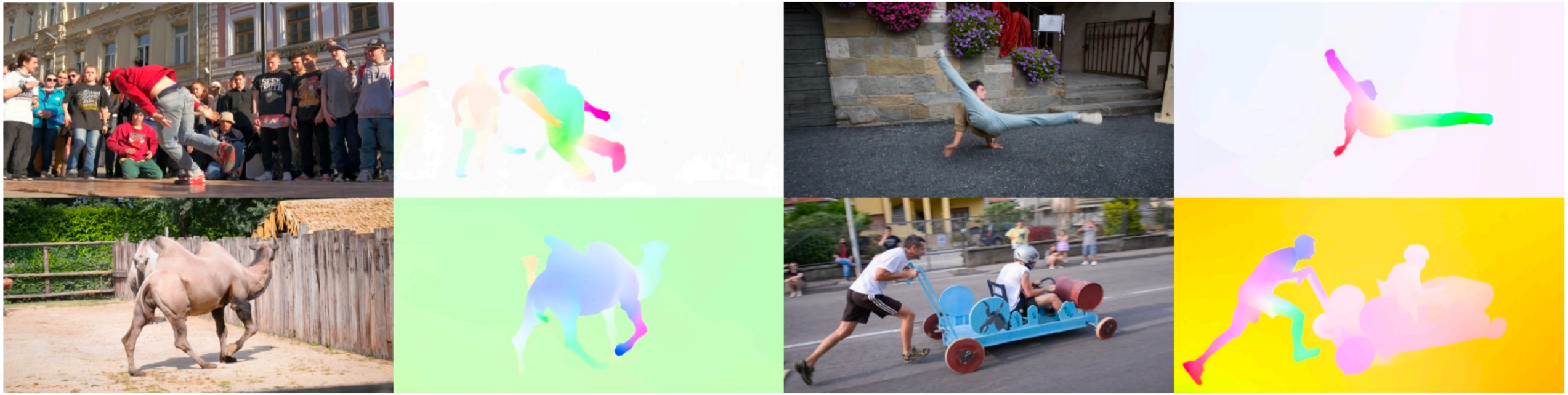
Significant improvement over prior art

[Teed and Deng ECCV 2020 Best paper]



Slide credit: Teed and Deng

Visual results on Davis (real-world)



Discussion

- Specific architecture hand-crafted for the problem
- Still largely feed-forward (well, unrolled recurrence)
- Scaffolding with traditional methods and inductive bias
  - Historically it all happened progressively in the span of multiple papers : traditional pieces of the approach got replaced with learnable ones
- Non-local matching is hard and requires specialized architectures

Is the hand-crafted structure (cost-volume, unrolled optimization on cost volume) really necessary?

The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth. With self-supervised pre-training, the combined use of synthetic and real data for supervised training, and technical innovations (infilling and step-unrolled denoising diffusion training) to handle noisy-incomplete training data, and a simple form of coarse-to-fine refinement, one can train state-of-the-art diffusion models for depth and optical flow estimation. Extensive experiments focus on quantitative performance against benchmarks, ablations, and the model's ability to capture uncertainty and multimodality, and impute missing values. Our model, DDVM (Denoising Diffusion Vision Model), obtains a state-of-the-art relative depth error of 0.074 on the indoor NYU benchmark and an Fl-all outlier rate of 3.26% on the KITTI optical flow benchmark, about 25% better than the best published method. For an overview see diffusion-vision.github.io

1 Introduction

Diffusion models have emerged as powerful generative models for high fidelity image synthesis, capturing rich knowledge about the visual world [21, 48, 55, 62]. However, at first glance, it is unclear whether these models can be as effective on many classical computer vision tasks. For example, consider two dense vision estimation tasks, namely, optical flow, which estimates frame-to-frame correspondences, and monocular depth perception, which makes depth predictions based on a single image. Both tasks are usually treated as regression problems and addressed with specialized architectures and task-specific loss functions, *e.g.*, cost volumes, feature warps, or suitable losses for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled

*DF is also affiliated with the University of Toronto and the Vector Institute.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Is the hand-crafted structure (cost-volume, unrolled optimization on cost volume) really necessary?

The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth. With self-supervised pre-training, the combined use of synthetic and real data for supervised training, and technical innovations (infilling and step-unrolled denoising diffusion training) to handle noisy-incomplete training data, and a simple form of coarse-to-fine refinement, one can train state-of-the-art diffusion models for depth and optical flow estimation. Extensive experiments focus on quantitative performance against benchmarks, ablations, and the model's ability to capture uncertainty and multimodality, and impute missing values. Our model, DDVM (Denoising Diffusion Vision Model), obtains a state-of-the-art relative depth error of 0.074 on the indoor NYU benchmark and an Fl-all outlier rate of 3.26% on the KITTI optical flow benchmark, about 25% better than the best published method. For an overview see diffusion-vision.github.io

1 Introduction

Diffusion models have emerged as powerful generative models for high fidelity image synthesis, capturing rich knowledge about the visual world [21, 48, 55, 62]. However, at first glance, it is unclear whether these models can be as effective on many classical computer vision tasks. For example, consider two dense vision estimation tasks, namely, optical flow, which estimates frame-to-frame correspondences, and monocular depth perception, which makes depth predictions based on a single image. Both tasks are usually treated as regression problems and addressed with specialized architectures and task-specific loss functions, *e.g.*, cost volumes, feature warps, or suitable losses for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled

*DF is also affiliated with the University of Toronto and the Vector Institute.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

- Use generative approach (diffusion)
  - Conditioned on two frames, denoise flow
- Still supervised (ground truth flow)
  - Existing data not good enough
  - Merge four large synthetic flow dataset for pre-training: FlyingThings3D, AutoFlow, Kubric, TartanAir
- Also applied to depth prediction

The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar,
Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth. With self-supervised pre-training, the combined use of synthetic and real data for supervised training, and technical innovations (infilling and step-unrolled denoising diffusion training) to handle noisy-incomplete training data, and a simple form of coarse-to-fine refinement, one can train state-of-the-art diffusion models for depth and optical flow estimation. Extensive experiments focus on quantitative performance against benchmarks, ablations, and the model's ability to capture uncertainty and multimodality, and impute missing values. Our model, DDVM (Denoising Diffusion Vision Model), obtains a state-of-the-art relative depth error of 0.074 on the indoor NYU benchmark and an Fl-all outlier rate of 3.26% on the KITTI optical flow benchmark, about 25% better than the best published method. For an overview see diffusion-vision.github.io

