COMPUTER VISION University of Freiburg
http://lmb.informatik.uni-freiburg.de

PAPER + POSTER

FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks

Eddy Ilg ilg@cs.uni-freiburg.de
Nikolaus Mayer mayern@cs.uni-freiburg.de
Tonmoy Saikia saikiat@cs.uni-freiburg.de
Margret Keuper keuper@cs.uni-freiburg.de
Alexey Dosovitskiy dosovits@cs.uni-freiburg.de
Thomas Brox brox@cs.uni-freiburg.de

CODE

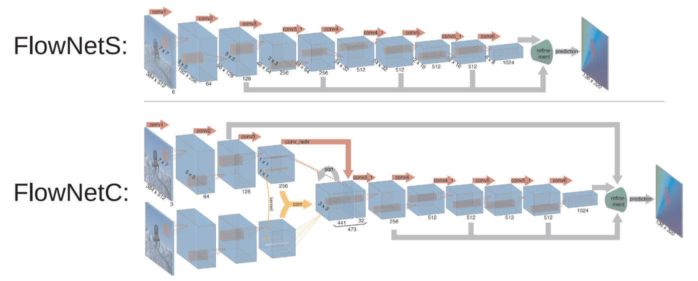
CVPR July 21-26 HONOLULU 2017

Contributions

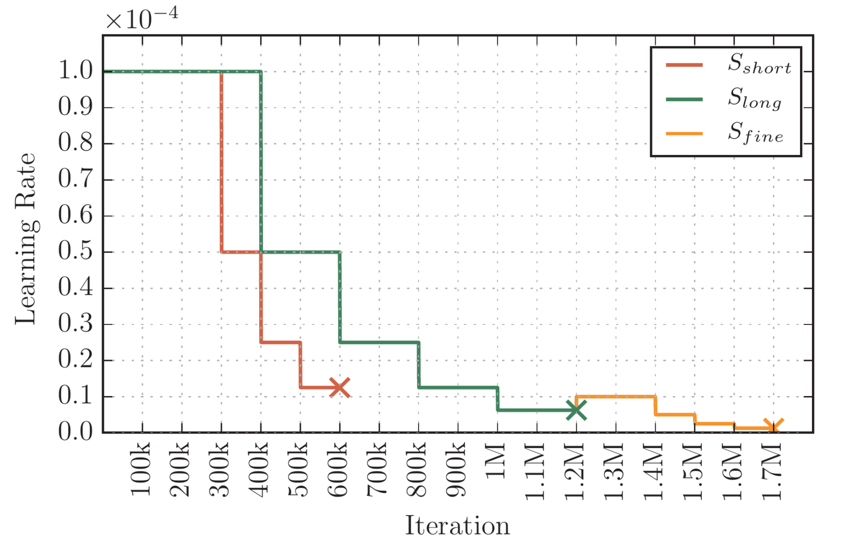
Goal: Improving optical flow estimation with FlowNet

Contributions:
- Dataset schedules
- Stacking multiple networks
- Small displacements and real data

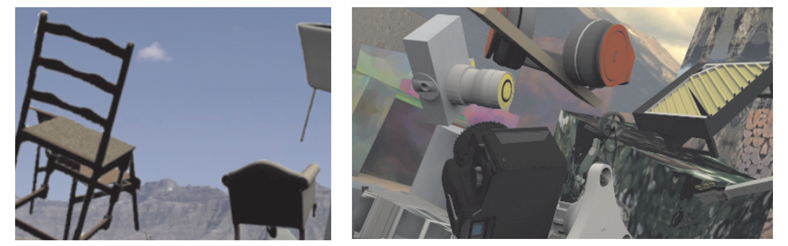
Results:
Performance on-par with other methods but at **8 to 140fps**



