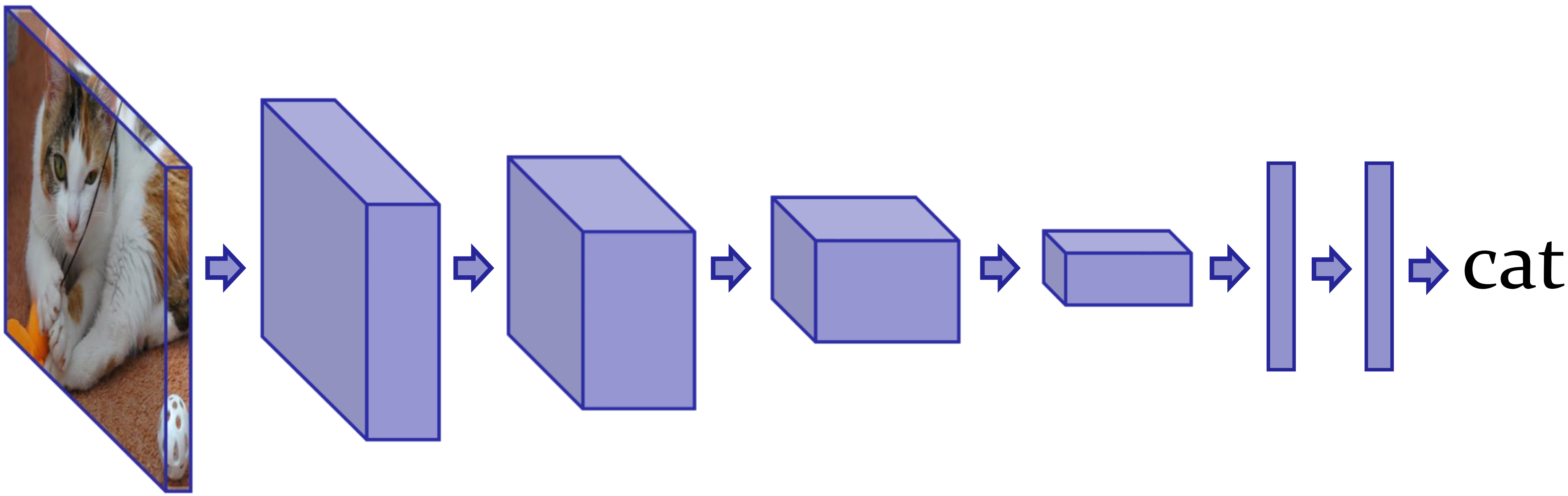
Up-convolutional networks and their applications

Alexey Dosovitskiy

University of Freiburg / Intel Labs

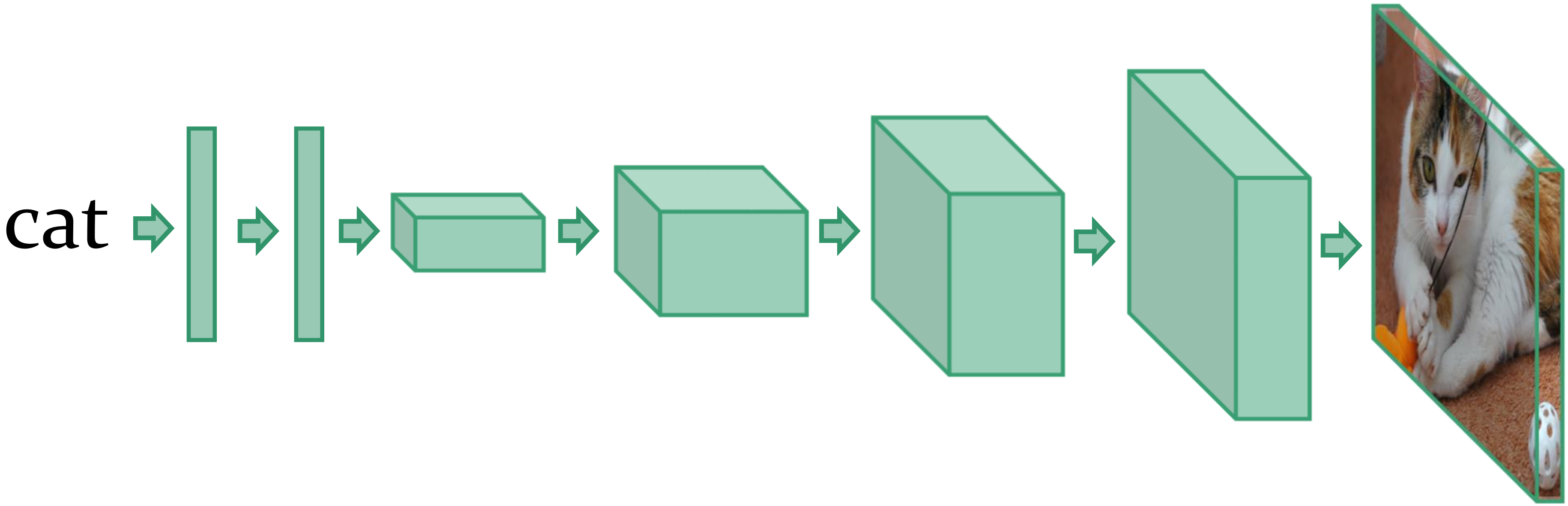
28.12.2016

Convolutional network



Convolutional network

Up-convolutional network



Up-convolutional network
(a.k.a. "deconvolutional")

This talk

- Up-convolutional networks
- End-to-end estimation of motion and depth
- Inverting ConvNets with perceptual metrics
- Visualizing neurons and generating images

- Learning to play Doom

Pooling and unpooling

- Pooling = shrinking the feature maps

Convolution with stride

2	1.5	0.8	1.2
1	1.7	1.5	1.7
0.9	1.3	2.3	1.6
1.4	0.5	2.1	1.9

2	0.8
0.9	2.3

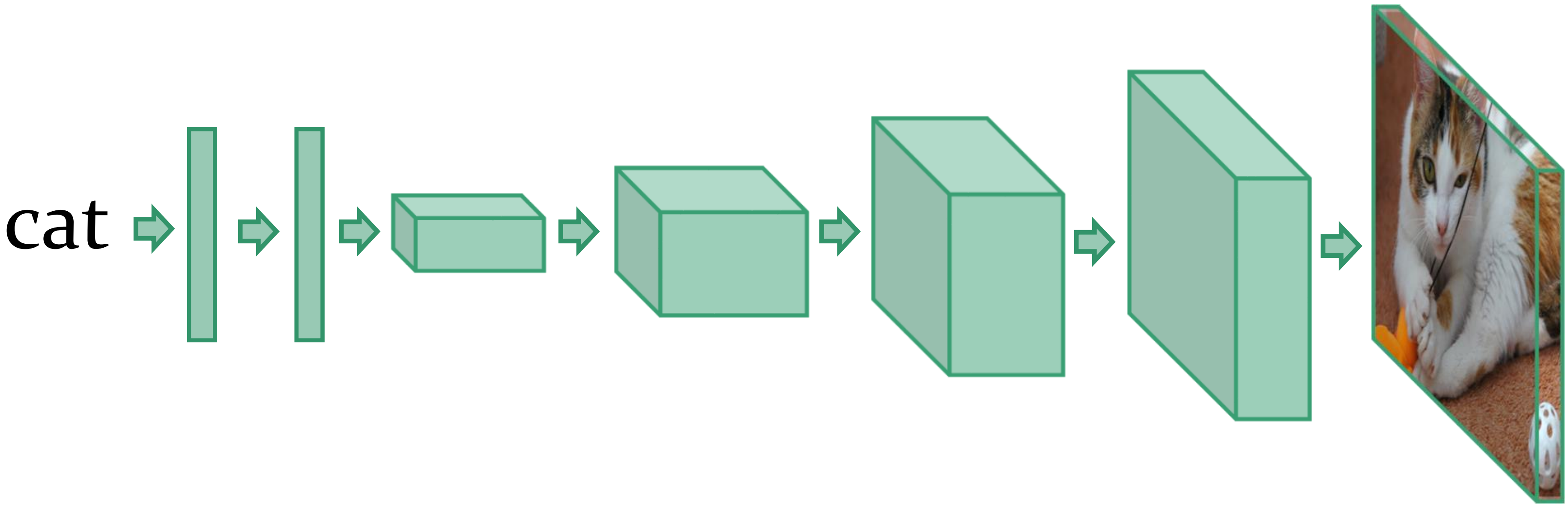
- Unpooling = expanding the feature maps (upscaling)

“Bed of nails” upsampling

2	0.8
0.9	2.3

2	0	**0.8**	0
0	0	0	0
0.9	0	**2.3**	0
0	0	0	0

Up-convolutional network



Up-convolutional network
(a.k.a. "deconvolutional")

What can we do with this thing?

End-to-end estimation of motion and depth

Joint work with the group of Daniel Cremers

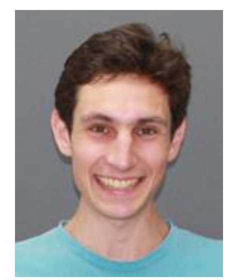
Philipp
Fischer

Eddy
Ilg

Nikolaus
Mayer

Philip
Häusser

Caner
Hazirbas



Vladimir
Golkov

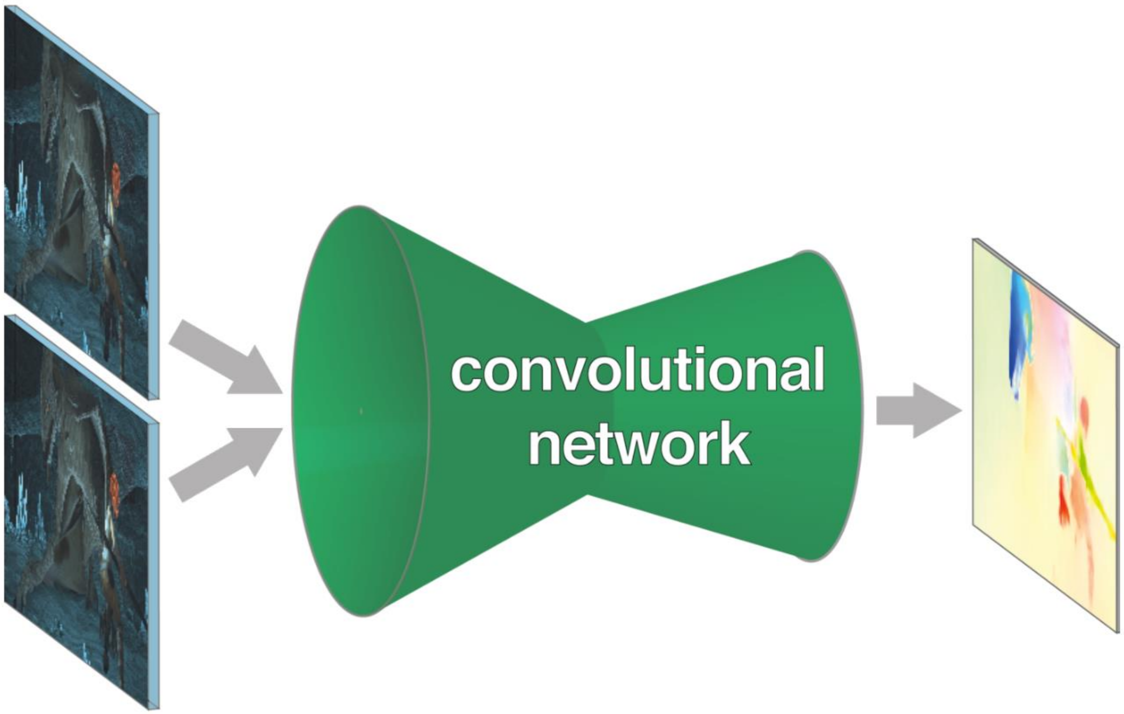
ICCV 2015, CVPR 2016, arxiv 2016

Optical flow estimation is difficult!

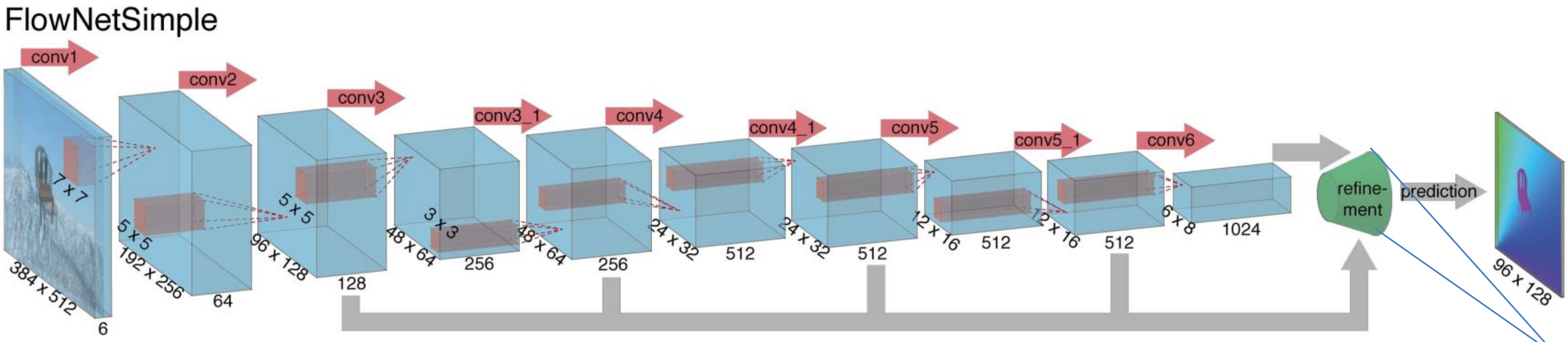
Optical flow estimation is difficult!