1 Introduction

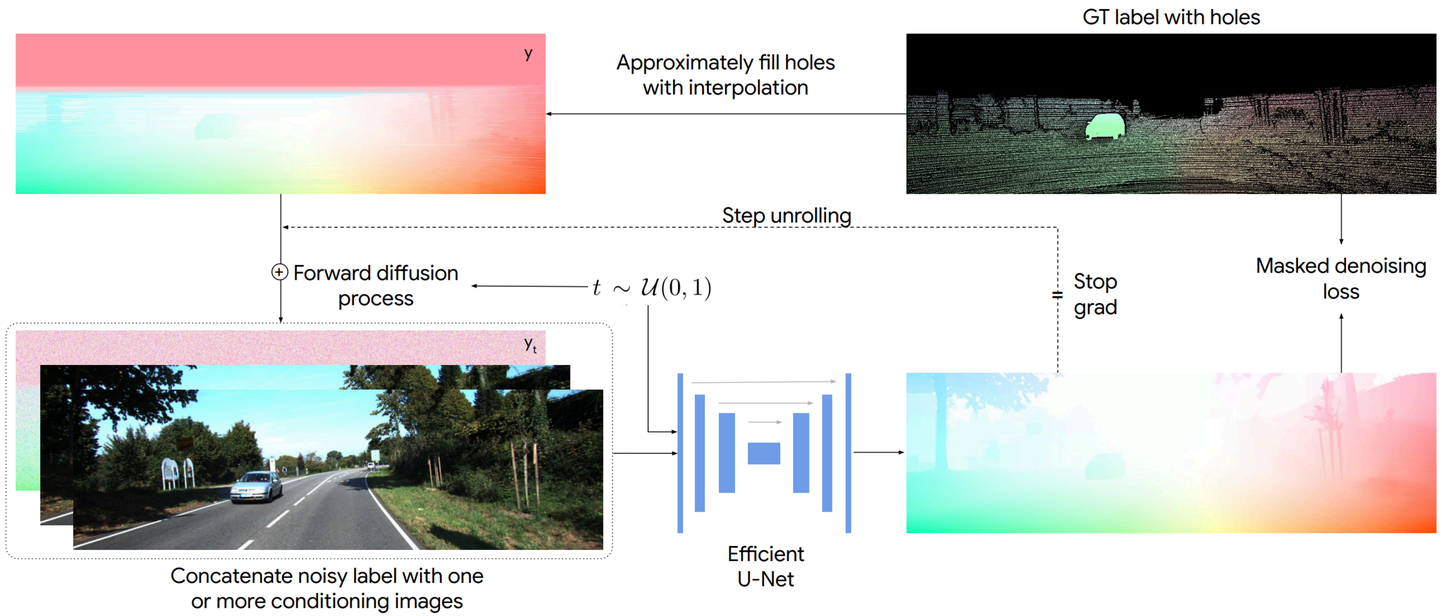
Diffusion models have emerged as powerful generative models for high fidelity image synthesis, capturing rich knowledge about the visual world [21, 48, 55, 62]. However, at first glance, it is unclear whether these models can be as effective on many classical computer vision tasks. For example, consider two dense vision estimation tasks, namely, optical flow, which estimates frame-to-frame correspondences, and monocular depth perception, which makes depth predictions based on a single image. Both tasks are usually treated as regression problems and addressed with specialized architectures and task-specific loss functions, *e.g.*, cost volumes, feature warps, or suitable losses for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled

*DF is also affiliated with the University of Toronto and the Vector Institute.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).



First frame



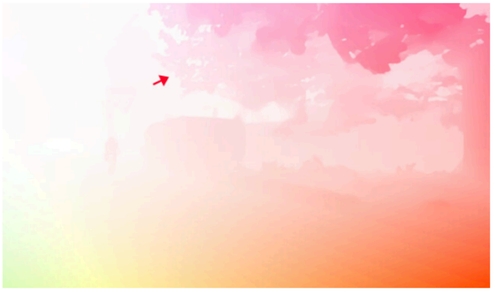
AF+FT



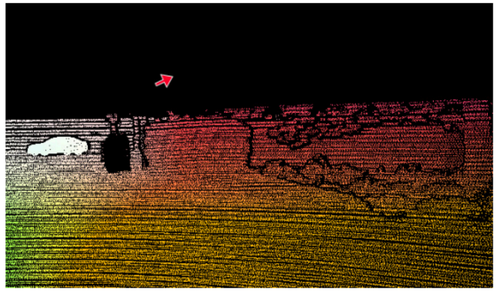
AF+FT+KU



AF+FT+KU+TA



GroundTruth



First frame



AF



AF+FT



AF+FT+KU



AF+FT+KU+TA



GroundTruth



The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth. With self-supervised pre-training, the combined use of synthetic and real data for supervised training, and technical innovations (infilling and step-unrolled denoising diffusion training) to handle noisy-incomplete training data, and a simple form of coarse-to-fine refinement, one can train state-of-the-art diffusion models for depth and optical flow estimation. Extensive experiments focus on quantitative performance against benchmarks, ablations, and the model's ability to capture uncertainty and multimodality, and impute missing values. Our model, DDVM (Denoising Diffusion Vision Model), obtains a state-of-the-art relative depth error of 0.074 on the indoor NYU benchmark and an Fl-all outlier rate of 3.26% on the KITTI optical flow benchmark, about 25% better than the best published method. For an overview see diffusion-vision.github.io

1 Introduction

Diffusion models have emerged as powerful generative models for high fidelity image synthesis, capturing rich knowledge about the visual world [21, 48, 55, 62]. However, at first glance, it is unclear whether these models can be as effective on many classical computer vision tasks. For example, consider two dense vision estimation tasks, namely, optical flow, which estimates frame-to-frame correspondences, and monocular depth perception, which makes depth predictions based on a single image. Both tasks are usually treated as regression problems and addressed with specialized architectures and task-specific loss functions, *e.g.*, cost volumes, feature warps, or suitable losses for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled

*DF is also affiliated with the University of Toronto and the Vector Institute.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Key contributions:

- Merge four large synthetic flow dataset: FlyingThings3D, AutoFlow, Kubric, TartanAir
- Training with real data is tricky: Has holes! Diffusion model would generate optical flow maps with holes as well…
- Coarse-to-fine training strategy

The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar,
Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth. With self-supervised pre-training, the combined use of synthetic and real data for supervised training, and technical innovations (infilling and step-unrolled denoising diffusion training) to handle noisy-incomplete training data, and a simple form of coarse-to-fine refinement, one can train state-of-the-art diffusion models for depth and optical flow estimation. Extensive experiments focus on quantitative performance against benchmarks, ablations, and the model's ability to capture uncertainty and multimodality, and impute missing values. Our model, DDVM (Denoising Diffusion Vision Model), obtains a state-of-the-art relative depth error of 0.074 on the indoor NYU benchmark and an Fl-all outlier rate of 3.26% on the KITTI optical flow benchmark, about 25% better than the best published method. For an overview see diffusion-vision.github.io

1 Introduction

Diffusion models have emerged as powerful generative models for high fidelity image synthesis, capturing rich knowledge about the visual world [21, 48, 55, 62]. However, at first glance, it is unclear whether these models can be as effective on many classical computer vision tasks. For example, consider two dense vision estimation tasks, namely, optical flow, which estimates frame-to-frame correspondences, and monocular depth perception, which makes depth predictions based on a single image. Both tasks are usually treated as regression problems and addressed with specialized architectures and task-specific loss functions, *e.g.*, cost volumes, feature warps, or suitable losses for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled



*DF is also affiliated with the University of Toronto and the Vector Institute.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

The Surprising Effectiveness of Diffusion Models for Optical Flow and Monocular Depth Estimation