Dataset Schedules



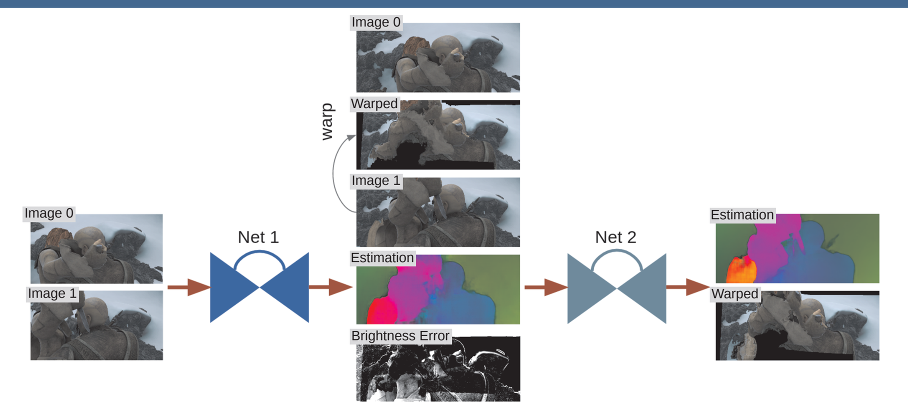
Architecture	Datasets	S_{short}	S_{long}	S_{fine}
FlowNetS	Chairs	4.45	-	-
	Chairs	-	4.24	4.21
	Things3D	-	5.07	4.50
	mixed	-	4.52	4.10
	Chairs→Things3D	-	4.24	**3.79**
FlowNetC	Chairs	3.77	-	-
	Chairs→Things3D	-	3.58	**3.04**



Observations:

- Best results can only be achieved when training first on Chairs and then on Things3D
- FlowNetC outperforms FlowNetS
- By modifying dataset schedules, results improve by 25-30%

Stacking Networks



Stack architecture	Training enabled Net1	Training enabled Net2	Warping included	Warping gradient enabled	Loss after Net1	Loss after Net2	EPE on Chairs test	EPE on Sintel train *clean*
Net1	✓	–	–	–	✓	–	3.01	3.79
Net1 + Net2	✗	✓	✗	–	–	✓	2.60	4.29
Net1 + Net2	✓	✓	✗	–	✗	✓	2.55	4.29
Net1 + Net2	✓	✓	✗	–	✓	✓	2.38	3.94
Net1 + W + Net2	✗	✓	✓	–	–	✓	1.94	**2.93**
Net1 + W + Net2	✓	✓	✓	✓	✗	✓	1.96	3.49
Net1 + W + Net2	✓	✓	✓	✓	✓	✓	**1.78**	3.33

State-of-the-art approaches rely on iterative methods. We stack two FlowNetS and make the following observations:

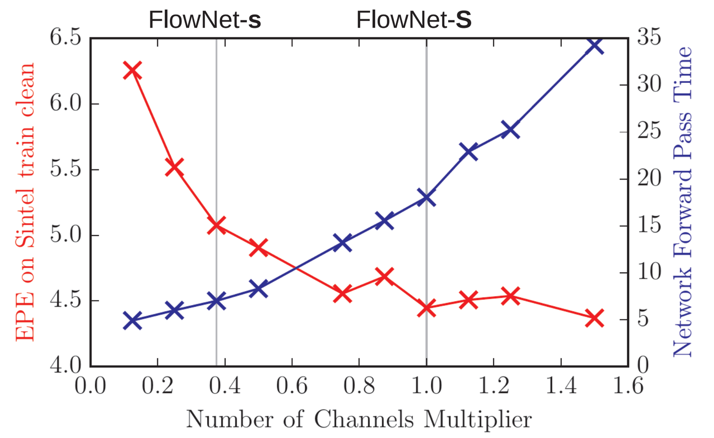
(1) Just stacking networks over-fits

(2) Including warped images as input to the second network always improves results

(3) Adding intermediate losses helps when training end-to-end

(4) Best results are obtained when keeping the first network fixed and only training the second network