End-to-end optical flow

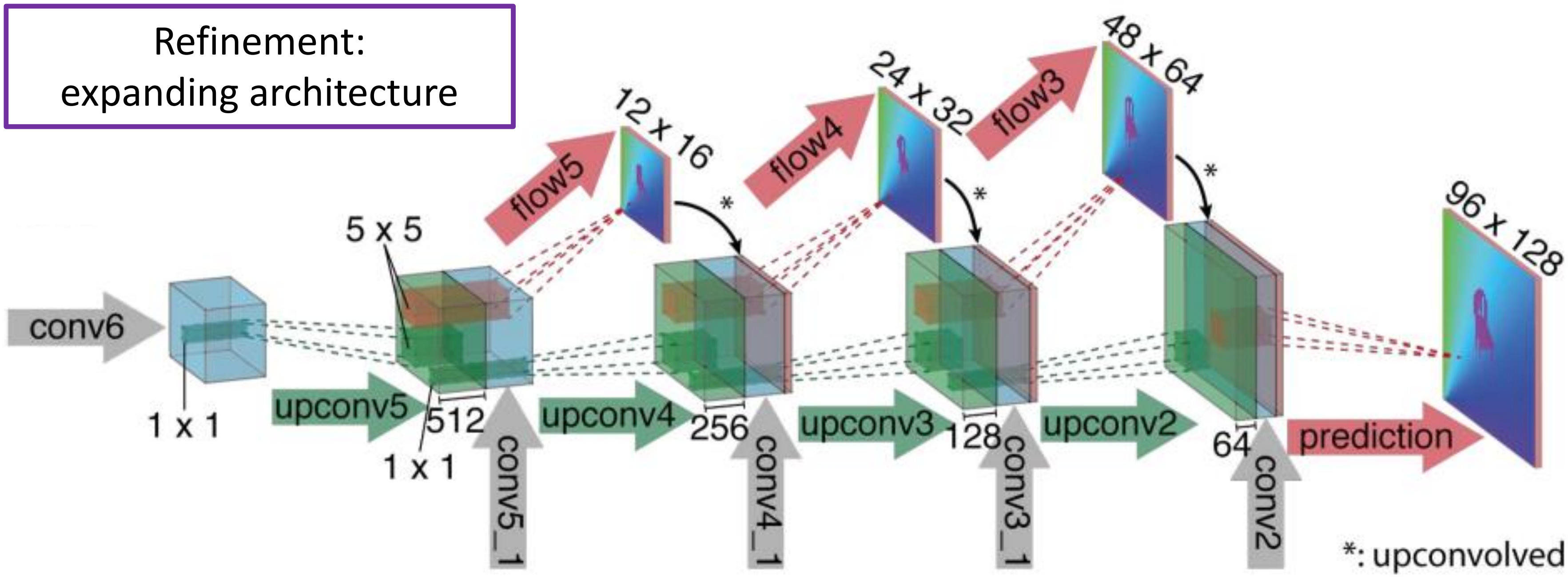
- Standard approach: matching + aggregation
- End to end: two frames in, flow out



Architecture: FlowNetSimple

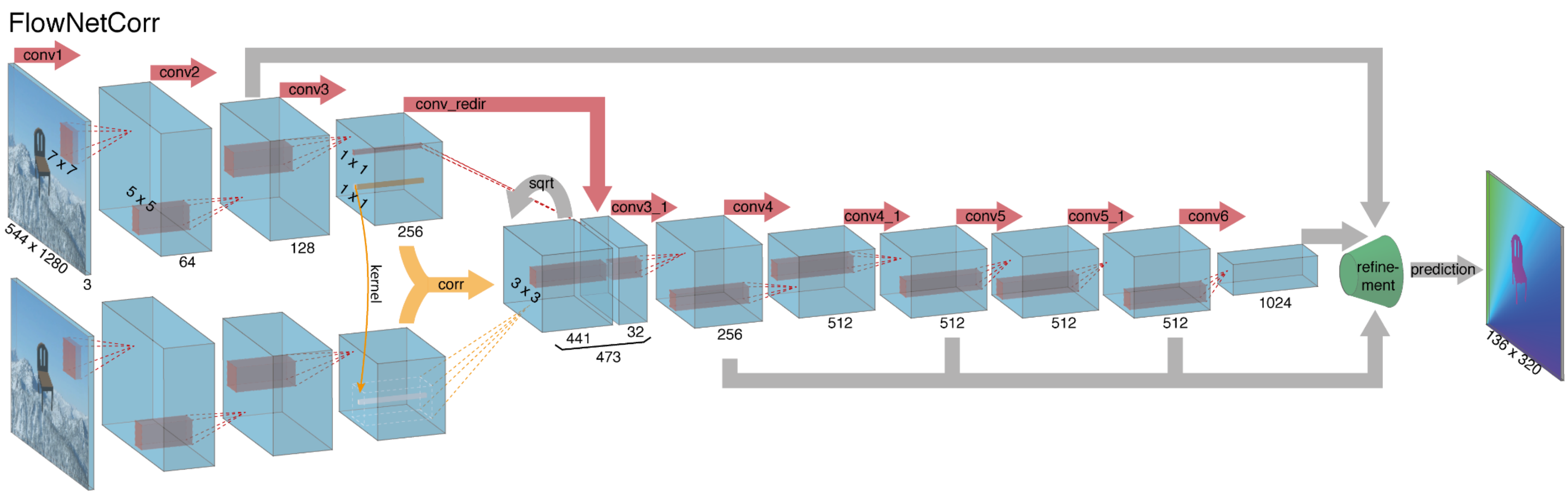


Refinement:
expanding architecture



[o] Vision COMPUTER VISION University of Freiburg

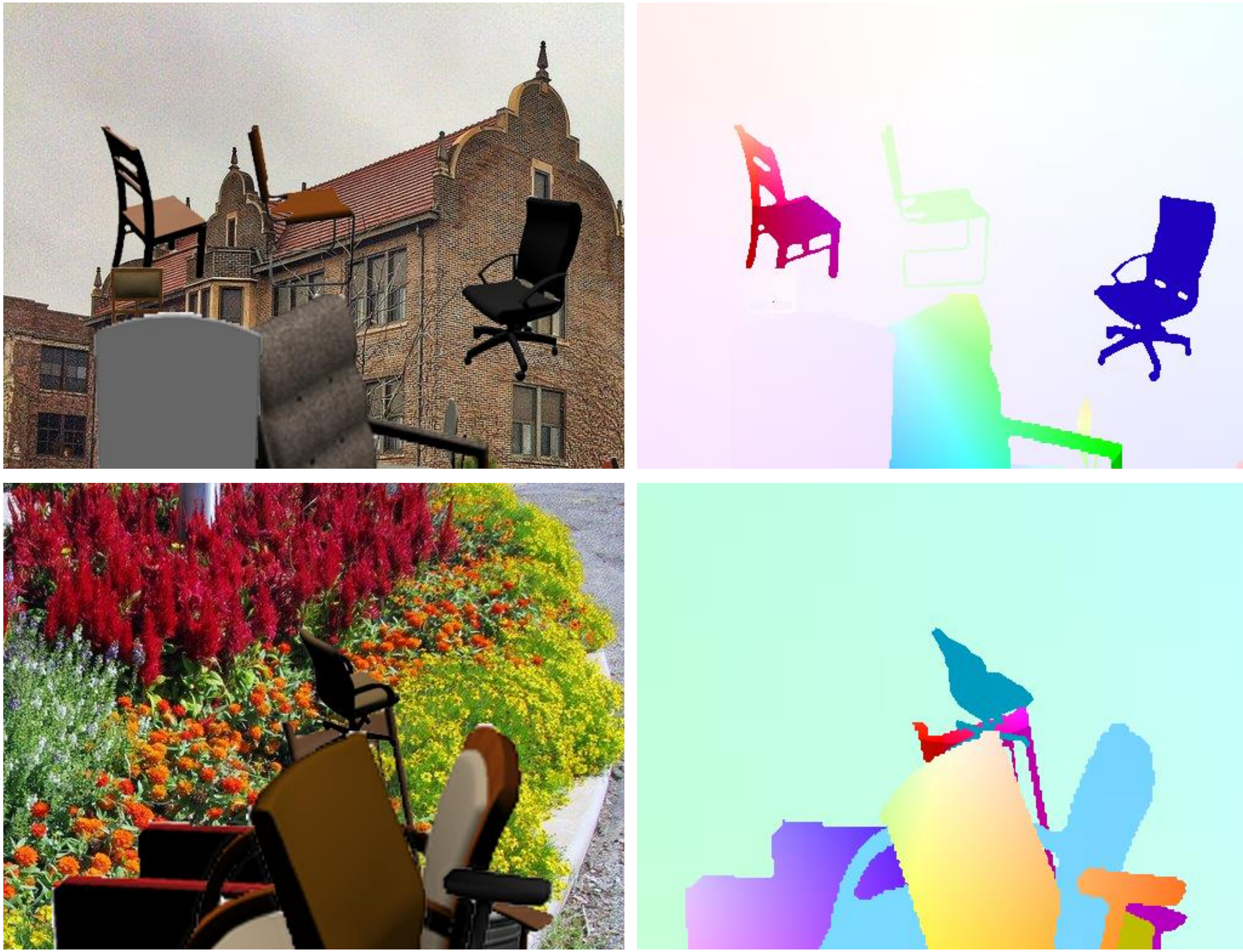
Architecture: FlowNetCorr



How to train?

- Of course, supervised!
- Where do we get training data?

The “Flying chairs” dataset



Rendered image   Optical flow

It works on “Flying chairs” !

Input images

EpicFlow (Revaud et al. 2015)

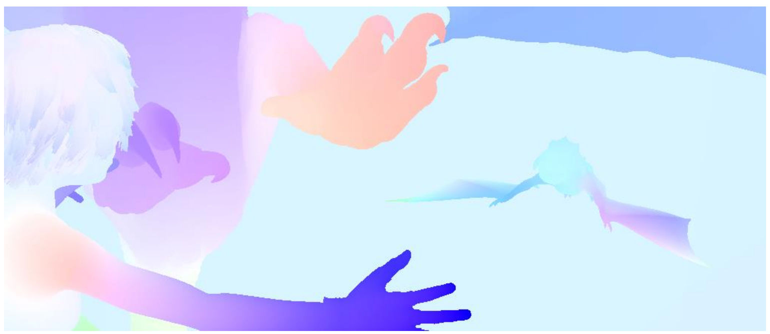
Ground truth

FlowNetCorr

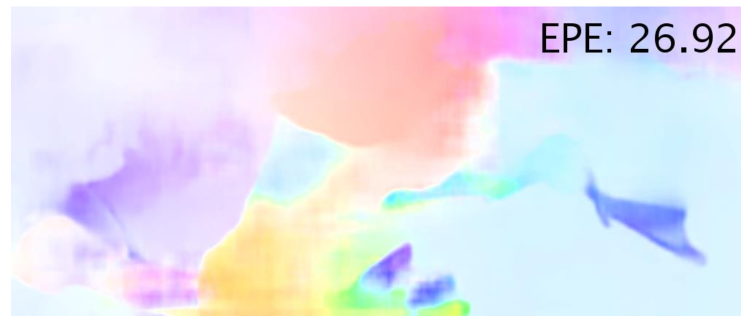
And it generalizes!

Input images

FlowNetSimple



Ground truth

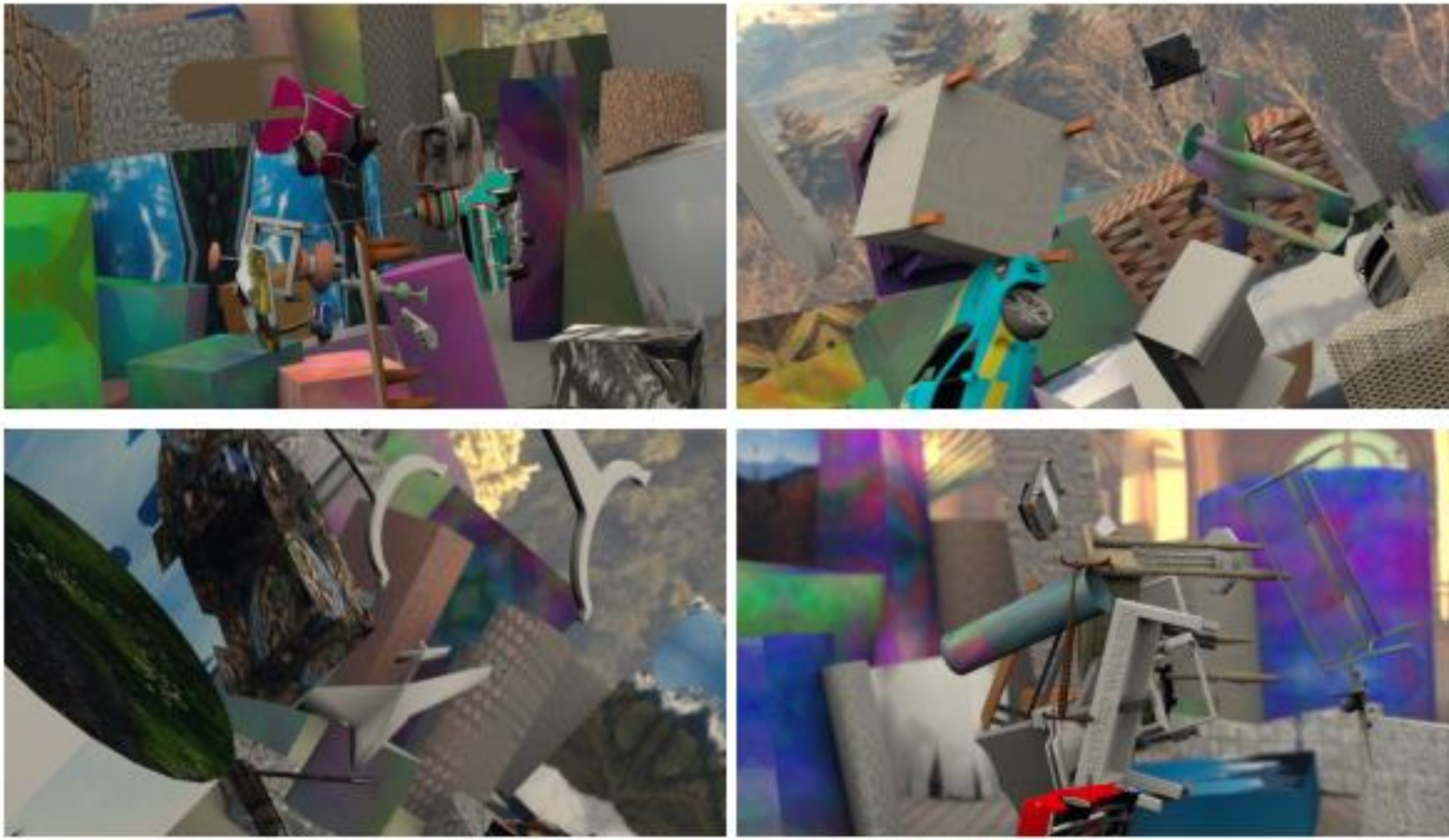


FlowNetCorr

LDOF (Brox-Malik 2011)

