Fast Guided Global Interpolation for Depth and Motion

Yu Li, Dongbo Min, Minh N. Do, Jiangbo Lu

ADSC
Illinois at Singapore Pte Ltd

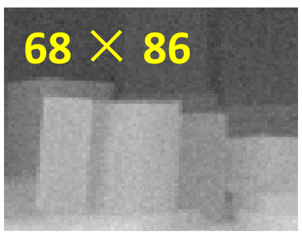
CNU
Chungnam National University

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

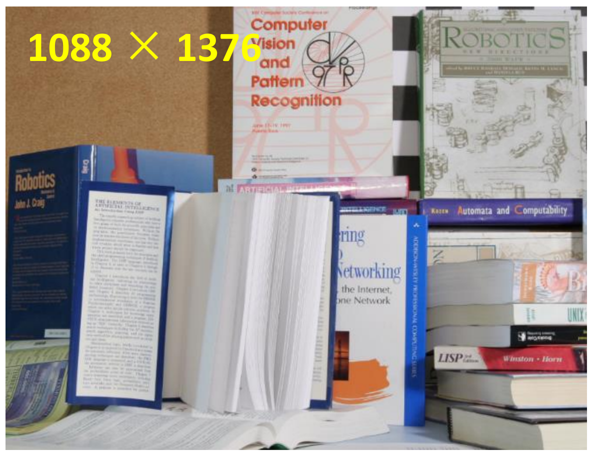
Introduction

Depth upsampling and ***motion interpolation*** are often required to generate a dense, high-quality, and high resolution depth map or optical flow field.

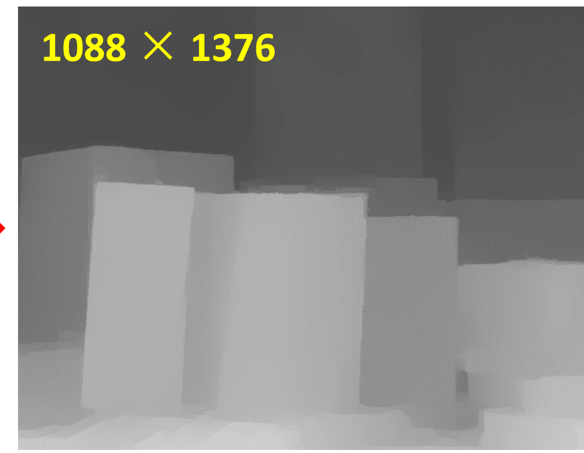
Low-res & noisy depth (ToF)



High-res. color guidance



High-res. depth



Depth upsampling (color guided)

Input TOF depth: noisy, low resolution, regularly distributed

Introduction

Depth upsampling and ***motion interpolation*** are often required to generate a dense, high-quality, and high resolution depth map or optical flow field.

Color frame and sparse matches from DM

Data density < 1%

Dense optical flow field

Motion interpolation

Input matches: typically reliable, but highly scattered, varying density

*[**DM**: Weinzaepfel et al., "DeepFlow: Large displacement optical flow with deep matching, ICCV 2013.]*

Motivation

Existing methods are often tailored to one *specific* task:

Depth upsampling	JBF [Kopf et al. 2007], MRF+nlm [Park et al. 2011], TGV [Ferstl et al. 2013], JGU [Liu et al. 2013], AR [Yang et al. 2014], Data-driven [Kwon et al. 2015], etc
Motion interpolation	EpicFlow [Revaud et al. 2015], [Drayer and Brox 2015] , [Leordeanu et al. 2013], etc

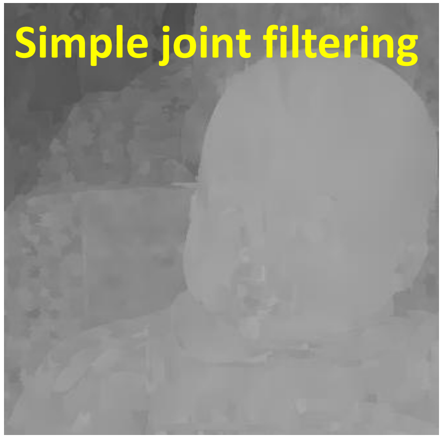
The common objective for both tasks is to ***densify*** a set of ***sparse data points***, either regularly distributed or scattered, to a full image grid through a 2D ***guided interpolation*** process.