Stack Configurations

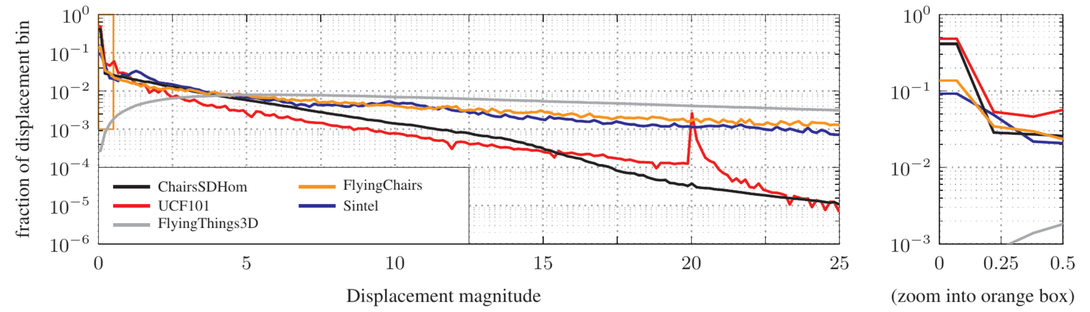


We scale the network by multiplying the numbers of channels by a constant. Using a factor of 0.375 (FlowNet-s) yields much faster networks with only slightly higher error.

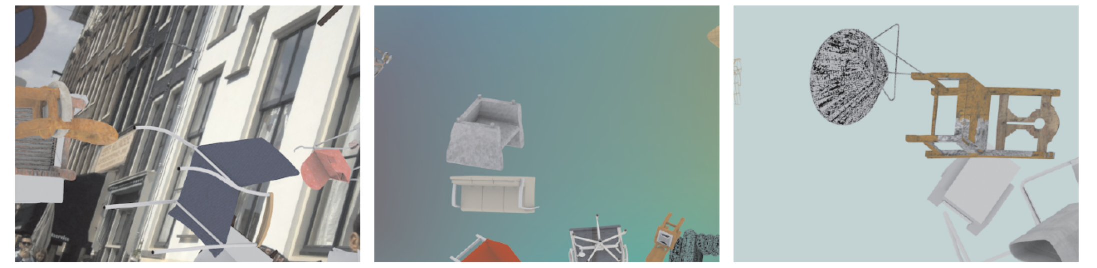
	Number of Networks 1	2	3	4
Architecture	s	ss	sss	–
Runtime	**7ms**	14ms	20ms	
EPE	4.55	3.22	3.12	
Architecture	S	SS	–	–
Runtime	18ms	37ms		
EPE	3.79	2.56		
Architecture	c	cs	css	csss
Runtime	17ms	24ms	31ms	36ms
EPE	3.62	2.65	2.51	2.49
Architecture	C	CS	CSS	–
Runtime	33ms	51ms	69ms	
EPE	3.04	2.20	**2.10**	

We then stack networks of different configurations and sizes obtaining runtimes from 7-140fps. Stacking more than three networks does not prove useful. The best result is obtained by stacking a FlowNetC with two FlowNetS on top.