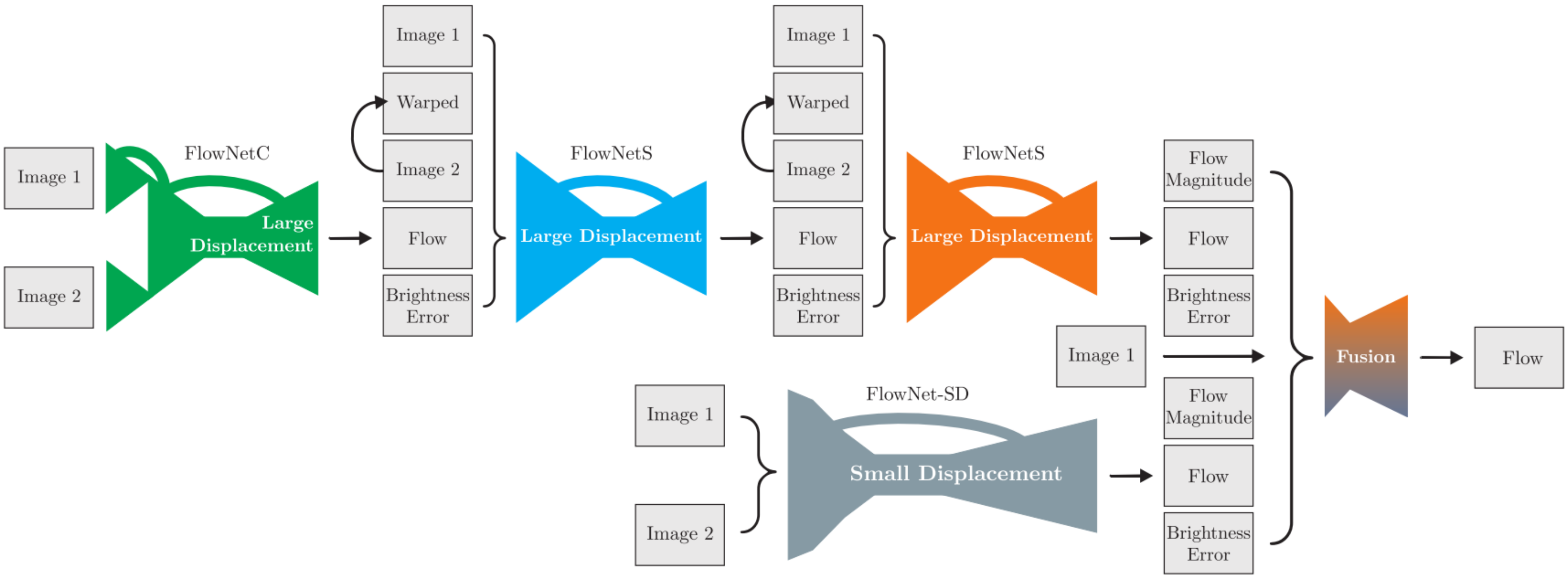
EpicFlow (Revaud et al. 2015)