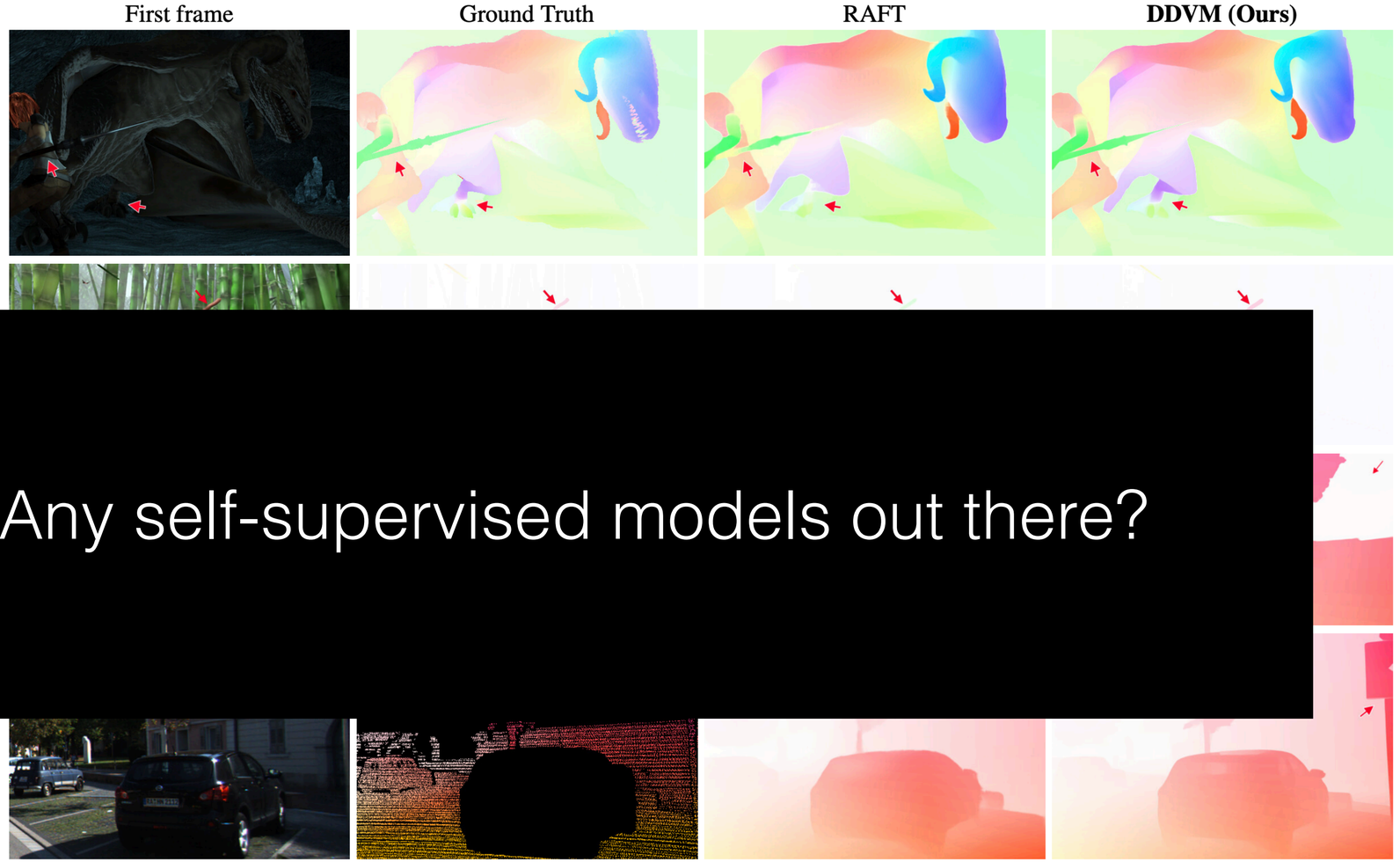
Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, David J. Fleet*

{srbs,irwinherrmann,junhwahur,abhiskar,deqingsun,davidfleet}@google.com

Google DeepMind and Google Research

Abstract

Denoising diffusion probabilistic models have transformed image generation with their impressive fidelity and diversity. We show that they also excel in estimating optical flow and monocular depth, surprisingly, without task-specific architectures and loss functions that are predominant for these tasks. Compared to the point estimates of conventional regression-based methods, diffusion models also enable Monte Carlo inference, e.g., capturing uncertainty and ambiguity in flow and depth.



Fully supervised… Any self-supervised models out there?

for depth. Without these specialized components or the regression framework, general generative techniques may be ill-equipped and vulnerable to both generalization and performance issues.

In this paper, we show that these concerns, while valid, can be addressed and that, surprisingly, a generic, conventional diffusion model for image to image translation works impressively well on both tasks, often outperforming the state of the art. In addition, diffusion models provide valuable benefits over networks trained with regression; in particular, diffusion allows for approximate inference with multi-modal distributions, capturing uncertainty and ambiguity (*e.g.* see Figure 1).

One key barrier to training useful diffusion models for monocular depth and optical flow inference concerns the amount and quality of available training data. Given the limited availability of labelled

*DF is also affiliated with the University of Toronto and the Vector Institute.

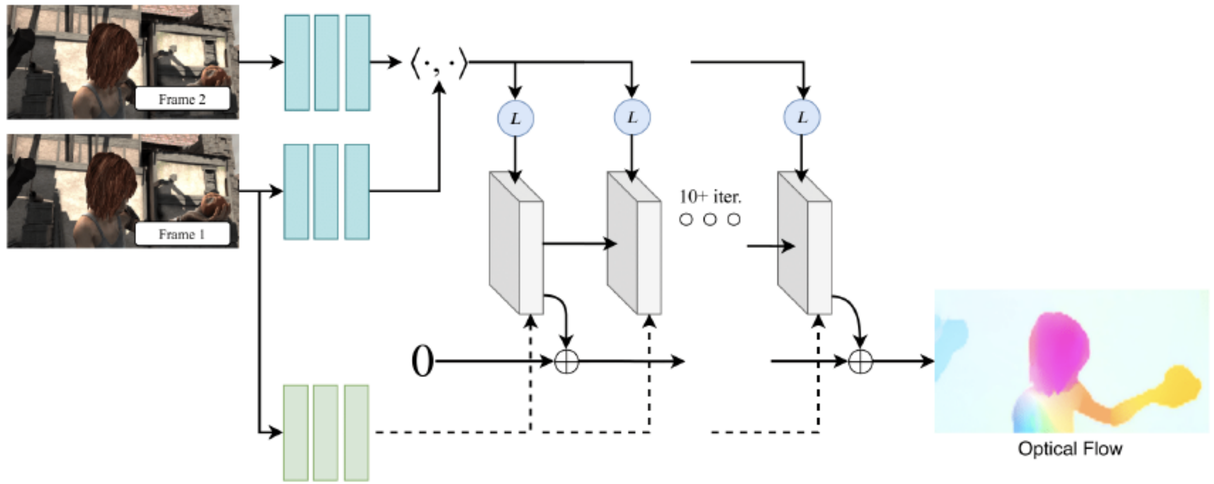
37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Self-Supervised Optical Flow

[Stone, Maurer et al. 2021, Jonschkowski et al. 2020, Meister et al. 2018, …]
(see SMURF for good discussion!)