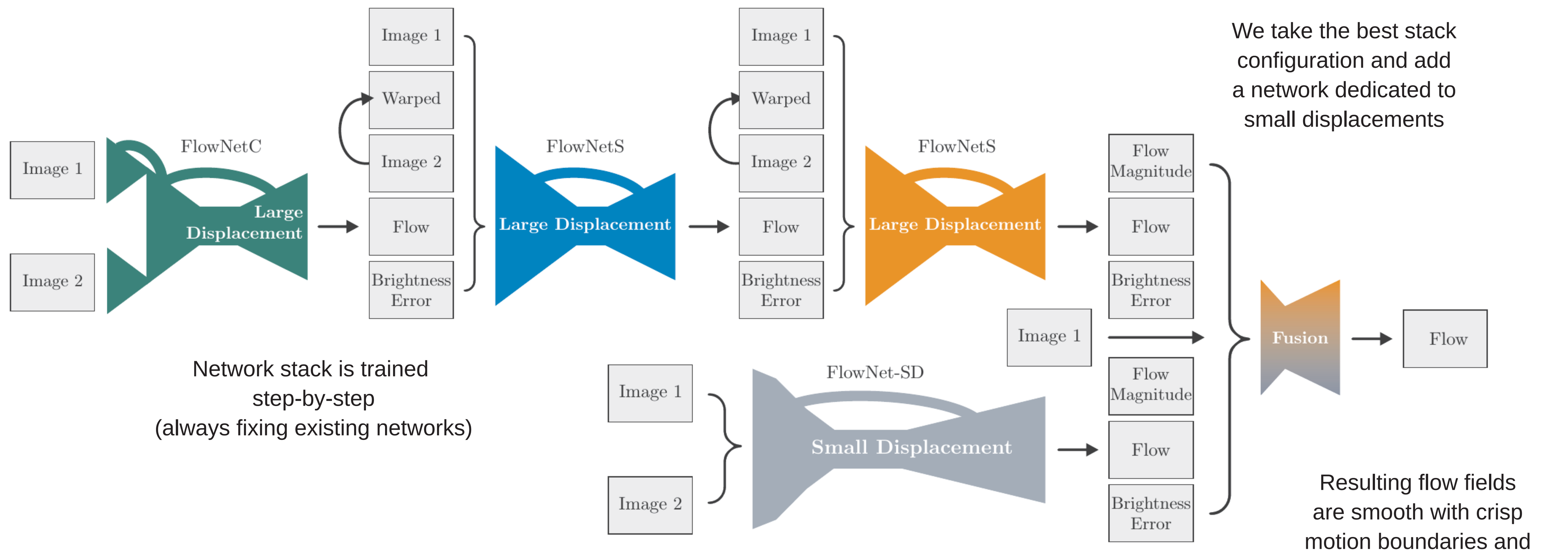
Small Displacements



- Analyzing the UCF dataset as one candidate of real-world data reveals that it on average has very small displacements
- We create an according dataset named ChairsSDHom
- Additionally, we cover the problem with featureless and feature-poor backgrounds by introducing samples with motionless blank and weak gradient backgrounds