Better dataset: FlyingThings 3D

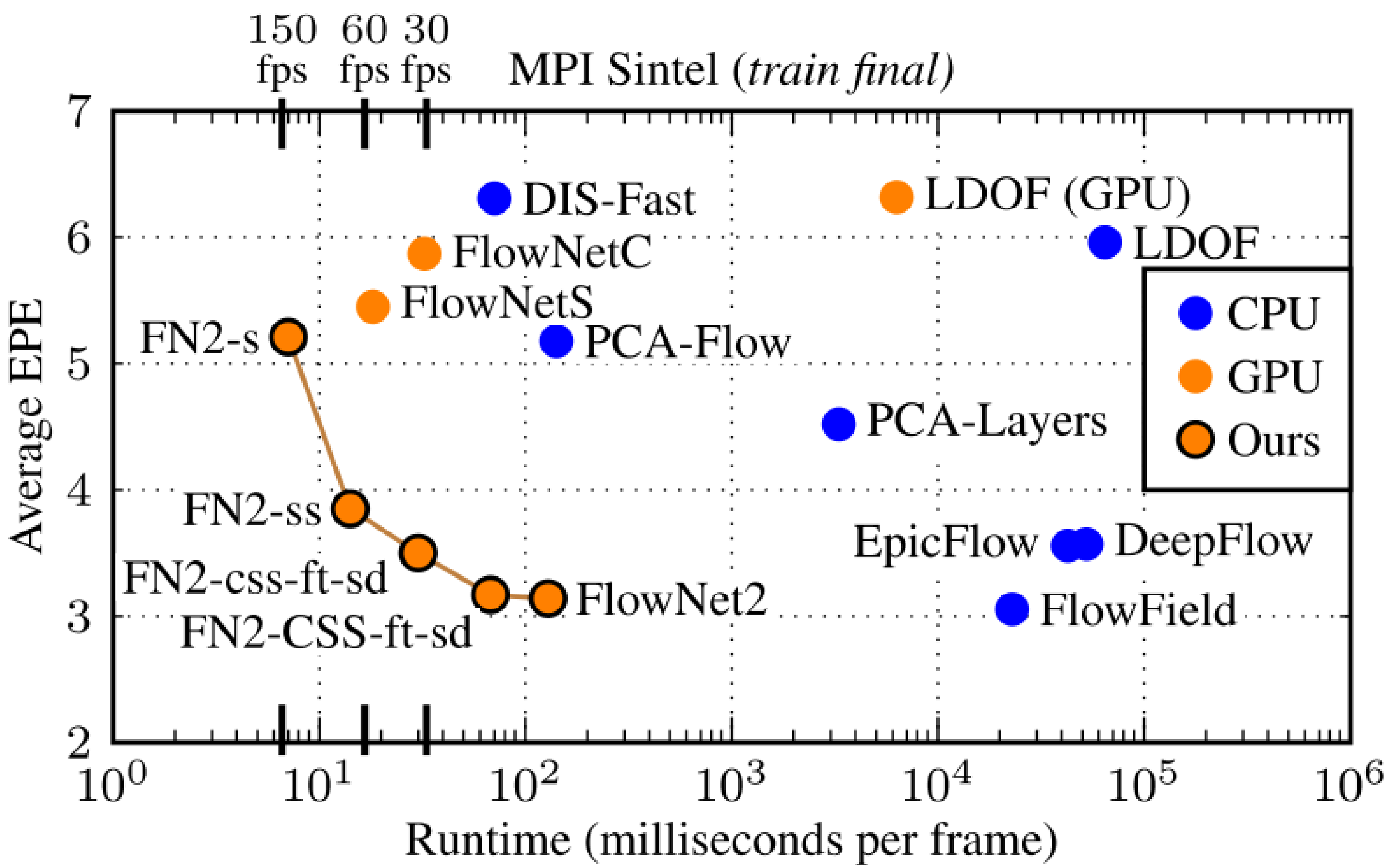


Better architecture: FlowNet 2.0

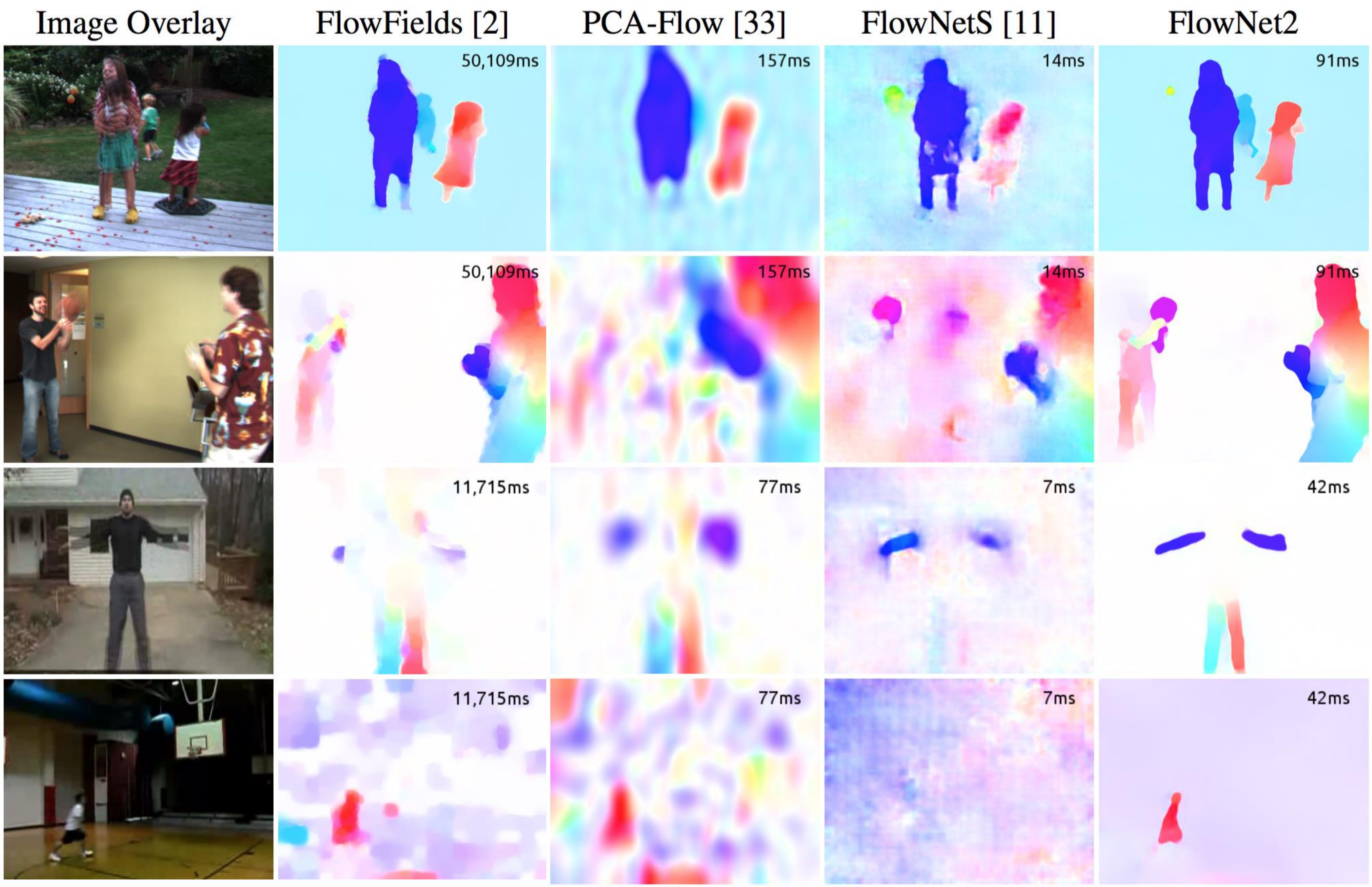
- Stacking and warping



FlowNet 2.0: Sintel



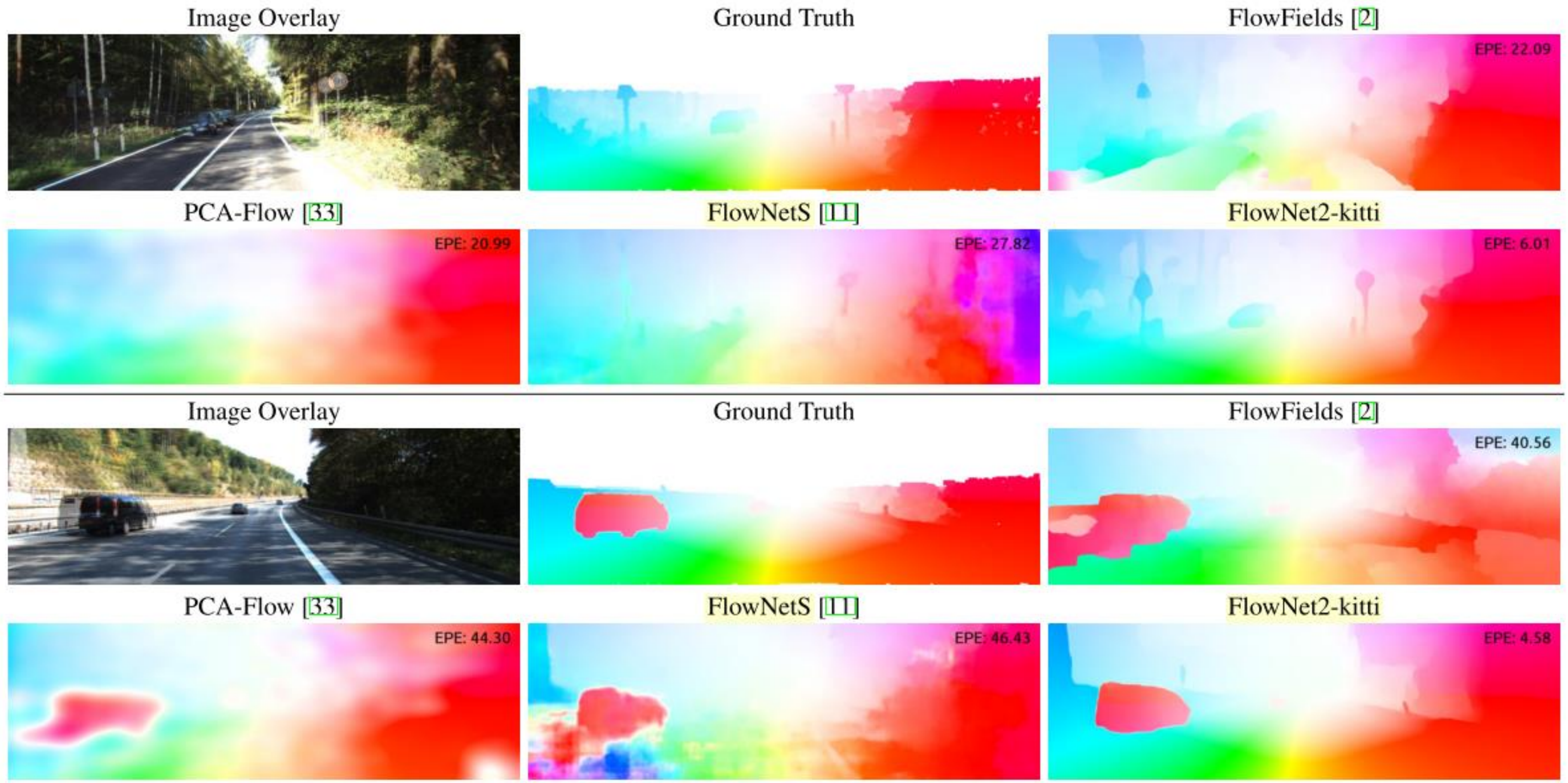
FlowNet 2.0: Real data



0.02-0.1 FPS

10-20 FPS

FlowNet 2.0: KITTI



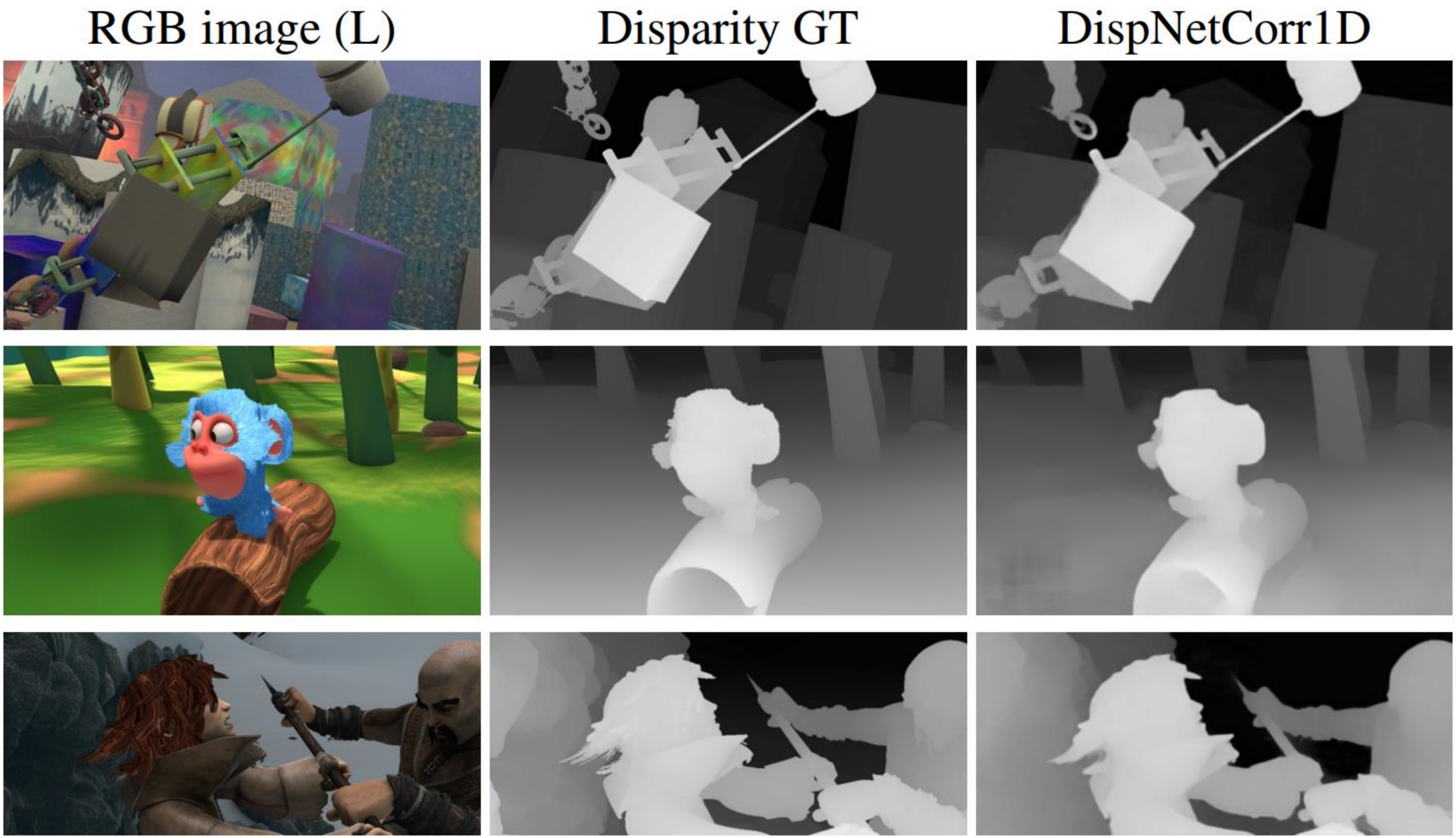
FlowNet 2.0: KITTI

	Method	Setting	Code	Fl-bg	Fl-fg	Fl-all	Density	Runtime
1	PRSM		code	**5.33 %**	17.02 %	**7.28 %**	100.00 %	300 s
C. Vogel, K. Schindler and S. Roth: 3D Scene Flow Estimation with a Piecewise Rigid Scene Model. ijcv 2015.								
2	OSF+TC			5.76 %	16.61 %	7.57 %	100.00 %	50 min
3	OSF		code	5.62 %	22.17 %	8.37 %	100.00 %	50 min
M. Menze and A. Geiger: Object Scene Flow for Autonomous Vehicles. Conference on Computer Vision and Pattern Recognition								
4	SSFAV			7.10 %	21.22 %	9.45 %	100.00 %	5 min
5	FlowNet2			10.75 %	**15.14 %**	11.48 %	100.00 %	0.12 s
6	SDF			8.61 %	26.69 %	11.62 %	100.00 %	TBA
M. Bai*, W. Luo*, K. Kundu and R. Urtasun: Exploiting Semantic Information and Deep Matching for Optical Flow. ECCV 2016.								
7	FSF+MS			8.48 %	29.62 %	12.00 %	100.00 %	2.7 s
8	CNNF+PMBP			10.08 %	23.18 %	12.26 %	100.00 %	45 min
9	CSF			10.40 %	30.33 %	13.71 %	100.00 %	80 s
Z. Lv, C. Beall, P. Alcantarilla, F. Li, Z. Kira and F. Dellaert: A Continuous Optimization Approach for Efficient and Accurate Sce								

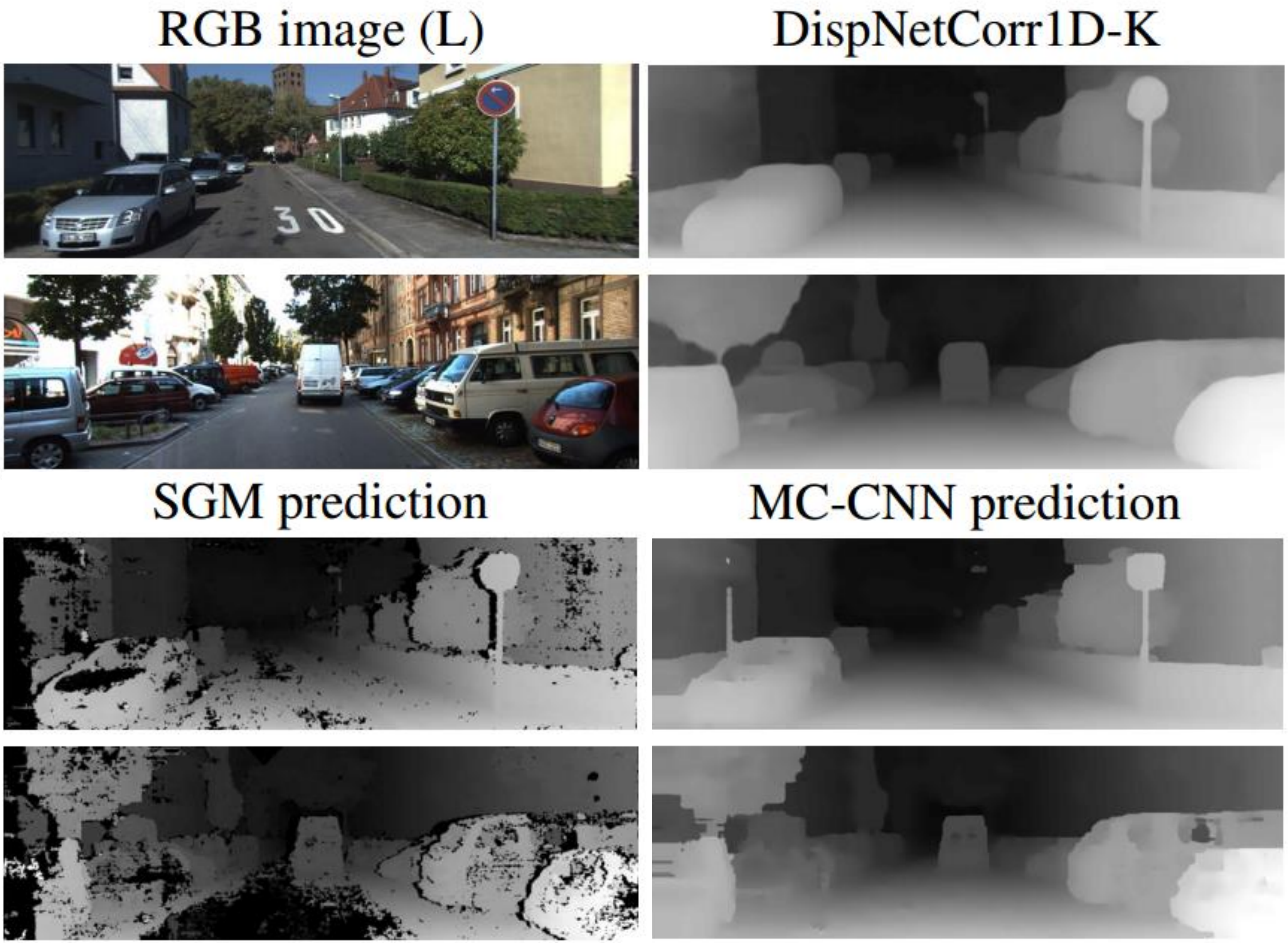
What next?

Depth estimation!