Self-Supervised Optical Flow

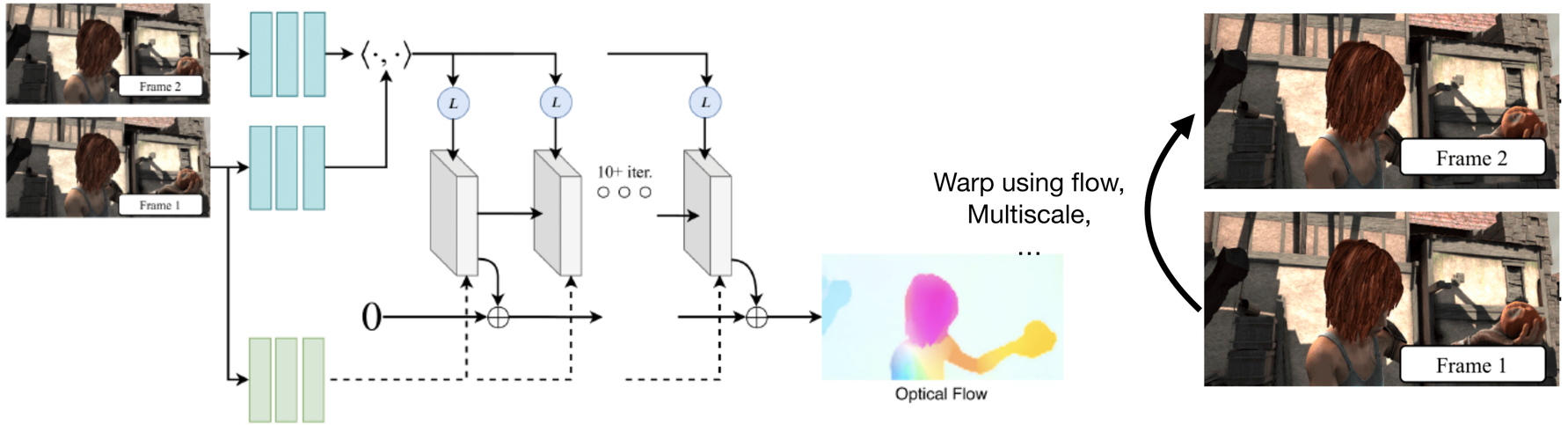
[Stone, Maurer et al. 2021, Jonschkowski et al. 2020, Meister et al. 2018, …]
(see SMURF for good discussion!)



Estimate flow with Neural Network…

Self-Supervised Optical Flow

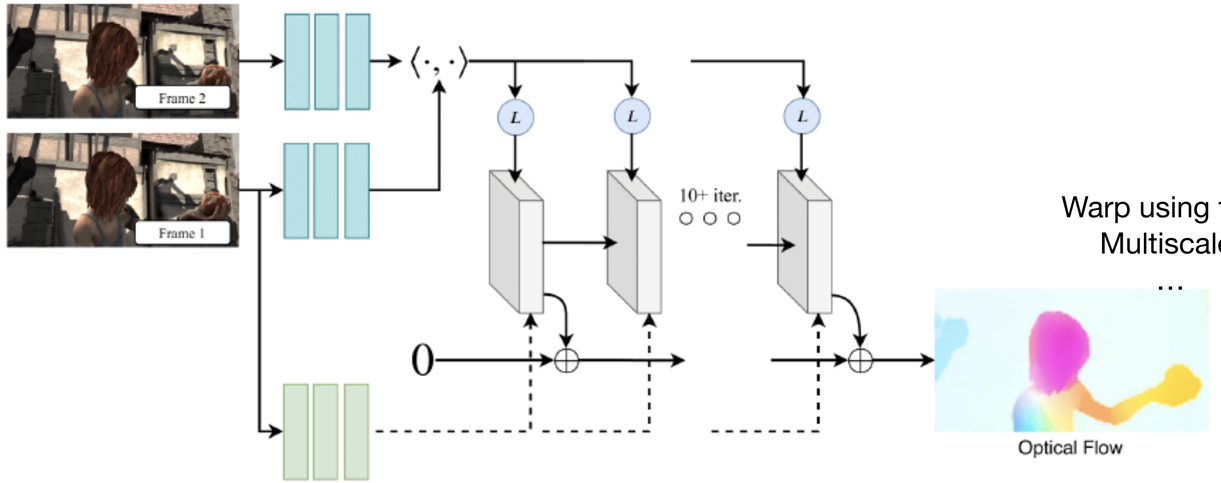
[Stone, Maurer et al. 2021, Jonschkowski et al. 2020, Meister et al. 2018, …]
(see SMURF for good discussion!)



Estimate flow with Neural Network…

Self-Supervised Optical Flow

[Stone, Maurer et al. 2021, Jonschkowski et al. 2020, Meister et al. 2018, …]
(see SMURF for good discussion!)



Warp using flow,
Multiscale,
…



$$\sum_{\mathbf{p}} \left| I_t(\mathbf{p}) - I_{t+1}\left(\mathbf{p} + \mathbf{w}_\mathbf{p}\right) \right|^2 + \lambda \sum_{\mathbf{q} \in N_p} \left| \mathbf{w}_\mathbf{p} - \mathbf{w}_\mathbf{q} \right|^2 + \dots$$

Estimate flow with Neural Network…

…but *supervise* flow with warping-based losses & regularizers!

SMURF

This CVPR 2021 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

SMURF: Self-Teaching Multi-Frame Unsupervised RAFT with Full-Image Warping

Austin Stone*[1] Daniel Maurer*[2] Alper Ayvaci[2] Anelia Angelova[1] Rico Jonschkowski[1]
[1]Robotics at Google [2]Waymo
{austinstone, anelia, rjon}@google.com {maurerd, ayvaci}@waymo.com

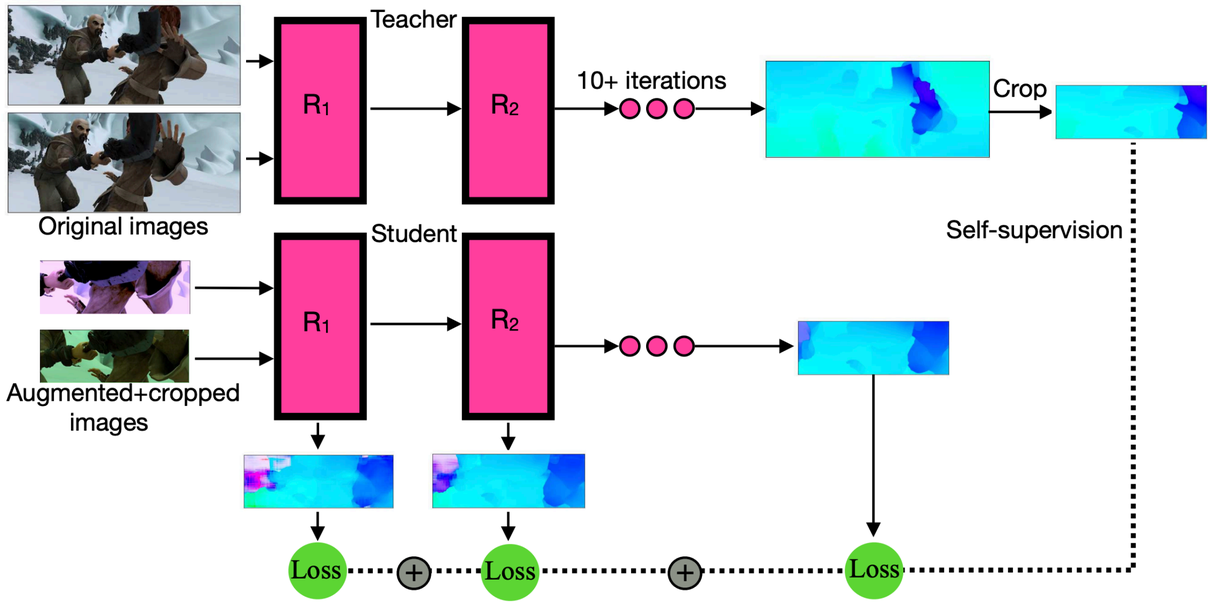
https://github.com/google-research/google-research/tree/master/smurf

Abstract

We present SMURF, a method for unsupervised learning of optical flow that improves state of the art on all benchmarks by 36% to 40% (over the prior best method UFlow) and even outperforms several supervised approaches such as PWC-Net and FlowNet2. Our method integrates architecture improvements from supervised optical flow, i.e. the RAFT model, with new ideas for unsupervised learning that include a sequence-aware self-supervision loss, a technique for handling out-of-frame motion, and an approach for learning effectively from multi-frame video data while still only requiring two frames for inference.