***Our approach*: Fast Guided Global Interpolation (FGI)**

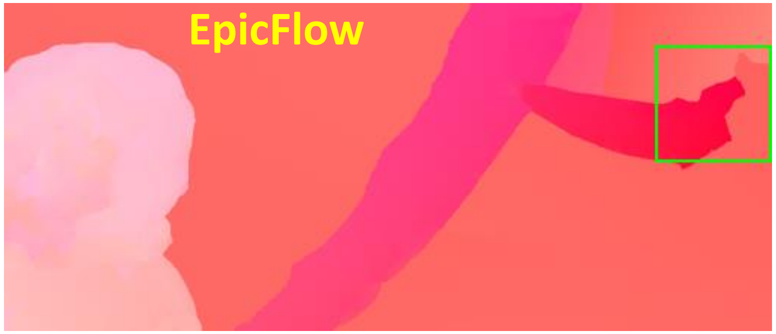
A unified approach that casts the guided interpolation problem into a hierarchical, global optimization framework.

Several Challenges e.g.

- Texture-copy artifacts due to inconsistent structures



- Large occlusions, long-range propagation and extrapolation

- Loss of thin structures, missing motion boundaries
- Complex algorithms and computationally inefficient

Single-scale WLS Method Vs. Our Method

Depth

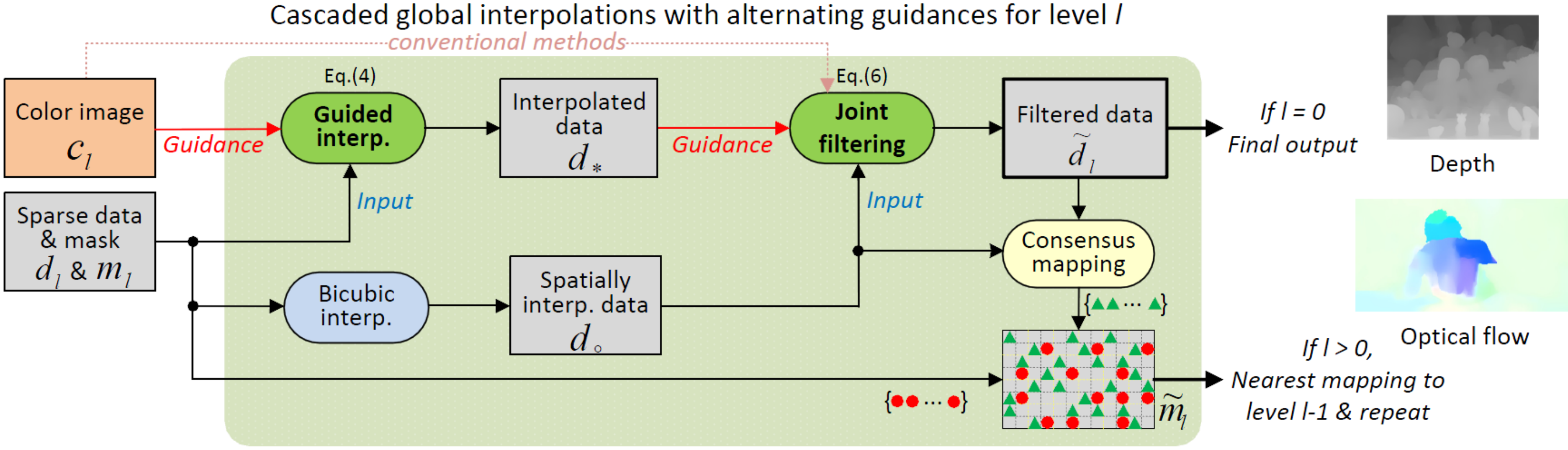
Color guidance | WLS | Our result | Ground truth

Optical flow

Color guidance | WLS | Our result | Ground truth

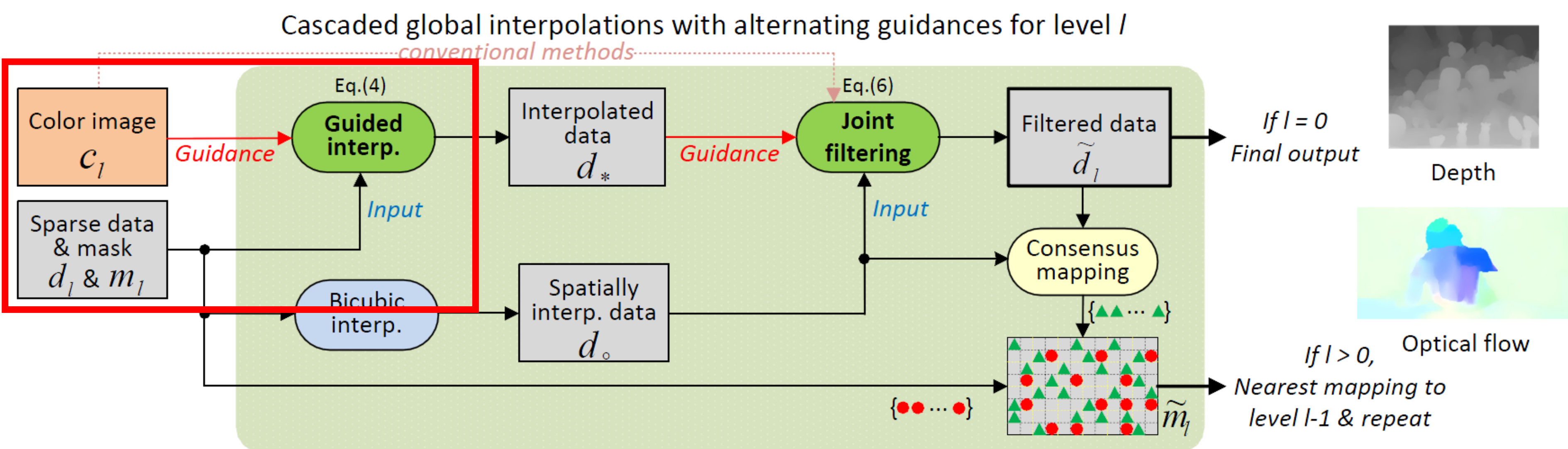
*[**WLS**: Farbman et al., “Edge-preserving decompositions for multi-scale tone and detail manipulation,” SIGGRAPH 2008]*
*[**FGS**: Min et al., "Fast global image smoothing based on weighted least squares," TIP 2014.]*