Disparity estimation

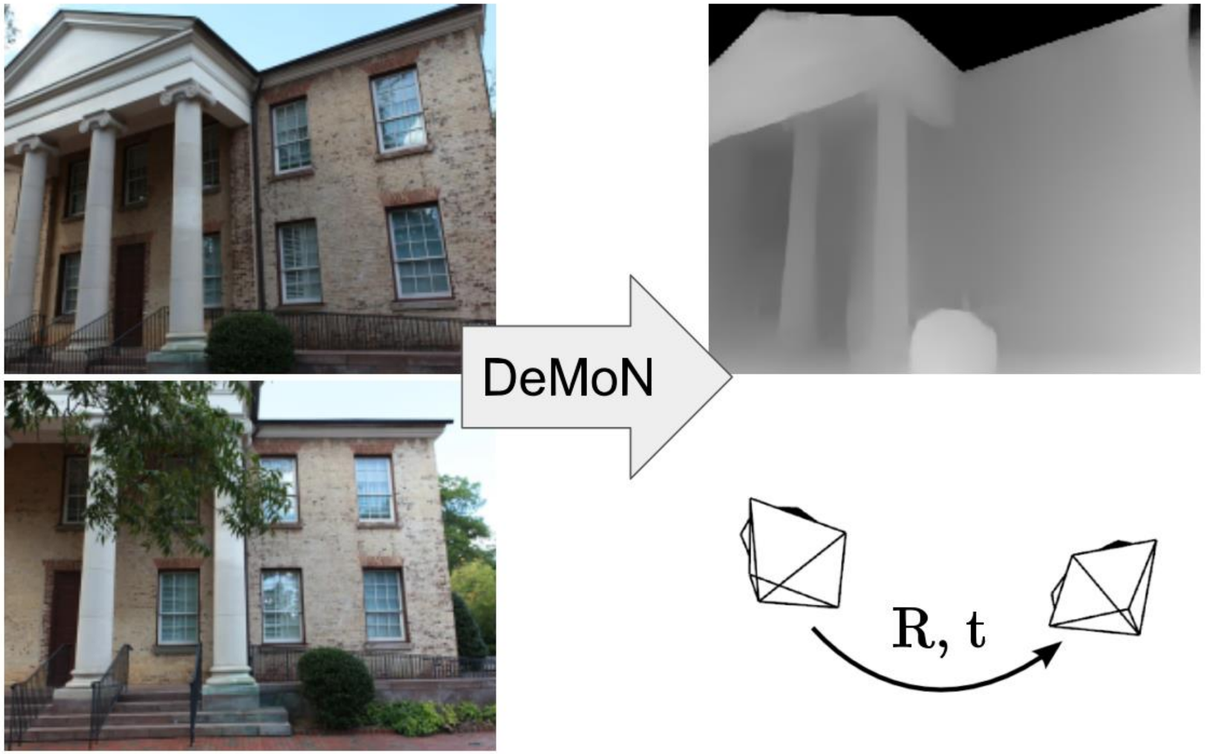


Disparity: KITTI



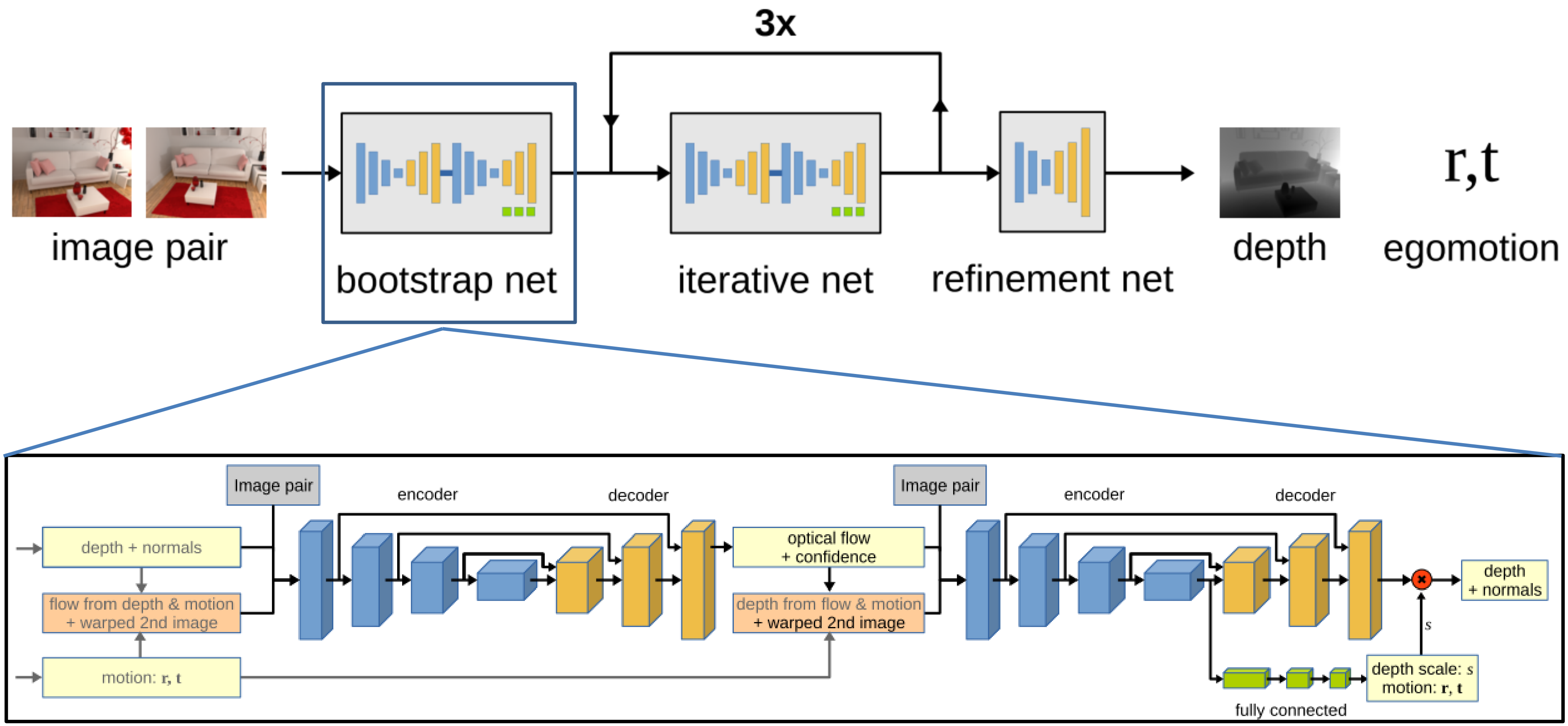
DeMoN: monocular stereo

- **Depth and Motion Network**
  - Two frames in
  - Depth and camera motion out

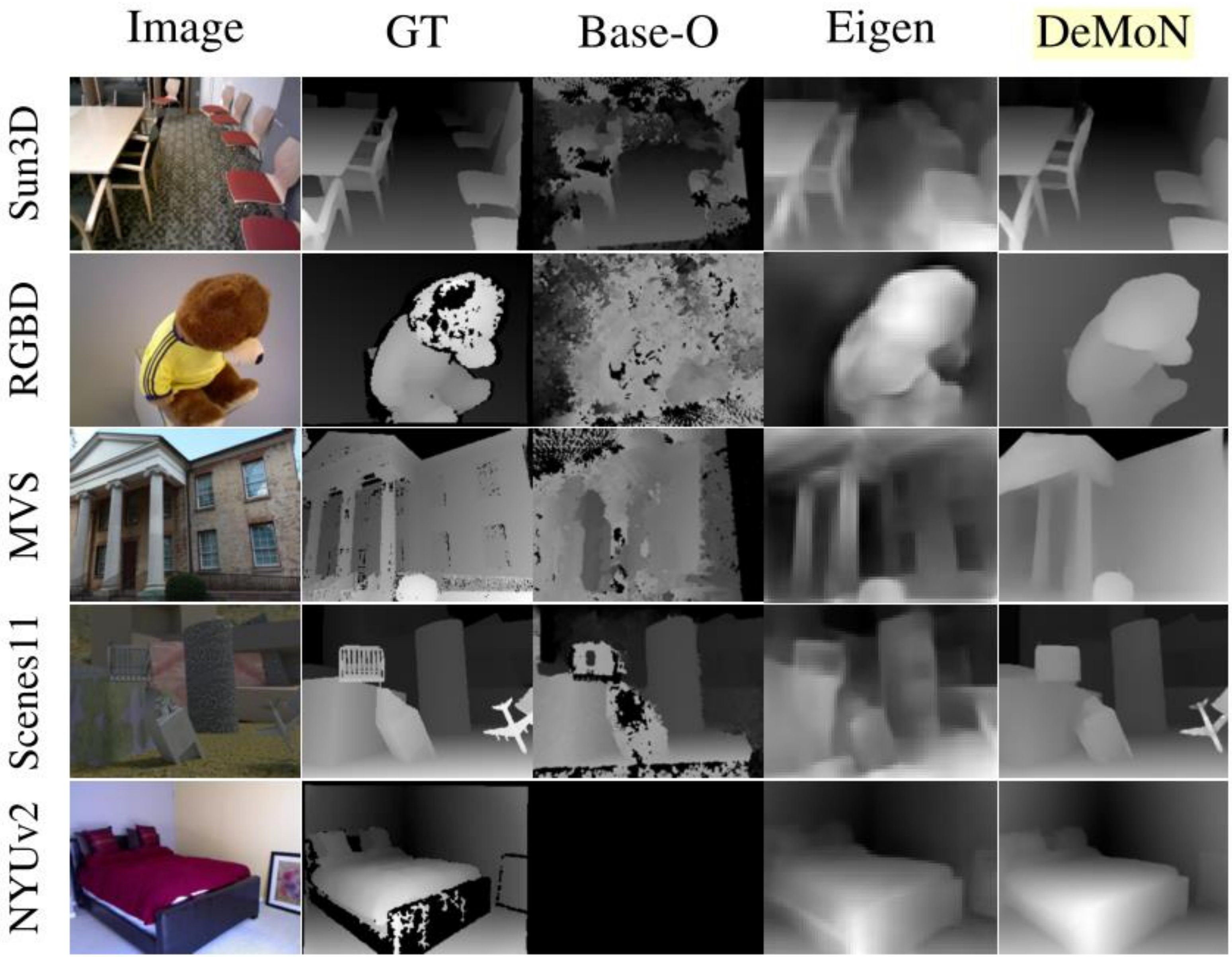


B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, T. Brox, arxiv 2016

DeMoN: architecture



DeMoN: architecture



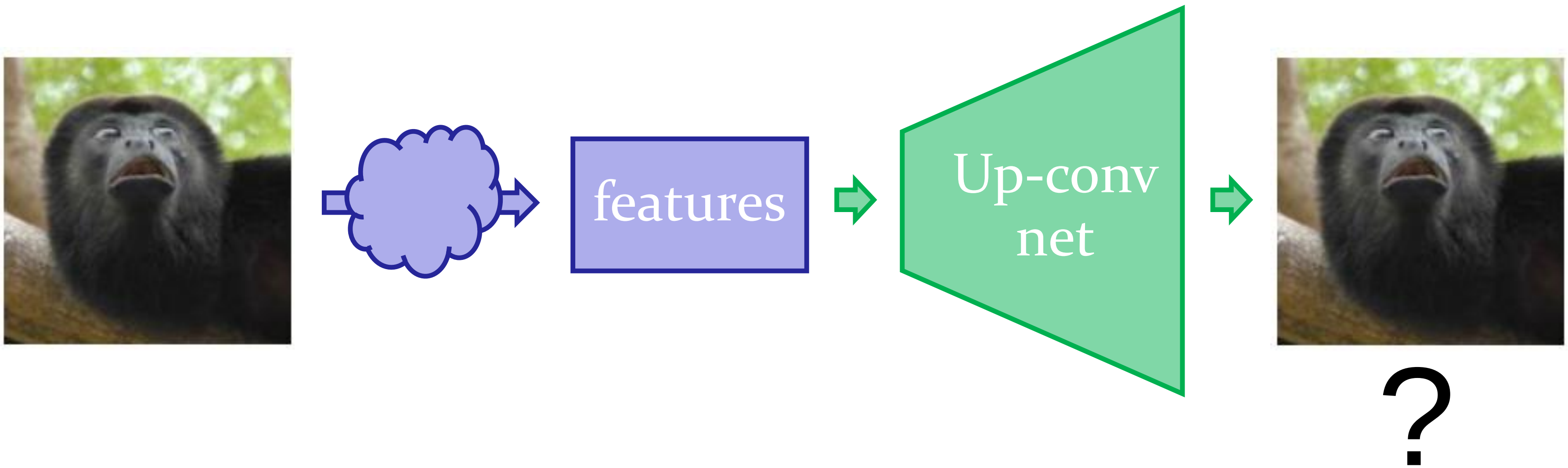
Summary

- (Up)ConvNets can estimate motion and depth end to end
- State-of-the-art performance at interactive framerates

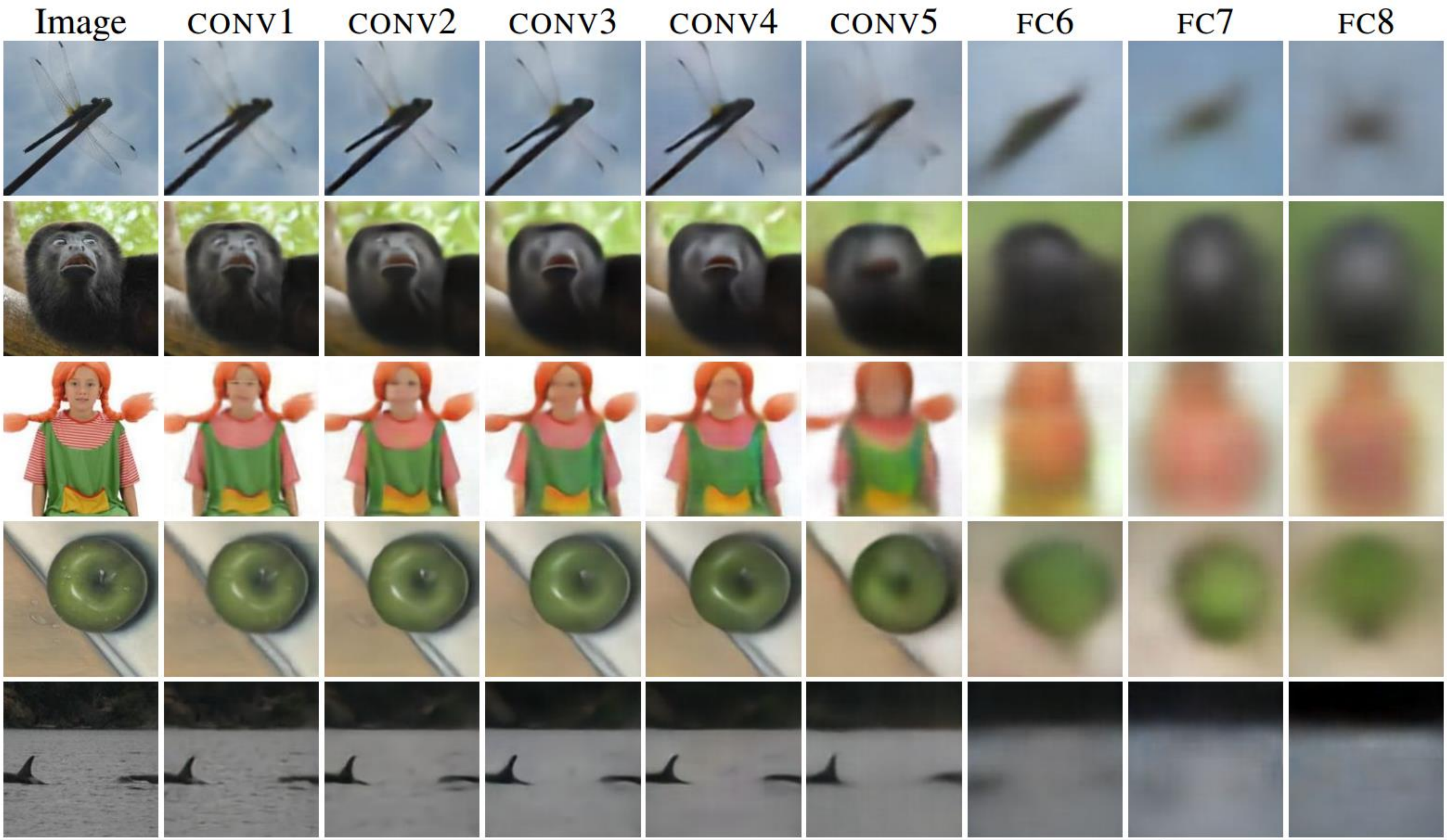
Inverting ConvNets with perceptual metrics

CVPR 2016, NIPS 2016

Inverting representations with ConvNets



Inverting AlexNet



Inverting AlexNet

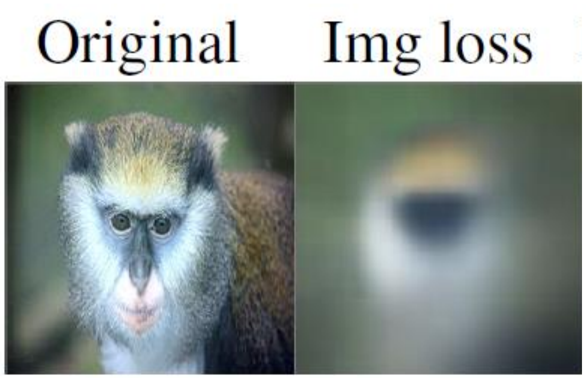
Why so blurry?

- Problem: the feature vector does not contain the precise locations of all details

- Solution: with appropriate loss function it need not!

Deep perceptual similarity metric

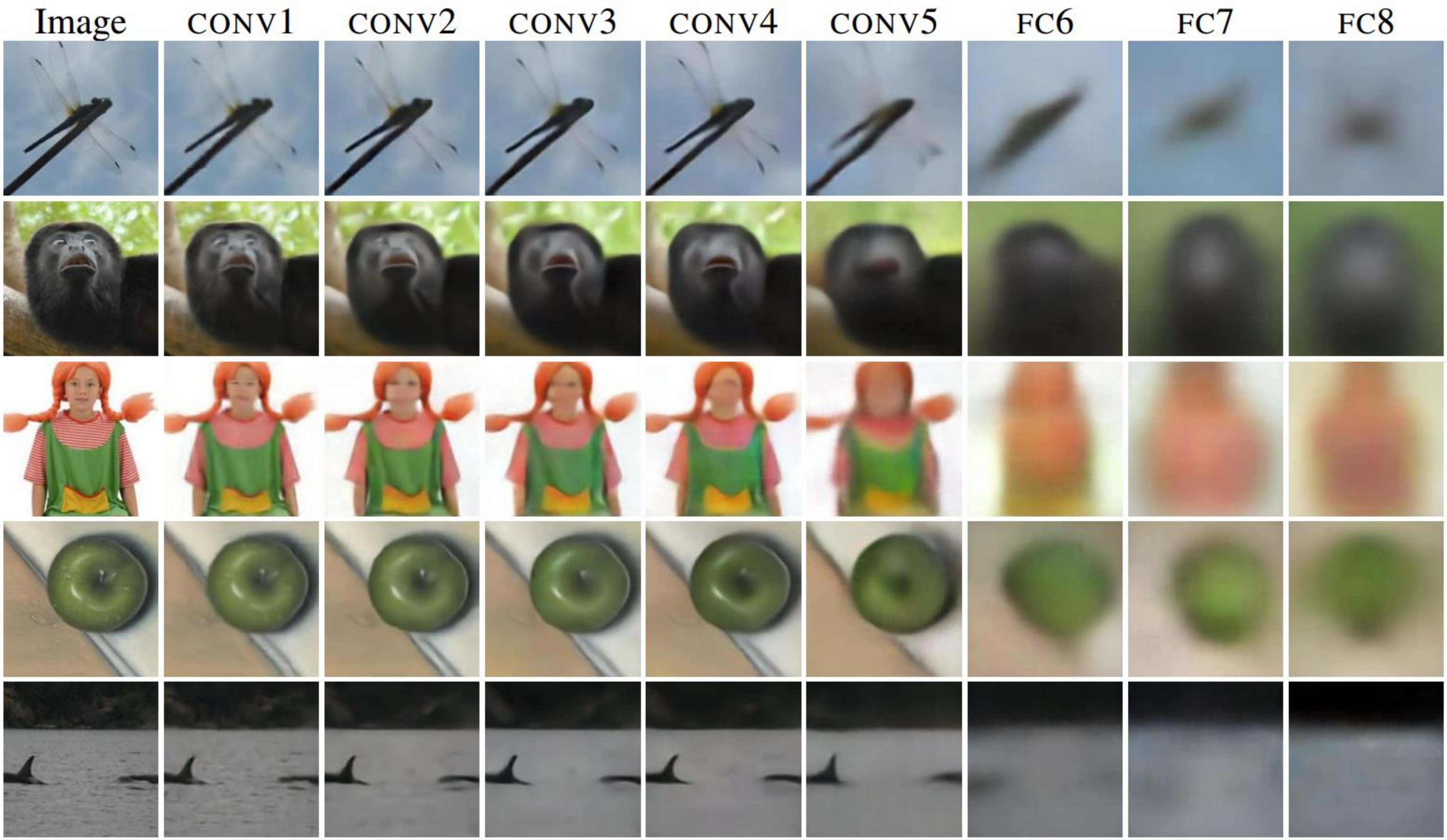
- Want to be sensitive to important properties, but invariant to irrelevant deformations
- Instead of the image space, measure image similarity in the feature space
- Add adversarial loss as a natural image prior