Unsupervised learning is a promising direction to address this issue as it allows training optical flow models from unlabeled videos of any domain. The unsupervised approach works by combining ideas from classical methods and supervised-learning – training the same neural networks as in supervised approaches but optimizing them with objectives such as smoothness and photometric similarity from classical methods. Unlike those classical methods, unsupervised approaches perform optimization not per image pair but jointly for the entire training set.

Since unsupervised optical flow takes inspiration from classical and supervised learning methods, we can make substantial progress by properly combining novel ideas with insights from these two directions. In this paper, we do exactly that and make the following three contributions:



Motivation:

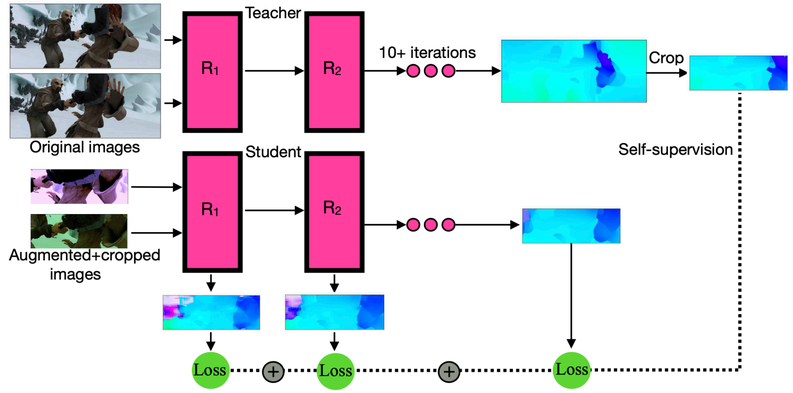
1. The model learns to ignore photometric augmentations
2. The model learns to make better predictions at the borders and in occluded areas of the image (with other tricks, see paper)
3. Early iterations of the recurrent model learn from the output at the final iteration

Loss

$$\mathcal{L}(D,\theta) = \omega_{photo}\mathcal{L}_{photo}(D,\theta) + \omega_{smooth}\mathcal{L}_{smooth}(D,\theta) + \omega_{self}\mathcal{L}_{self}(D,\theta),$$

Essentially the loss is where the inductive bias is

$$\mathcal{L}_{photo}(D,\theta) = \frac{1}{HW}\sum O_1 \odot \rho\Big(I_1, w\big(I_2, V_1\big)\Big)$$



Architecture itself mostly RAFT

$$\mathcal{L}_{smooth(k)}(D,\theta) = \frac{1}{HW}\sum\left(\exp\left(-\frac{\lambda}{3}\sum_c\left|\frac{\partial I_{1_c}}{\partial x}\right|\right)\odot\left|\frac{\partial^k V_1}{\partial x^k}\right| + \exp\left(-\frac{\lambda}{3}\sum_c\left|\frac{\partial I_{1_c}}{\partial y}\right|\right)\odot\left|\frac{\partial^k V_1}{\partial y^k}\right|\right).$$

$$\mathcal{L}_{self}(D,\theta) = \frac{1}{HW}\sum \hat{M}_1 \odot \Big(1 - M_1\Big) \odot c\Big(\hat{V}_1, V_1\Big)$$

Loss

$$\mathcal{L}(D,\theta) = \omega_{photo}\mathcal{L}_{photo}(D,\theta) + \omega_{smooth}\mathcal{L}_{smooth}(D,\theta) + \omega_{self}\mathcal{L}_{self}(D,\theta),$$

Still fair amount of inductive bias in loss and architecture

Element-wise multiplication

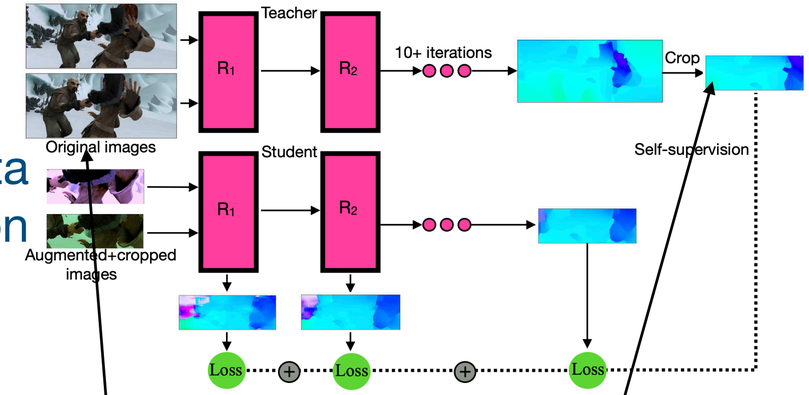
Warp

Flow field

$$\mathcal{L}_{photo}(D,\theta) = \frac{1}{HW}\sum O_1 \odot \rho\Big(I_1, w\big(I_2, V_1\big)\Big)$$

Occlusion mask based on off-the-shelf inferred depth

Fancy photometric (pixel) loss based on census transform (brightness ordering rather than value)



Architecture itself mostly RAFT

$$\mathcal{L}_{smooth(k)}(D,\theta) = \frac{1}{HW}\sum\left(\exp\left(-\frac{\lambda}{3}\sum_c\left|\frac{\partial I_{1_c}}{\partial x}\right|\right)\odot\left|\frac{\partial^k V_1}{\partial x^k}\right| + \exp\left(-\frac{\lambda}{3}\sum_c\left|\frac{\partial I_{1_c}}{\partial y}\right|\right)\odot\left|\frac{\partial^k V_1}{\partial y^k}\right|\right).$$

Edge-aware regularization based on image and flow derivatives

$$\mathcal{L}_{self}(D,\theta) = \frac{1}{HW}\sum \hat{M}_1 \odot \Big(1 - M_1\Big) \odot c\Big(\hat{V}_1, V_1\Big)$$

Leverage flow $\widehat{V}_1$ computed on uncropped version of frame to improve things at frame boundary