Our Pipeline: Overview



- A *hierarchical* (*coarse-to-fine*), *multi-pass* guided interpolation framework
- Divide the problems into a sequence of interpolation tasks each with smaller scale factors
- Gradually fill the large gap between the sparse measurement and the dense data

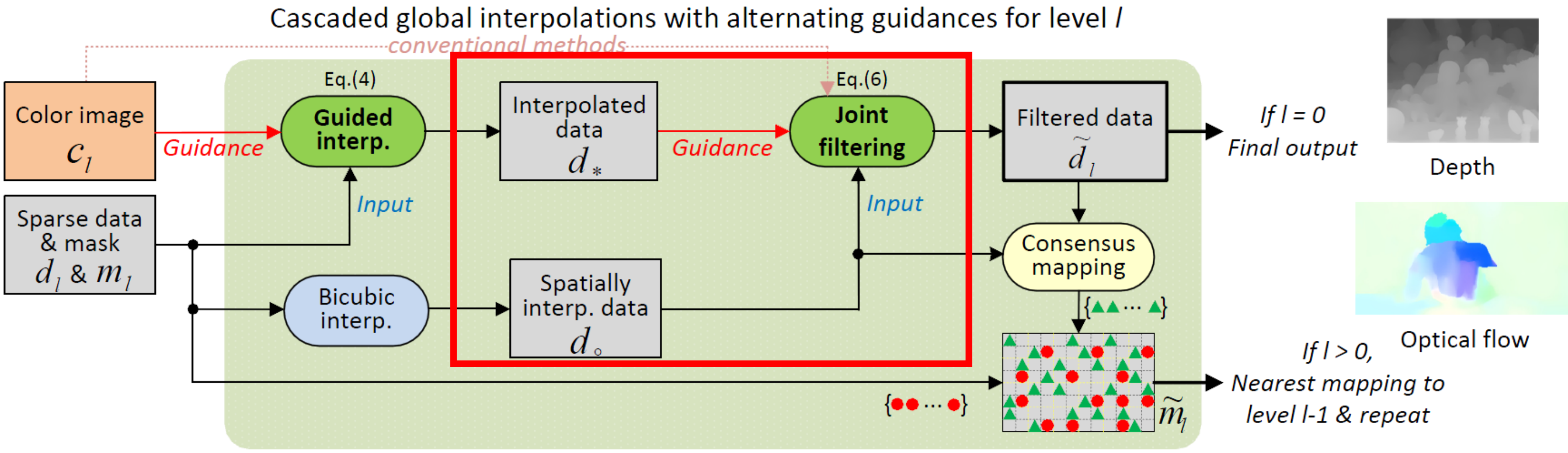
Our Pipeline: Filtering with Alternating Guidances



- From the coarse level *l*= *L*-1, we upsample the signal by a factor of 2 at each level by solving the following weighted least square (WLS) using the recent **FGS** solver.
- **Guided interp. :** $\mathcal{E}(\mathbf{d}_*) = (\mathbf{d}_* - \mathbf{d}_l)^\top \mathbf{M}_l (\mathbf{d}_* - \mathbf{d}_l) + \lambda_1 \mathbf{d}_*^\top \mathbf{A}_{c_l} \mathbf{d}_*$
  - The color image c_l as the guidance
  - $\mathbf{A}_{c_l}$ is the spatially varying Laplacian matrix defined by c_l