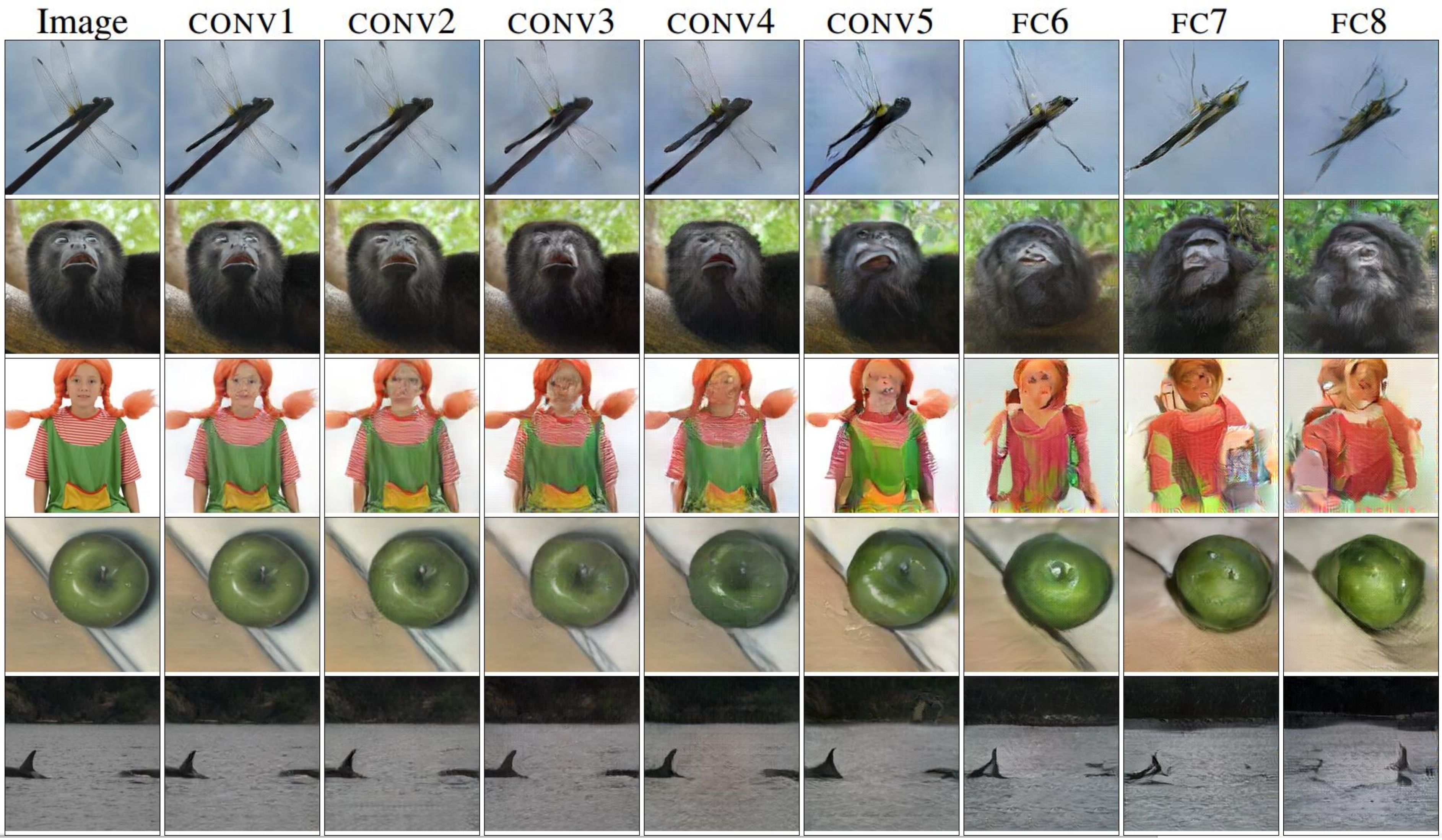
Deep perceptual similarity metric

- Want to be sensitive to important properties, but invariant to irrelevant deformations
- Instead of the image space, measure image similarity in the feature space
- Add adversarial loss as a natural image prior

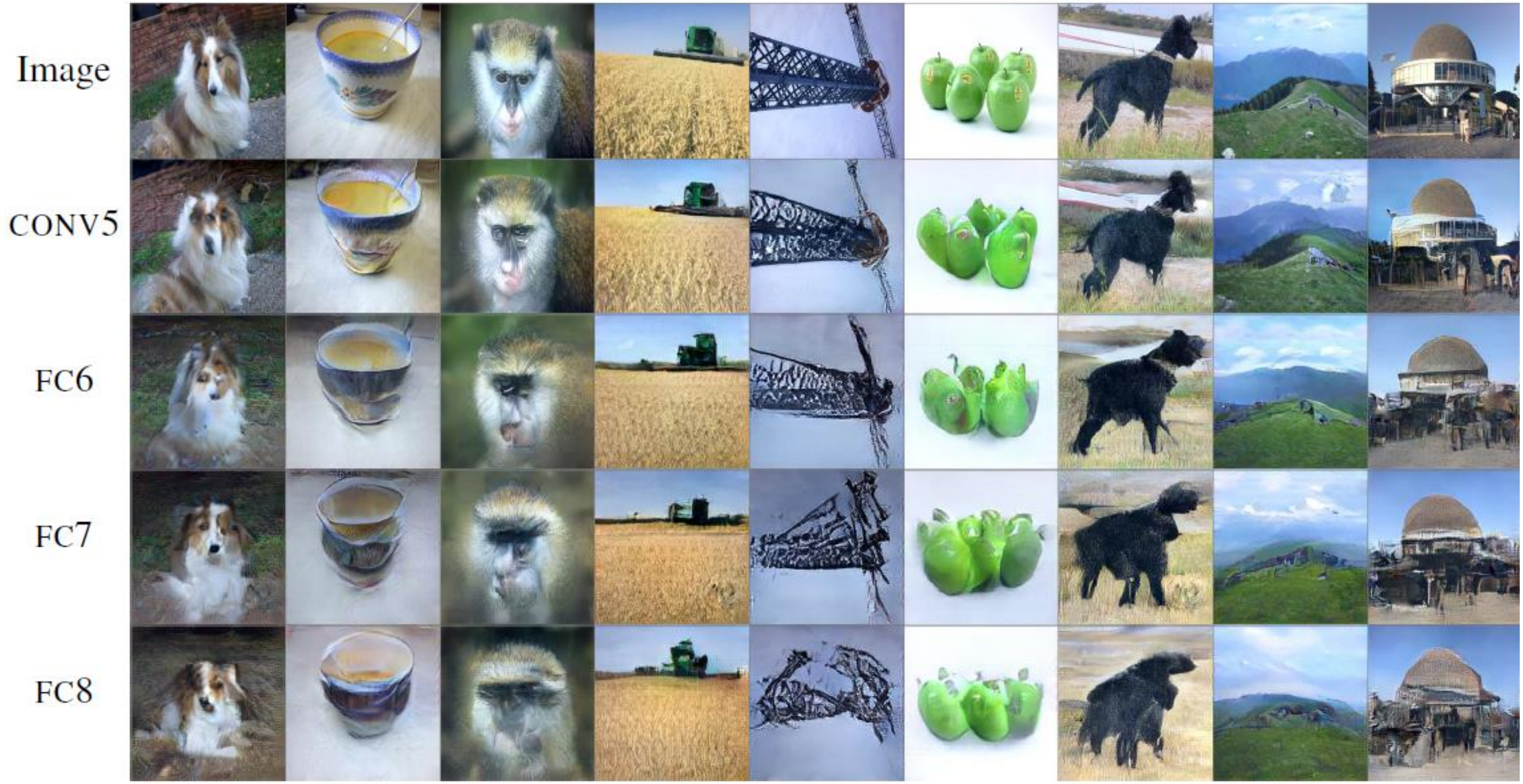
Inverting AlexNet: Euclidean loss



Inverting AlexNet: DeePSiM loss



Inverting AlexNet: more results



Practical applications

- Superresolution [Johnson et al. 2016], [Ledig et al. 2016]
- Image compression
- Denoising
- Analysis of deep networks
- Generative models