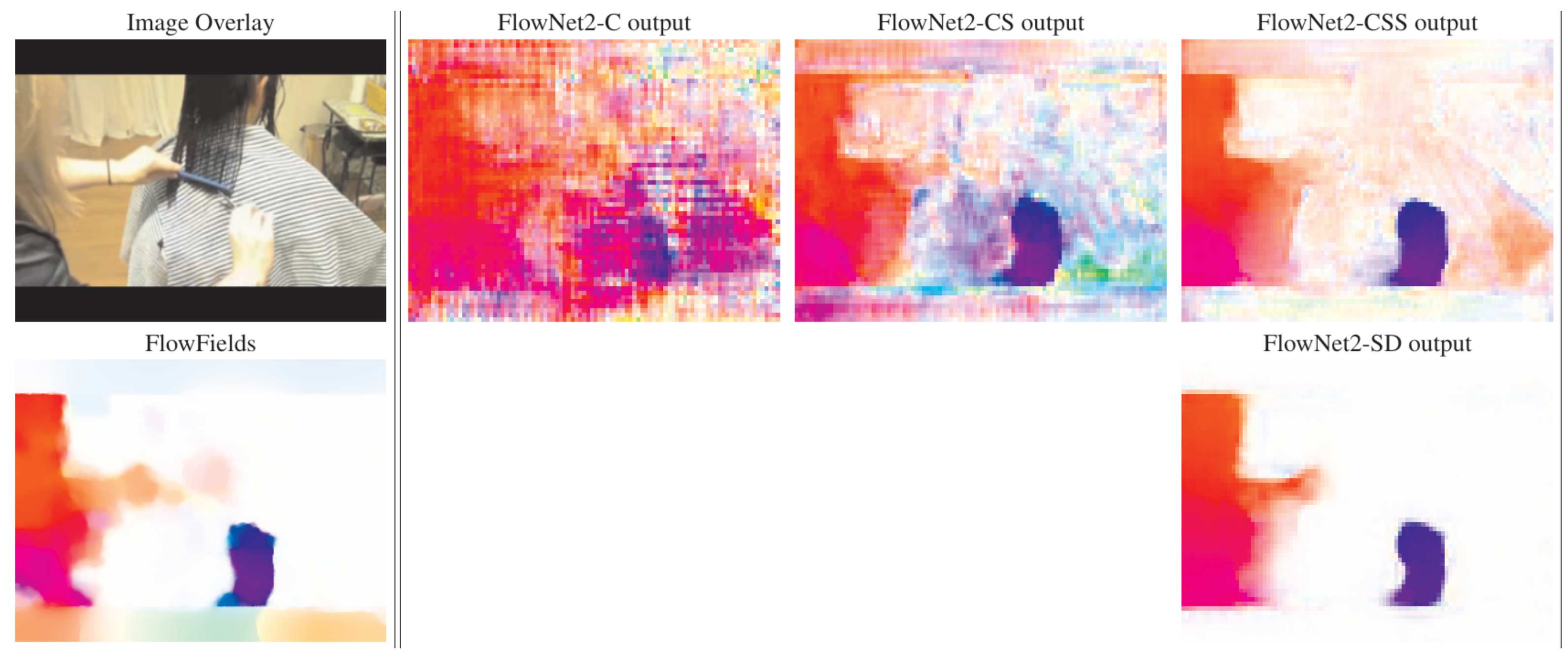
FlowNet2 Architecture



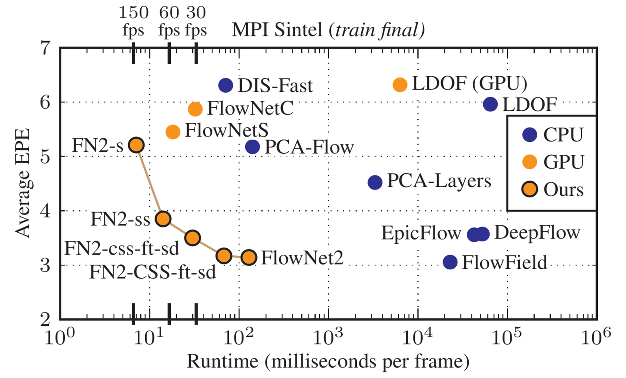
Network stack is trained step-by-step (always fixing existing networks)

We take the best stack configuration and add a network dedicated to small displacements

Resulting flow fields are smooth with crisp motion boundaries and performance is on-par with state-of-the-art



Performance



FlowNet2 outperforms any other method in terms of accuracy vs. speed

Achieved accuracy on motion segmentation and action recognition is now comparable to other methods

	Motion Seg. F-Measure	Motion Seg. Extracted Objects	Action Recog. Accuracy
LDOF-CPU [7]	79.51%	28/65	79.91%†
DeepFlow [32]	**80.18%**	29/65	**81.89%**
EpicFlow [22]	78.36%	27/65	78.90%
FlowFields [2]	79.70%	**30/65**	–
FlowNetS [11]	56.87%‡	3/62‡	55.27%
FlowNet2-css-ft-sd	77.88%	26/65	–
FlowNet2-CSS-ft-sd	79.52%	30/65	**79.64%**
FlowNet2	**79.92%**	**32/65**	79.51%

Acknowledgements

Partially funded by the ERC Starting Grant VideoLearn, the DFG Grant BR 3815/7-1 and the EU Horizon2020 project TrimBot2020

Flow Results