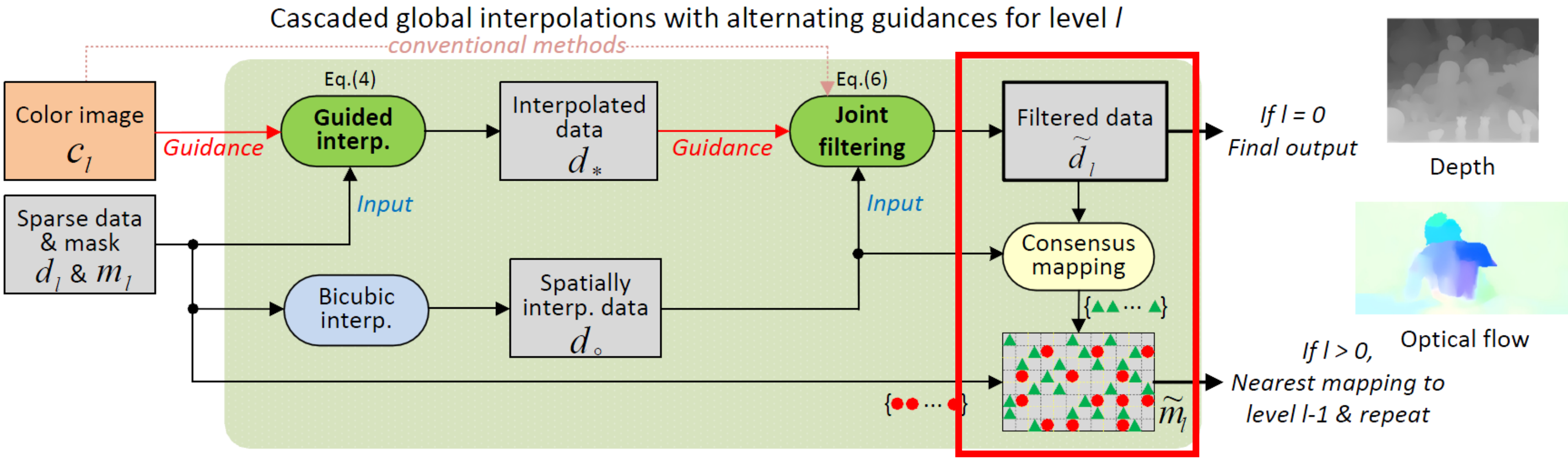
*[**FGS**: Min et al., "Fast global image smoothing based on weighted least squares," TIP 2014.]*
Why FGS? *100 ms for filteirng 1MPixels RGB images on 1 CPU core. More details in our paper.*

Our Pipeline: Filtering with Alternating Guidances



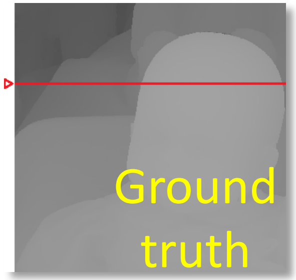
- Next, another WLS is solved with the output d_* as guidance and bicubic interpoplated signal as input.
- **Joint filtering**: $\mathcal{E}(\tilde{\mathbf{d}}_l) = (\tilde{\mathbf{d}}_l - \mathbf{d}_\circ)^\top(\tilde{\mathbf{d}}_l - \mathbf{d}_\circ) + \lambda_2 \tilde{\mathbf{d}}_l^\top \mathbf{A}_{d_*} \tilde{\mathbf{d}}_l$
  - The intermediate interpolated map d_* as the guidance
  - $\mathbf{A}_{d_*}$ is the spatially varying Laplacian matrix defined by d_*

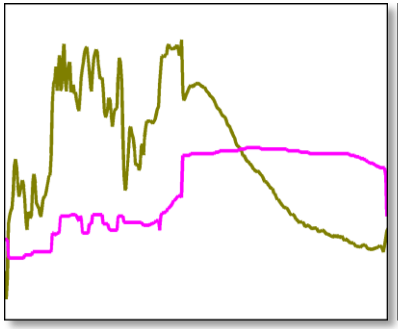
Our Pipeline: Consensus-Based Data Augmentation



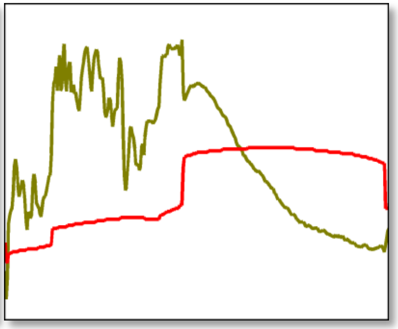
- Then, check the **consistency** between the output and the bicubic upsampled data, and pick the most consistent points to add to the data mask map $\widetilde{m}_l$
  - The bibubic upsampled data is free from texture-copying
  - Proceed in a non-overlapping patch fashion (2x2 patches)
- The entire process is repeated until the finest level (*l* = 0) is reached.