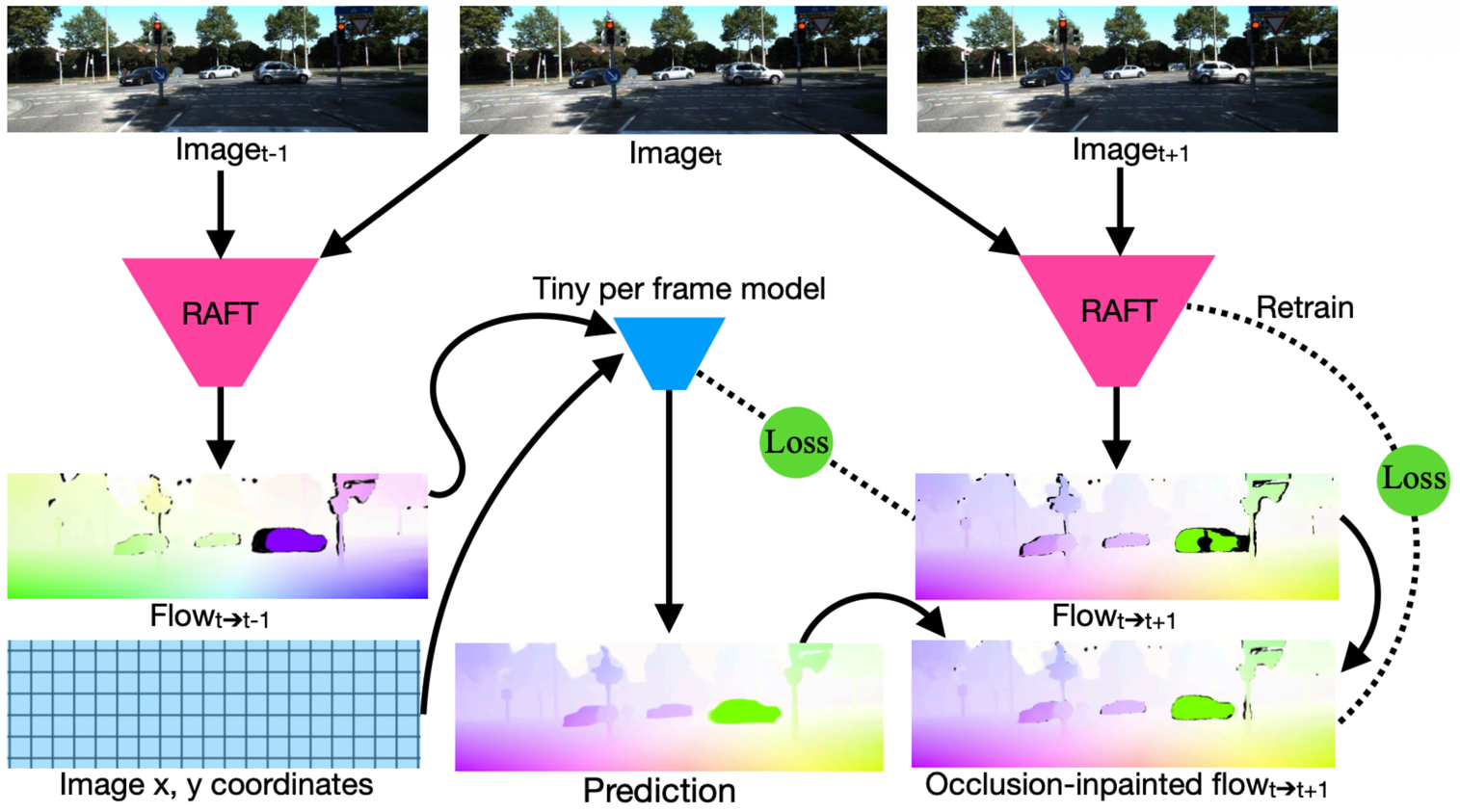


Figure 3. Multi-frame self-supervision. From a sequence of three frames $t-1$, t, and $t+1$, we compute the backward flow ($t \to t-1$) and the forward flow ($t \to t+1$). The backward flow is then inverted via a tiny model (which is trained for this frame pair) and used to inpaint occluded regions in the forward flow. This inpainted flow field is then used to retrain the RAFT model.

SMURF results

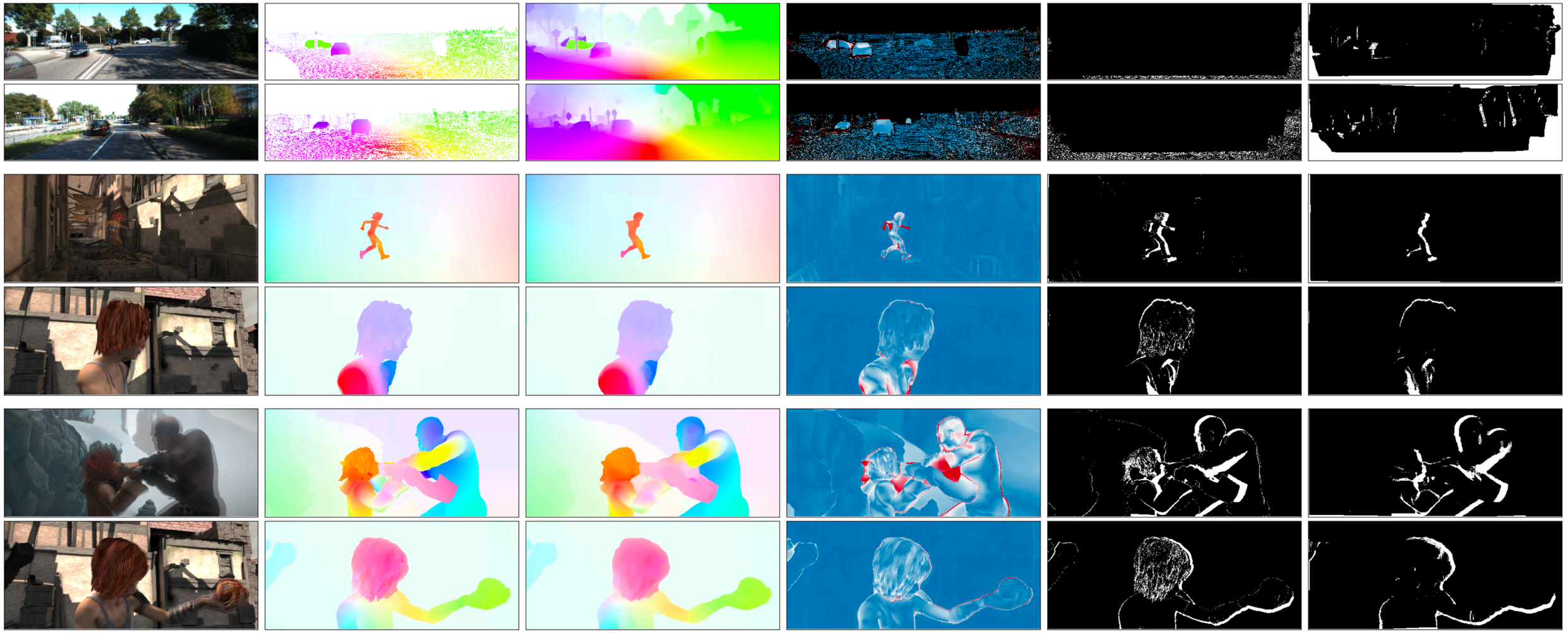


Figure 4. Qualitative results for SMURF on two random examples from KITTI 2015, Sintel Clean, and Sintel Final (top to bottom) not seen during training. These results show the model's ability to estimate fast motions, relatively fine details, and substantial occlusions.

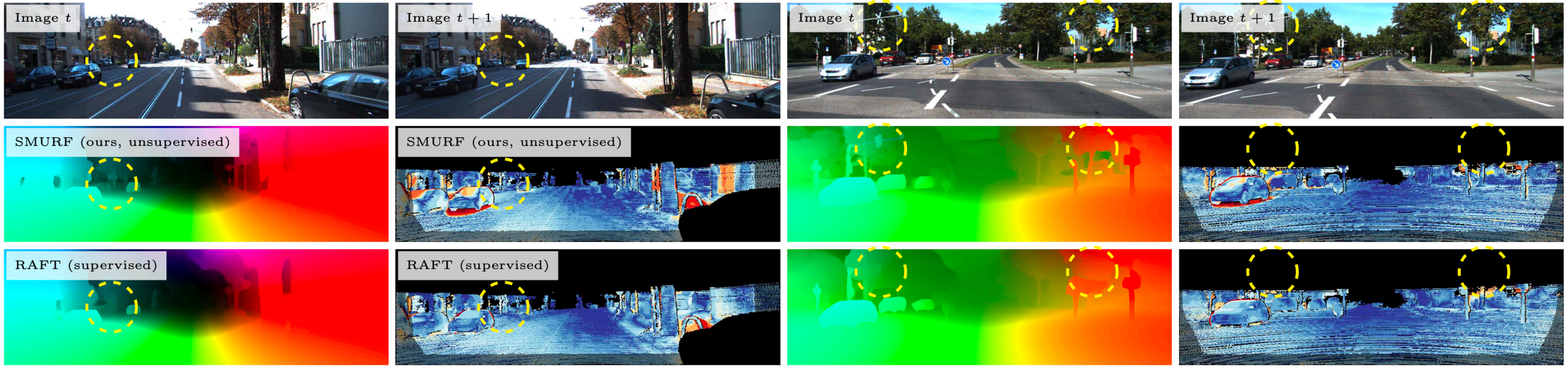


Figure 5. Qualitative comparison of unsupervised SMURF and supervised RAFT. Maybe unsurprisingly supervised training in domain reduces errors especially for challenging and ambiguous cases, e.g. reflections and shadows of moving objects. But interestingly, there are also areas (highlighted) where our unsupervised method works better than the supervised alternative, apparently at small objects in places where labels are sparse and potentially imperfect (left) or non-existent (right).