Visualizing neurons and generating images

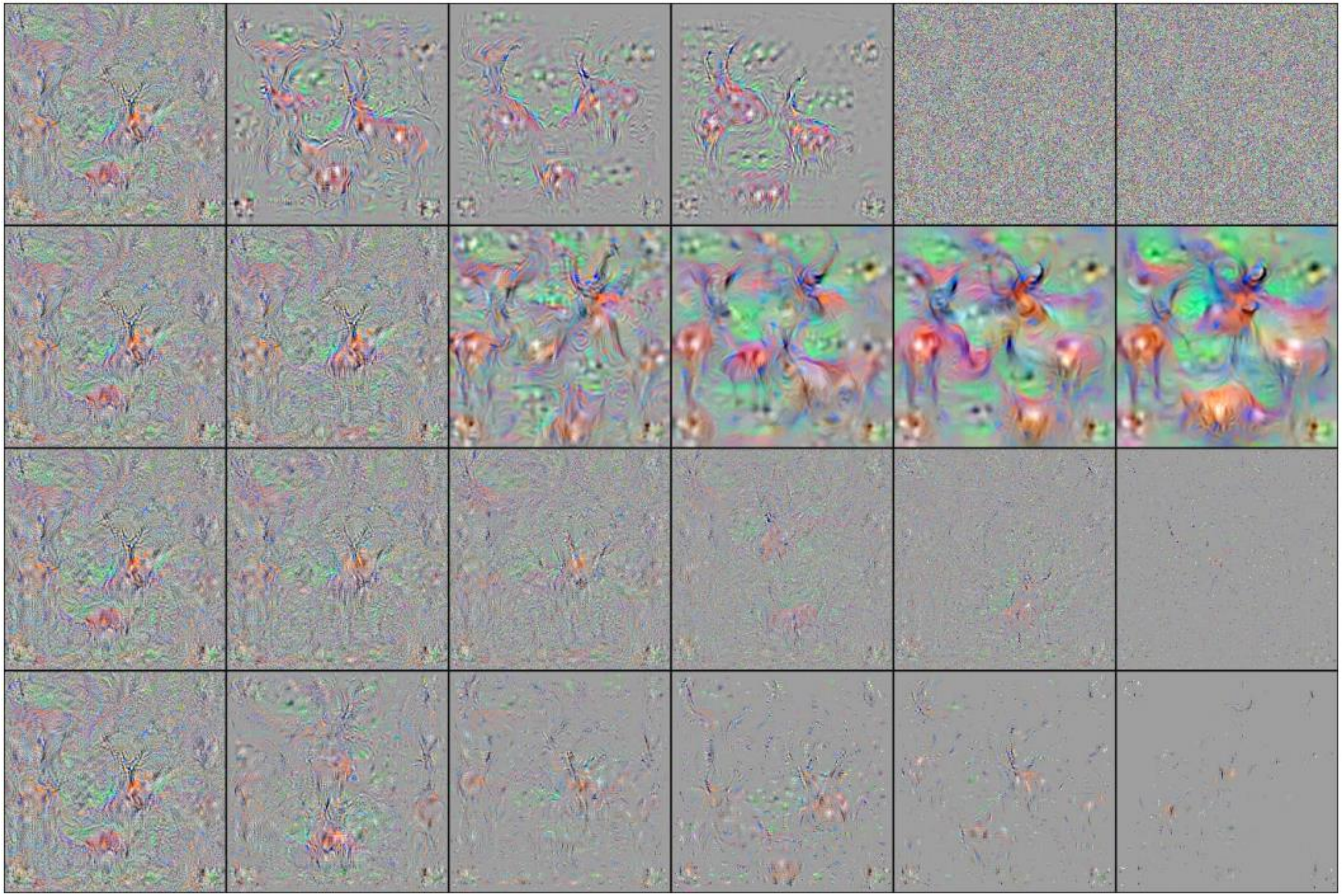
Anh Nguyen

Jason Yosinski

Jeff Clune

NIPS 2016, arxiv 2016

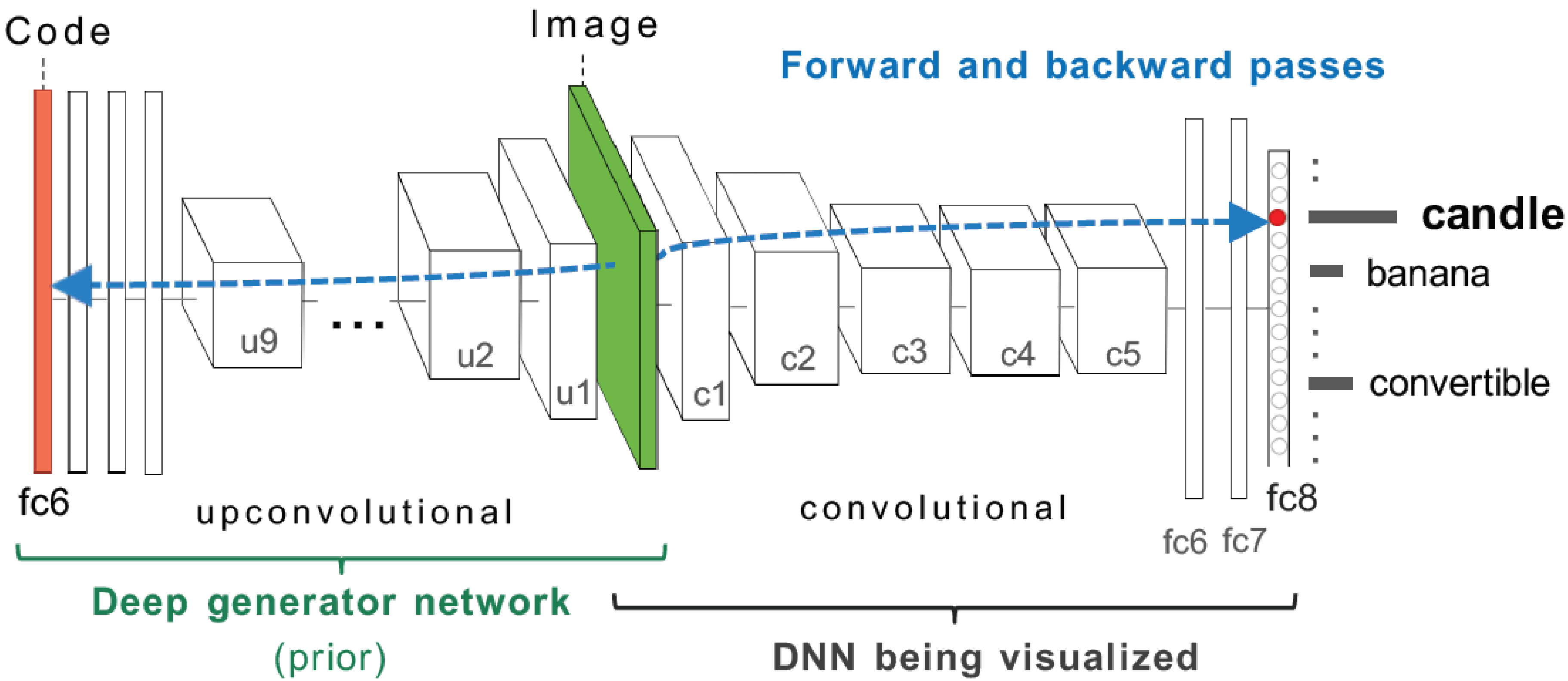
Activation maximization



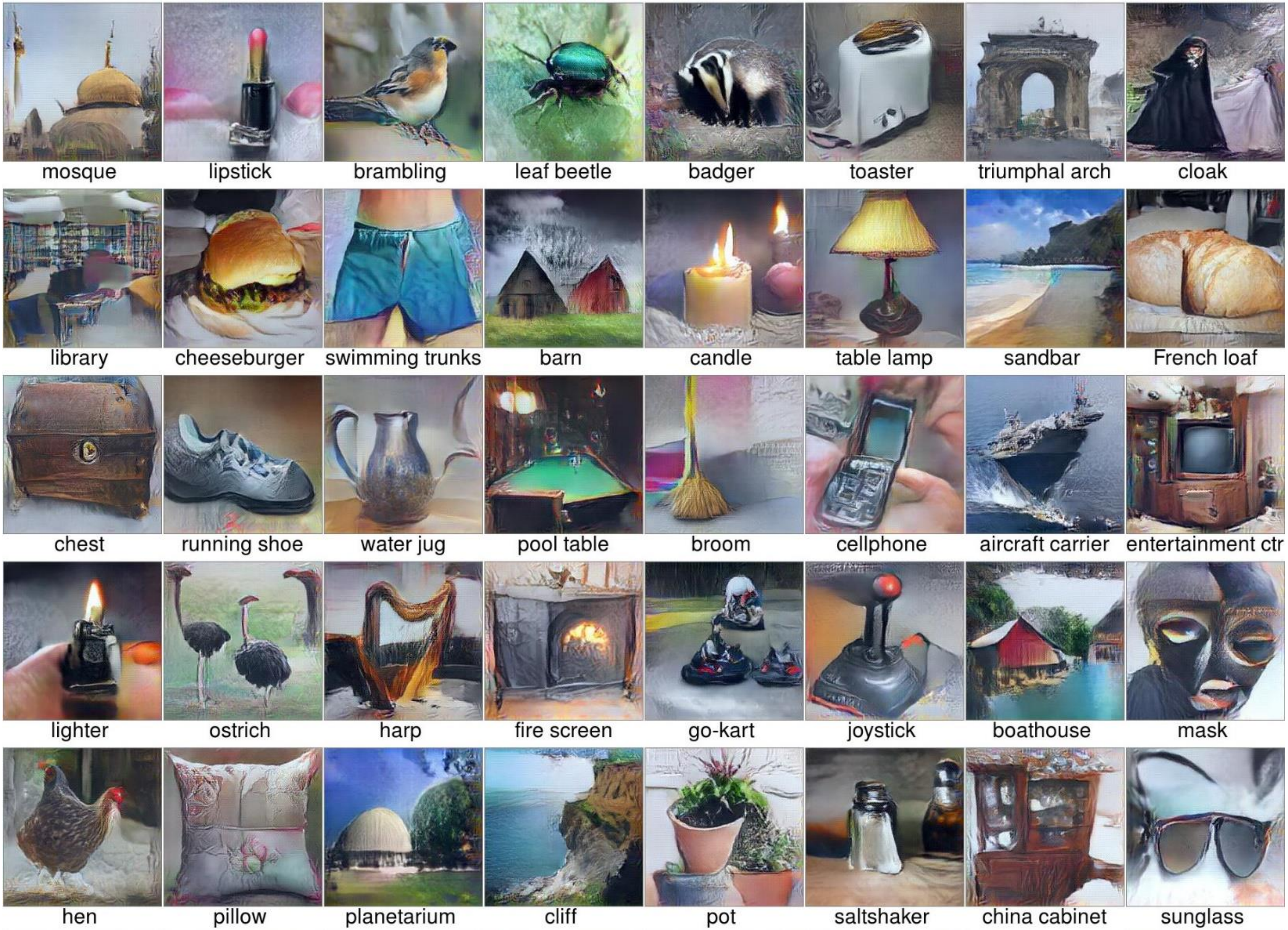
Yosinski et al. 2015

Hand-crafted priors are not good enough

AM with an UpConvNet prior

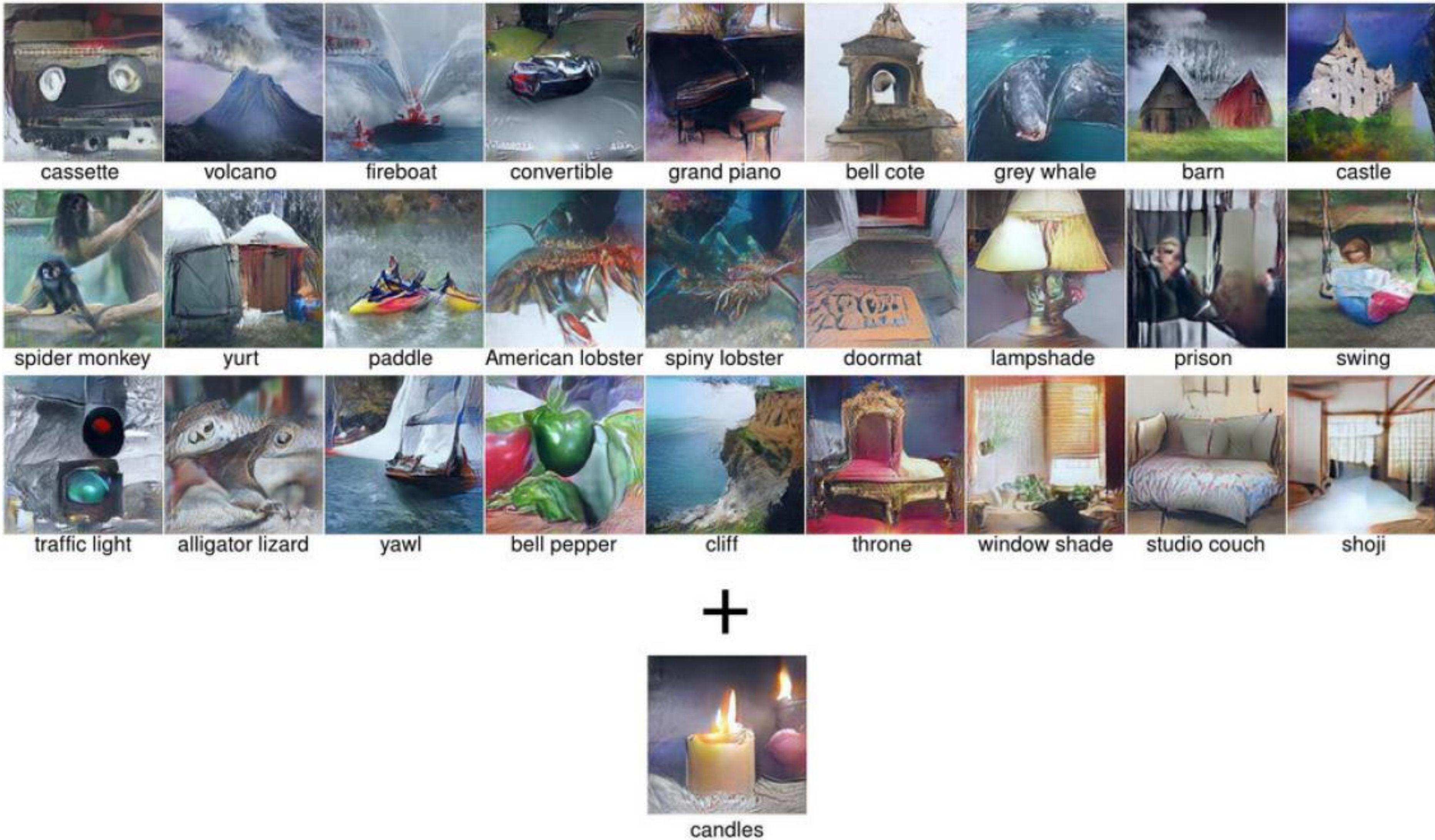


Activating FC8 neurons: ImageNet

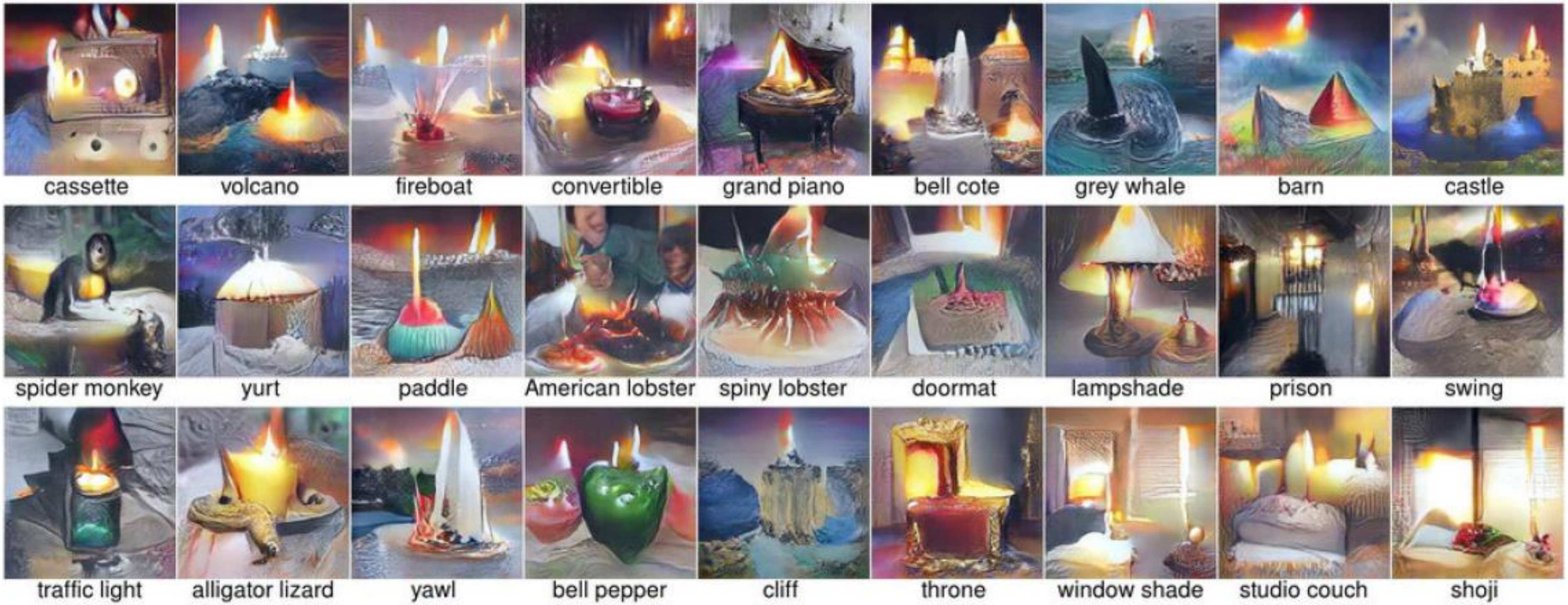


Activating FC8 neurons: Places

Activating FC8 neurons: 2 classes



Activating FC8 neurons: 2 classes



Activating neurons from different layers

Generative model?

- Pictures are nice, but is it a real generative model?
- If we add some noise during optimization, it is!