1D Scanline Illustration



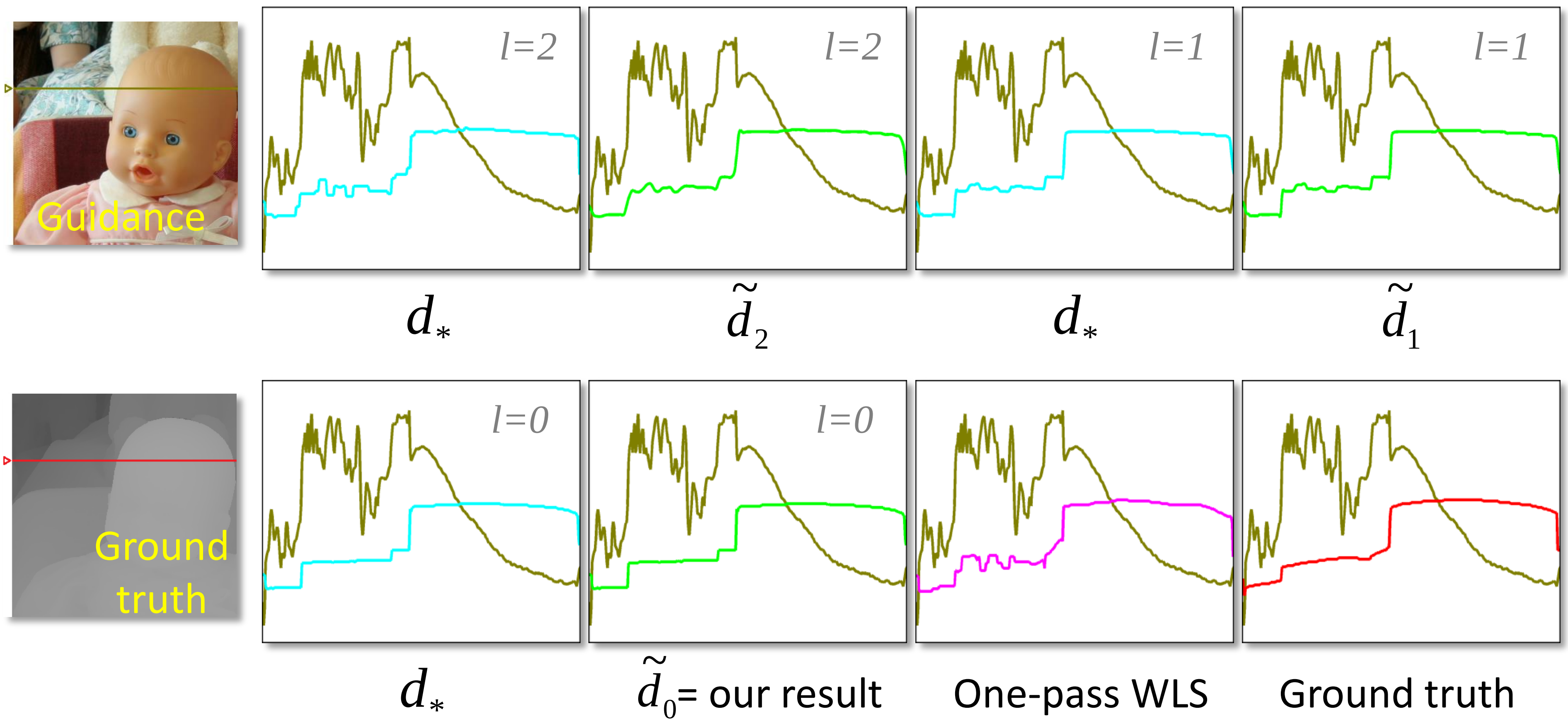


One-pass WLS

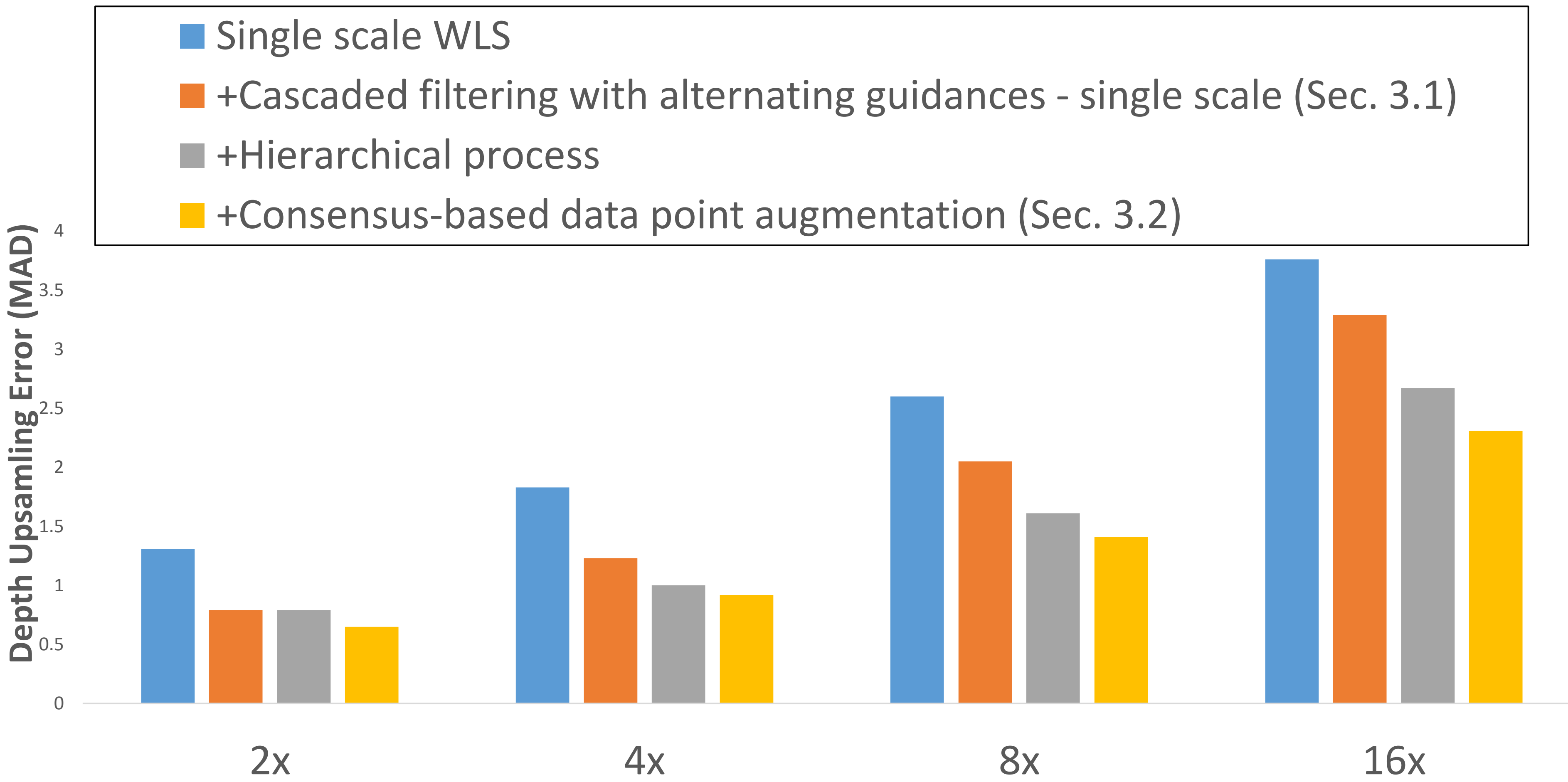


Ground truth

1D Scanline Illustration