Transformers

Transformers are good at non-local stuff

But quadratic cost makes them challenging with 4D cost volume

Requires smart encoding

PERCEIVER IO: A GENERAL ARCHITECTURE FOR STRUCTURED INPUTS & OUTPUTS

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu,

David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier Hénaff,

Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, João Carreira

DeepMind

FlowFormer: A Transformer Architecture for Optical Flow

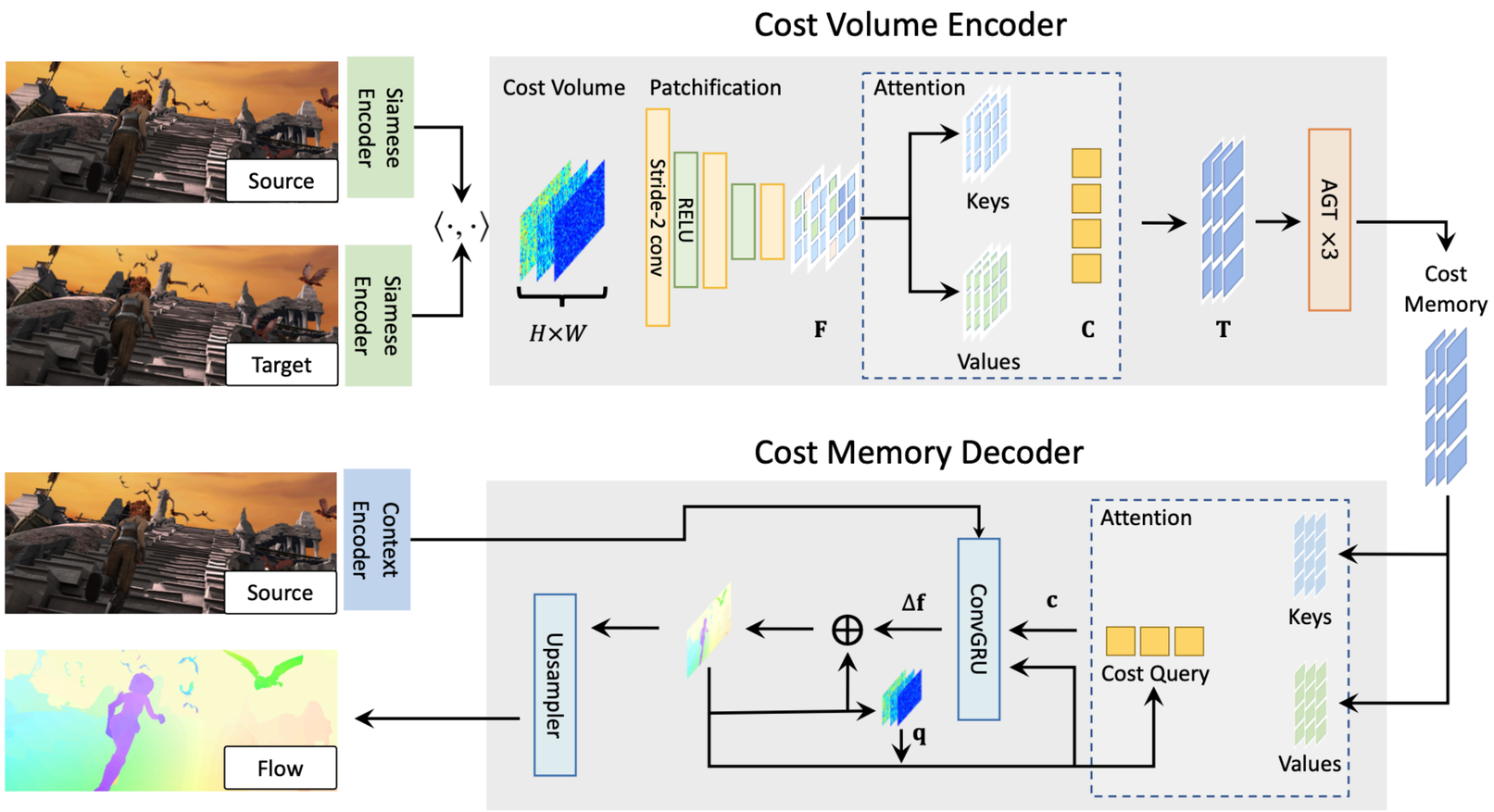
Zhaoyang Huang[1,3*], Xiaoyu Shi[1,3⋆], Chao Zhang[2], Qiang Wang[2], Ka Chun Cheung[3], Hongwei Qin[4], Jifeng Dai[4], and Hongsheng Li[1†]

FlowFormer++: Masked Cost Volume Autoencoding for Pretraining Optical Flow Estimation

Xiaoyu Shi[1,2*] Zhaoyang Huang[1,2*] Dasong Li[1] Manyuan Zhang[1]
Ka Chun Cheung[2] Simon See[2] Hongwei Qin[3] Jifeng Dai[4] Hongsheng Li[1†]

[1]Multimedia Laboratory, The Chinese University of Hong Kong
[2]NVIDIA AI Technology Center [3]SenseTime Research [4]Tsinghua University

Flowformer



Cost volume

Patchification + tokenization

-> latent 4D cost volume

Alternate-Group Transformer Layer

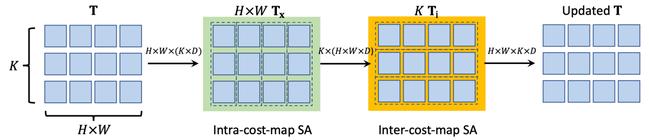


Fig. 2: Alternate-Group Transformer Layer. The alternate-group transformer layer (AGT) alternatively groups tokens in $\mathbf{T}$ into $H \times W$ groups that contains K tokens ($\mathbf{T}_x$) and K groups that contains $H \times W$ tokens ($\mathbf{T}_i$), and encode tokens inside groups via self-attention and ss self-attention [8] respectively.

Memory, self attention

Decoder still recurrent

Fig. 1: Architecture of FlowFormer. FlowFormer estimates optical flow in three steps: 1) building a 4D cost volume from image features. 2) A cost volume encoder that encodes the cost volume into the cost memory. 3) A recurrent transformer decoder that decodes the cost memory with the source image context features into flows.