Pipeline Validation on Depth Upsampling



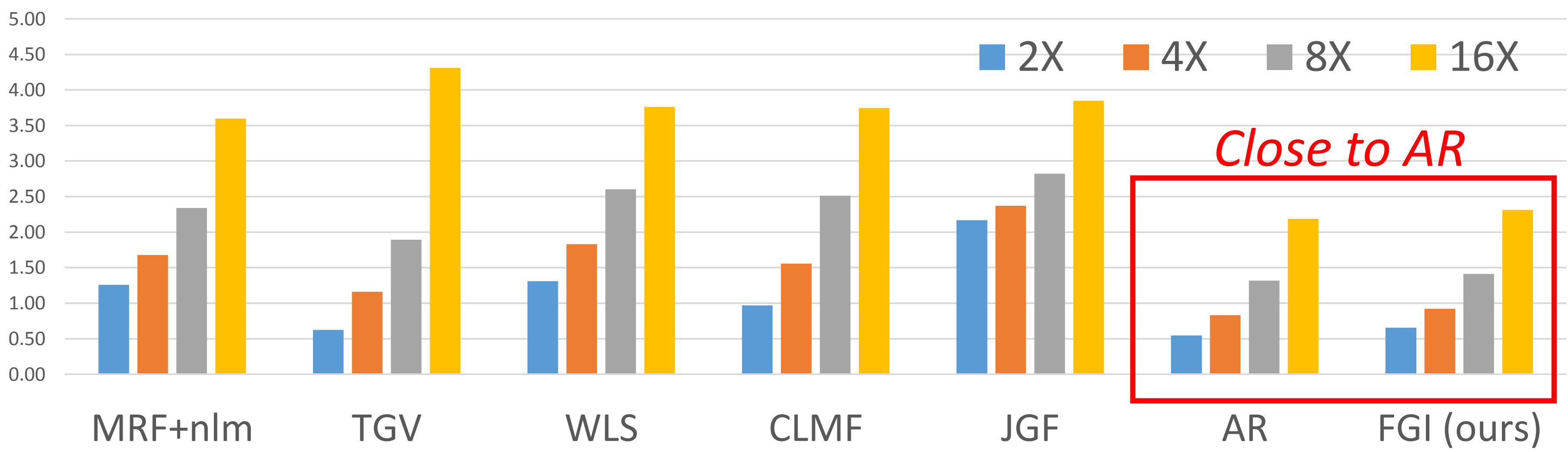
Depth Upsampling Results

Average runtime to upsample a 272 × 344 depth to 1088 × 1376 (in *seconds*)

MRF+nlm	TGV	AR	GF	CLMF	**FGI(ours)**
170	420	900	1.3	2.4	**0.6**

1000x faster than AR

Average Depth Upsampling Error on ToF Synthetic Dataset (6 cases)



Our framework also improves other edge-aware smoothing filters, e.g. the guided filter

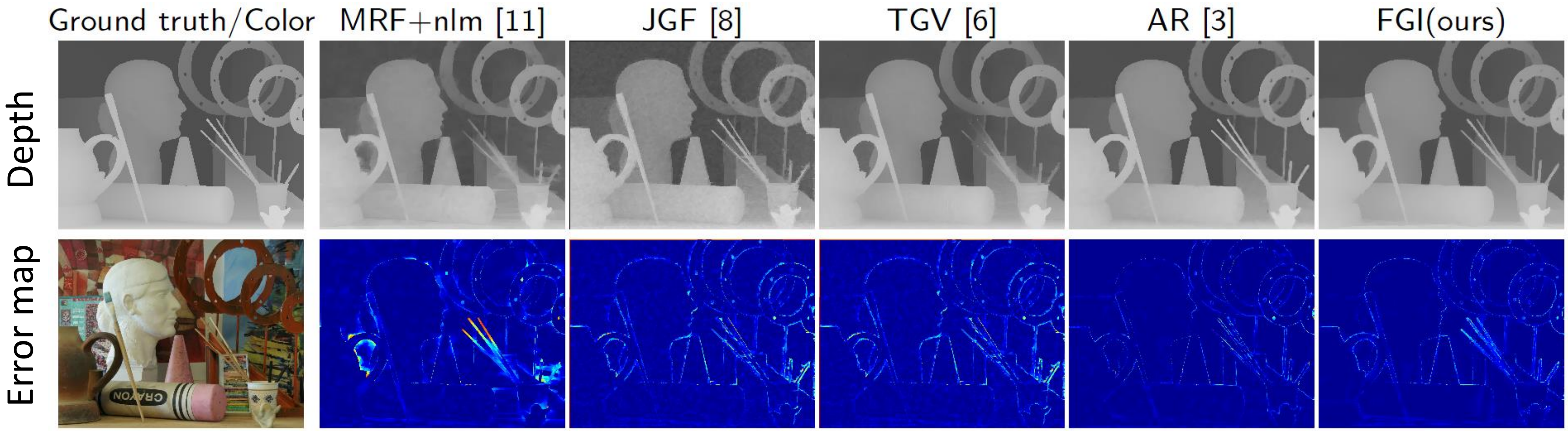
Depth avg. error	2x	4x	8x	16x
Single-pass **GF**	1.31	1.54	2.04	3.12
GF in our framework	**1.06**	**1.21**	**1.63**	**2.59**