FlowFormer++

Better cost volume encoding with Masked AutoEncoder

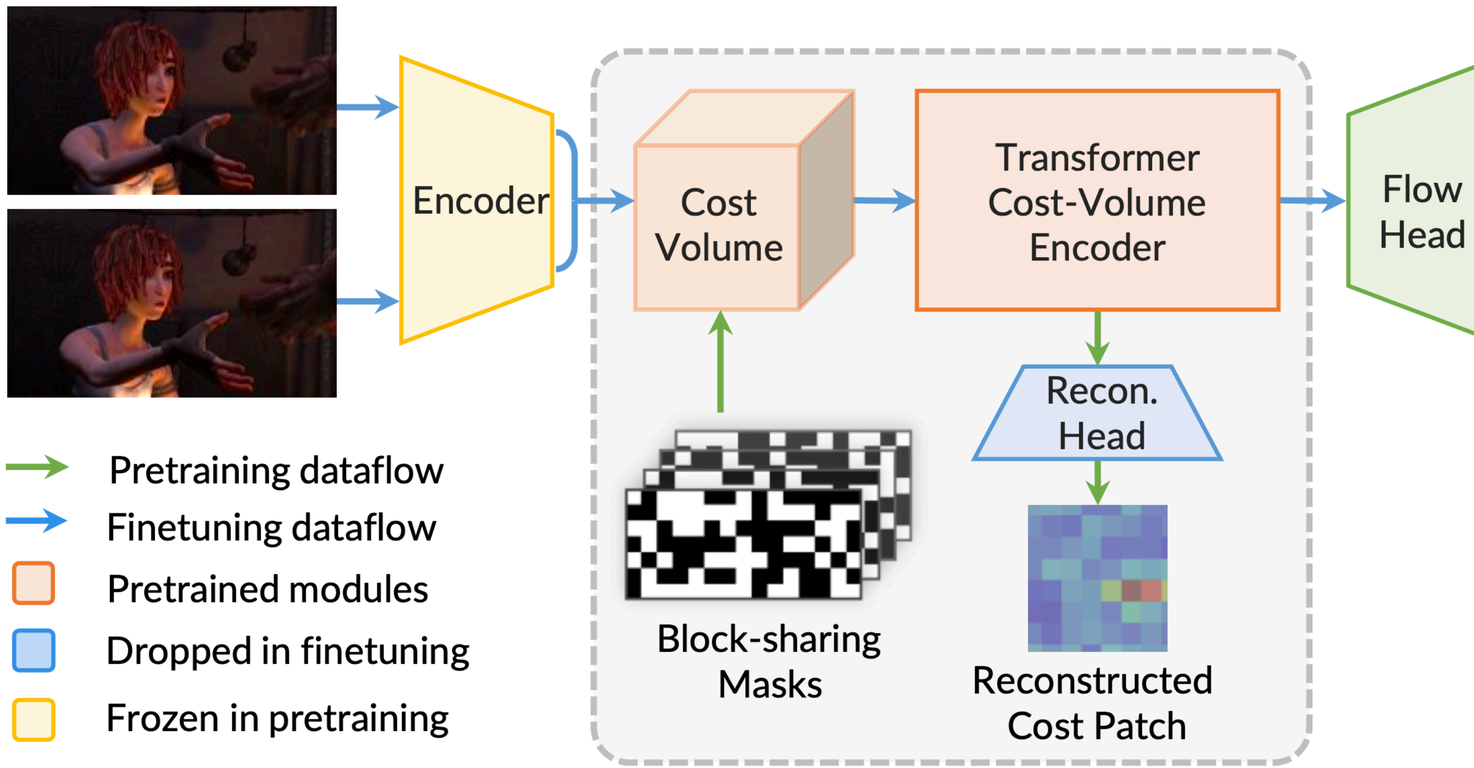


Figure 1. **Overview of FlowFormer++.** The core compone
FlowFormer architecture is the transformer-based cost-volum
coder. We propose Masked Cost Volume Autoencoding to pre
the cost-volume encoder. During pretraining, a portion of cos
ues are masked and the cost-volume encoder is required to re
struct masked cost patches.

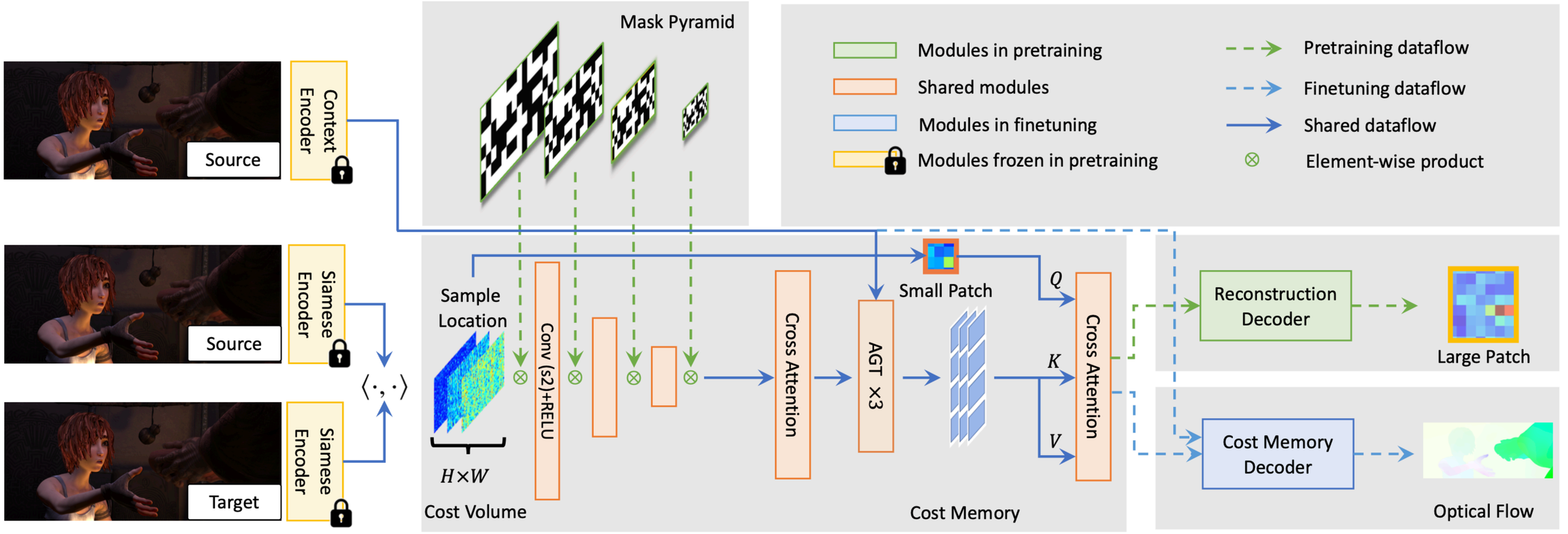


Figure 2. **Architecture of FlowFormer++.** During pertaining, FlowFormer++ freezes the image and context encoders, block-wisely masks the cost volume, and learns to reconstruct larger cost patches from small cost patches to pretrain cost-volume encoder. In fine-tuning, FlowFormer++ uses the full cost volume, removes the reconstruction decoder, and adds the cost memory decoder to learn optical flow, which naturally falls back to the FlowFormer architecture but inherits the pretrained parameters in the cost-volume encoder.

Flowformer discussion

Still specialized : cost volume, recurrence, special encoding

Transformers are great but memory is a huge challenge (especially with 4D data)

Optical Flow Discussion

Challenge: non-local matching, Lagrangian nature

- Sometimes convergence with depth estimate, sometimes not

Has moved from manually crafted to supervised to self-supervised

Still requires some inductive biases and specialized architecture

- Not fully bitter-lessoned.

Still a fair question whether it’s a real task

How connected is it to generative video models? Their tokenization?