*[**GF**: He et al., "Guided image filtering," ECCV 2010.]*

Depth Upsampling Results

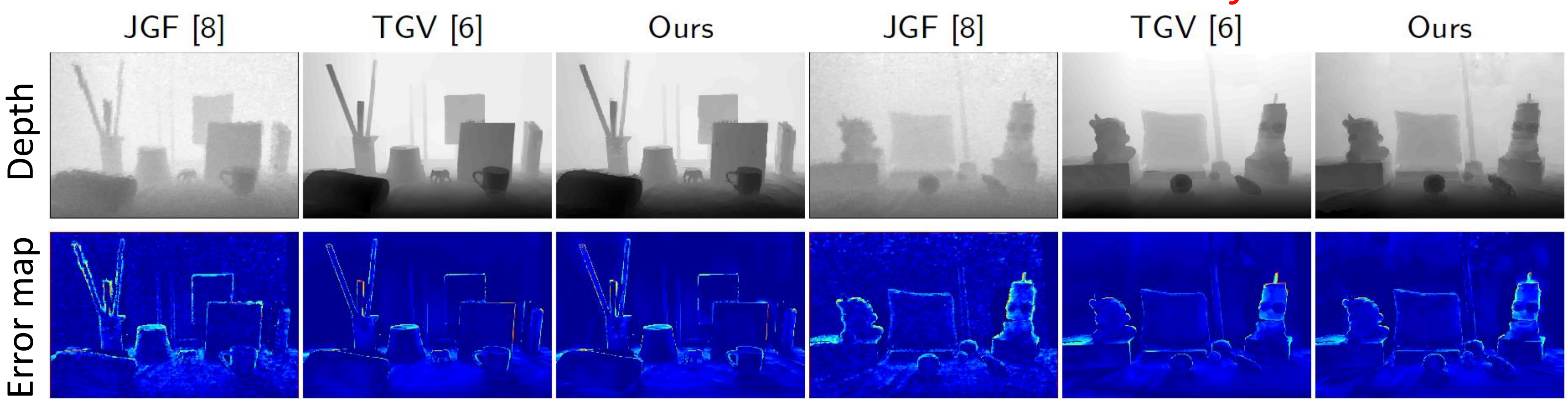
1000x faster than AR

ToF Synthetic Dataset



ToFMark Dataset

650x faster than TGV

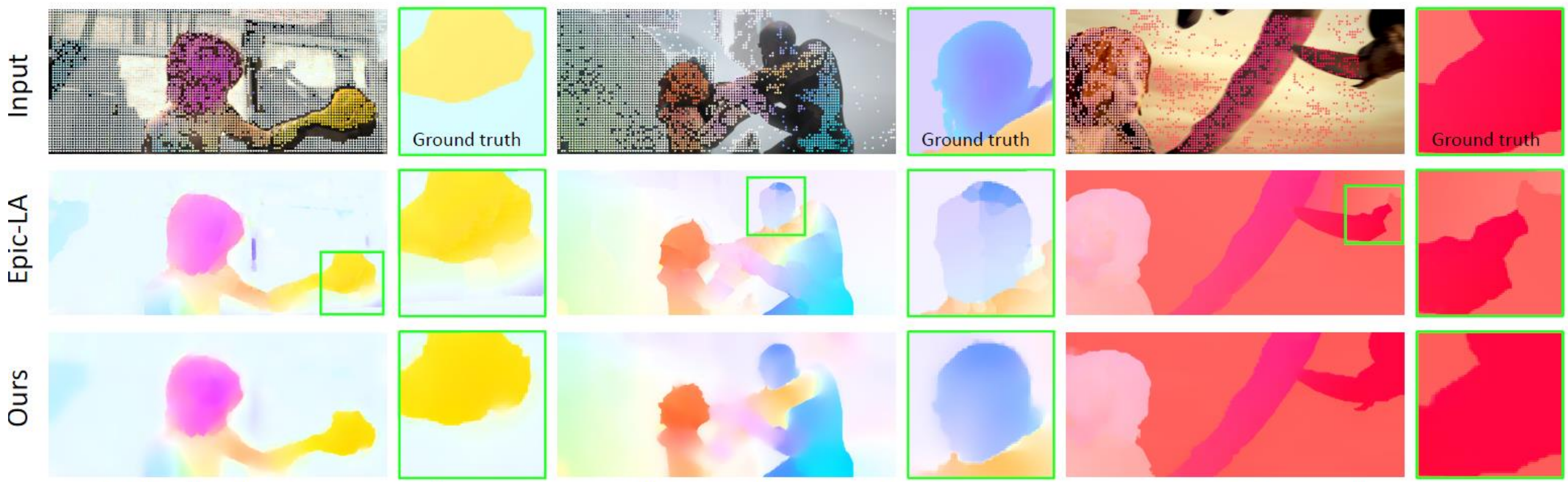


Motion Interpolation Results

Performance (EPE) on MPI Sintel training set

	WLS	EpicFlow-NW	EpicFlow-LA	FGI(ours)
Clean	3.23	3.17	2.65	2.75
Final	4.68	4.55	4.10	4.14
Runtime (sec)	0.21	0.80	0.94	0.39

Close to EpicFlow, but over 2x faster



(a) Weak edge in color guidance (b) Sparsely scattered points (c) Extrapolation

Performance (EPE) on the MPI Sintel testing benchmark

	FlowFields[13]	EpicFlow[2]	PH-Flow[37]	**FGI (ours)**	Deep+R[15]	SPM-BP[38]	DeepFlow[14]	PCA-Layers[39]	MDP-Flow2[40]
Clean	3.748	4.115	4.388	4.664	5.041	5.202	5.377	5.730	5.837
Final	5.810	6.285	7.423	6.607	6.769	7.325	7.212	7.886	8.445

Conclusion

- **General & versatile technique**:
  - Tackle **both** depth and motion interpolation tasks, and potentially more
  - Generally applicable to other edge-aware smoothing filters, e.g. GF
- **Competitive results** while running **much faster** than task-specific state-of-the-art methods
- **Simple & effective**:
  - No *color edge detection* & *variational minimization* in [Revaud et al., CVPR'15]
  - No *domain transform filtering* for post-smoothing in [Barron & Poole, ECCV'16]
- **Further acceleration** on GPUs and FPGA, offering a common engine for guided interpolation

Project page (code is available):

http://publish.illinois.edu/visual-modeling-and-analytics/fast-guided-global-interpolation/