Plug-and-play generative networks

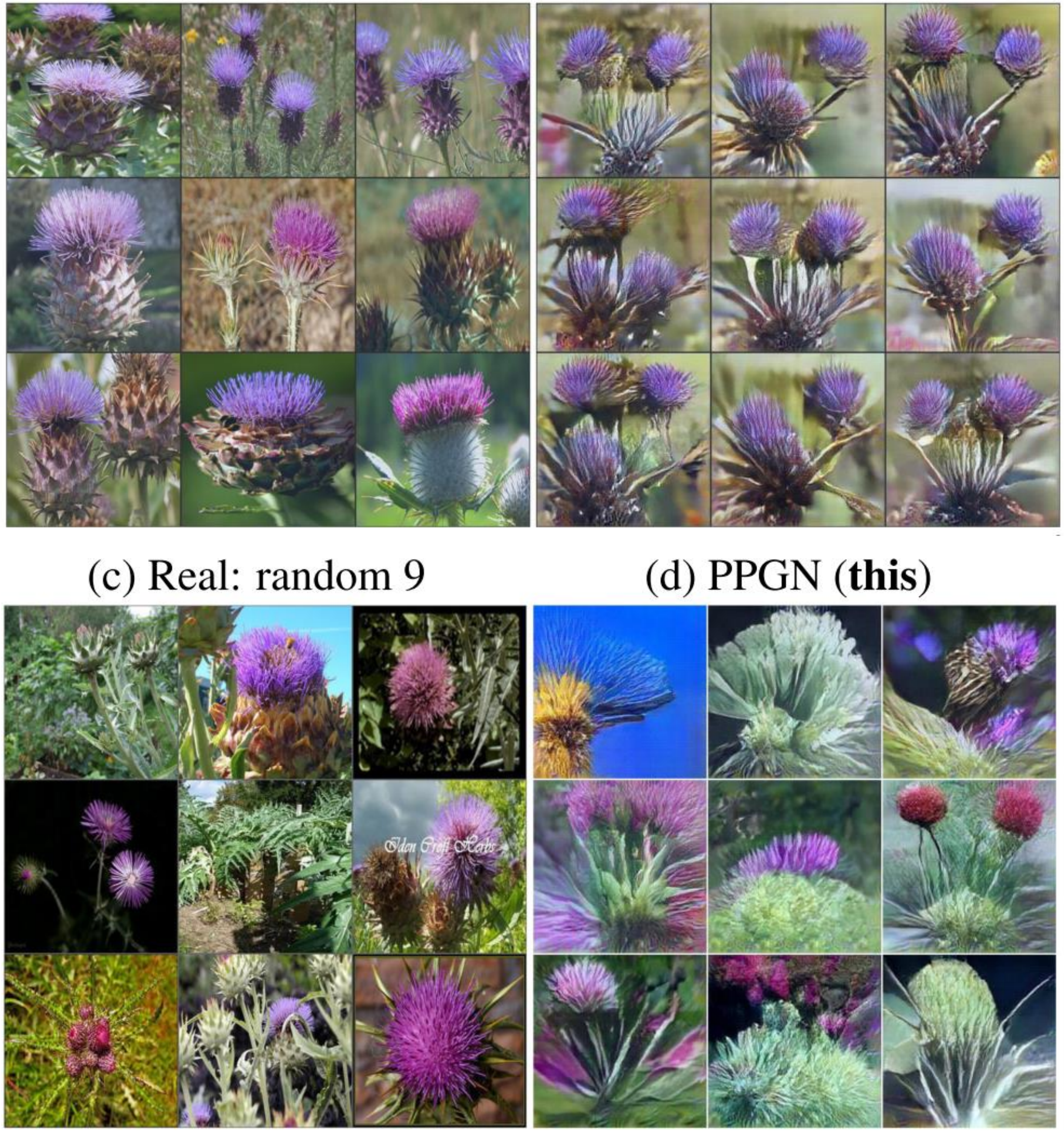
(a) Real: top 9

(c) Real: random 9

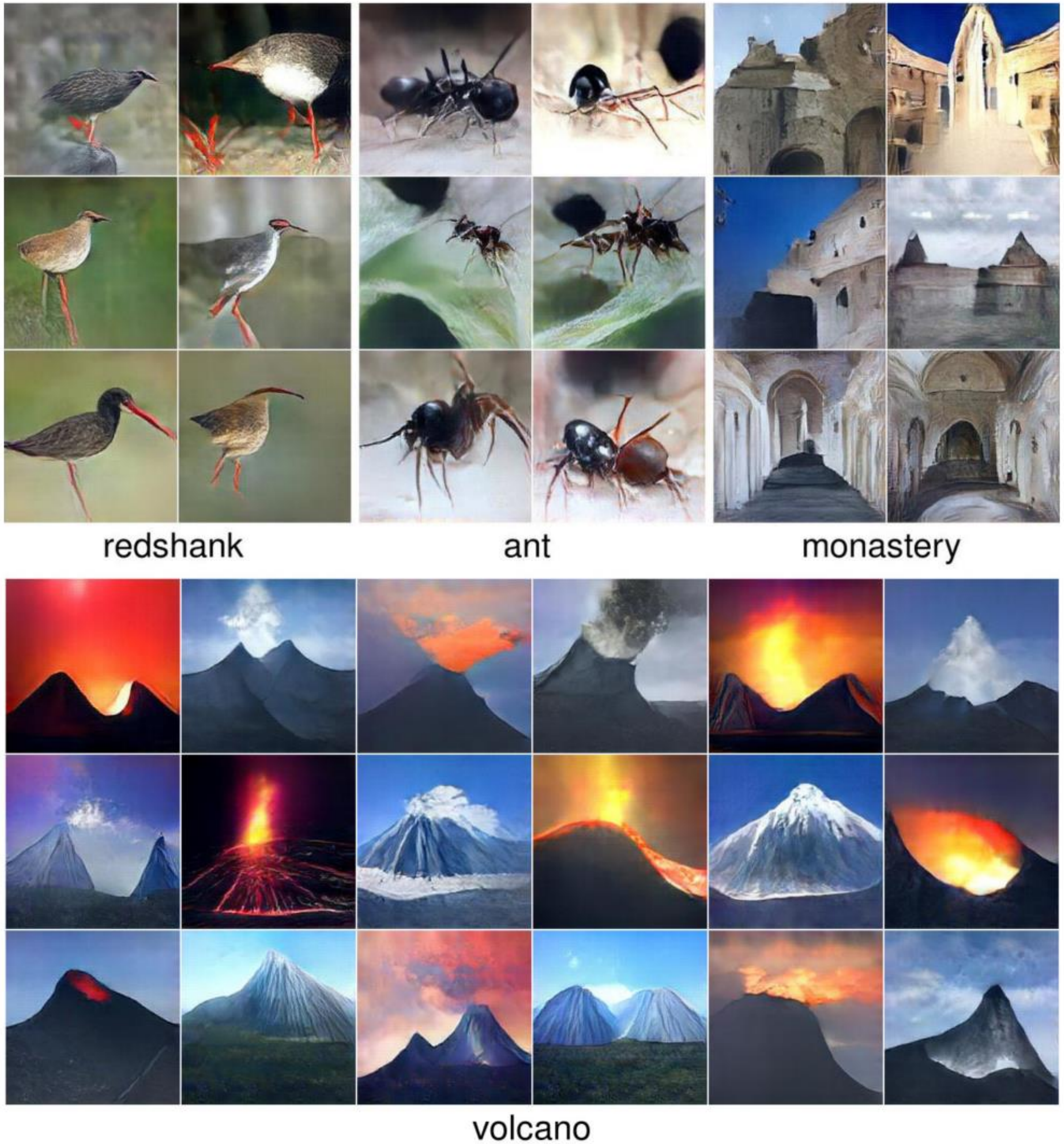
Plug-and-play generative networks

(a) Real: top 9 (b) DGN-AM [36]

(c) Real: random 9 (d) PPGN (**this**)

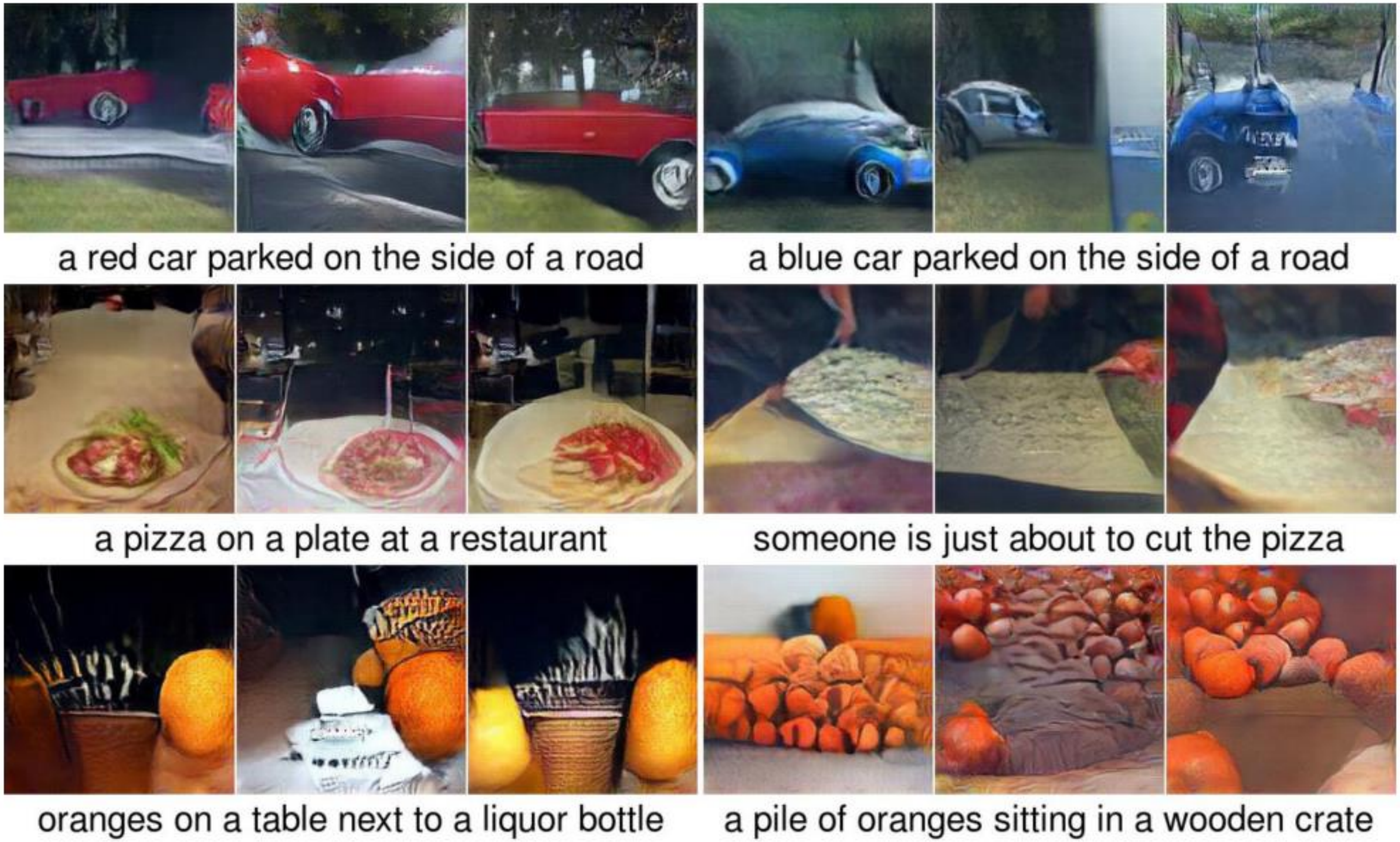


Plug-and-play generative networks



[O] Vision
COMPUTER VISION University of Freiburg

PPGN: sentence to image



Summary

- Perceptual metrics for better image generation
- ConvNets are surprisingly invertible
- Plug-and-play generative networks produce great high resolution images

Sensorimotor control
(learning to play Doom)

Vladlen
Koltun

arxiv 2017

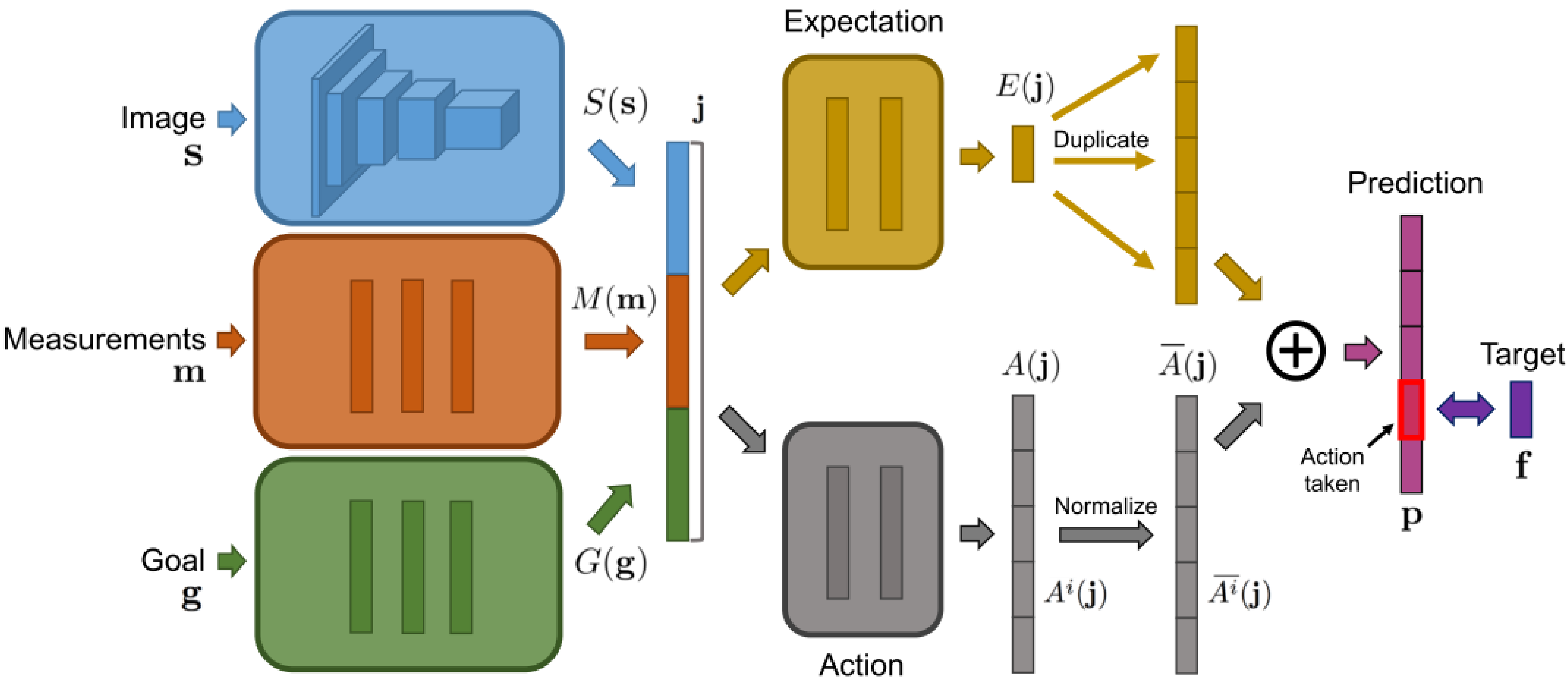
Motivation

Reinforcement learning	Real life
Single goal	Multitude of goals
Scalar reward	Rich sensory stream
Maximize returns	Explore the world

Reinforcement learning vs real life

- Idea: predict future *measurements*
  - Hunger, pain, cold, sleep
  - Health, ammo, frags
- Formulate goals in terms of these
  - Minimize hunger, pain, cold, sleepiness
  - Maximize health, ammo, frags
- Predict with simple supervised training

Architecture



ViZDoom tasks

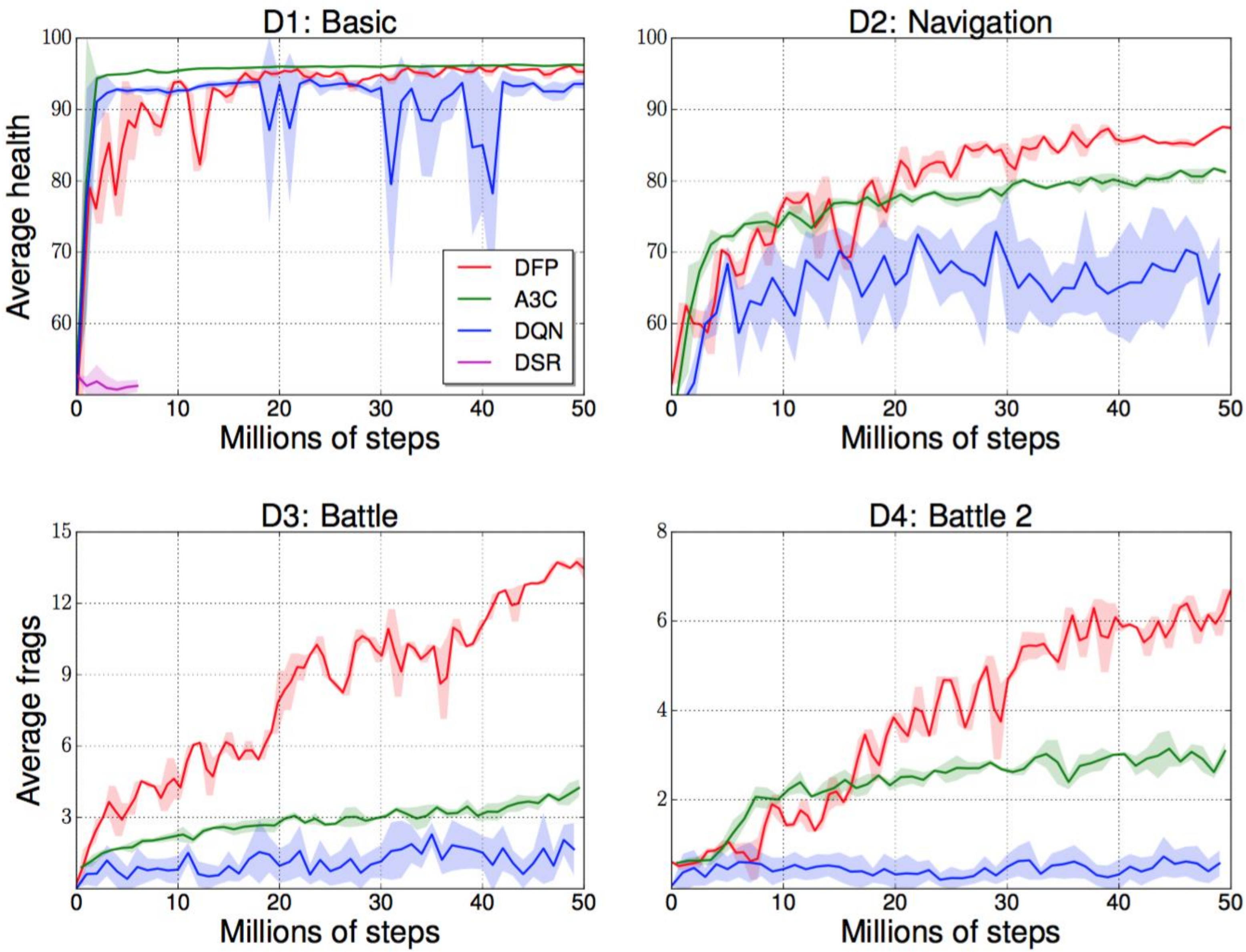
D1: Basic

D2: Navigation

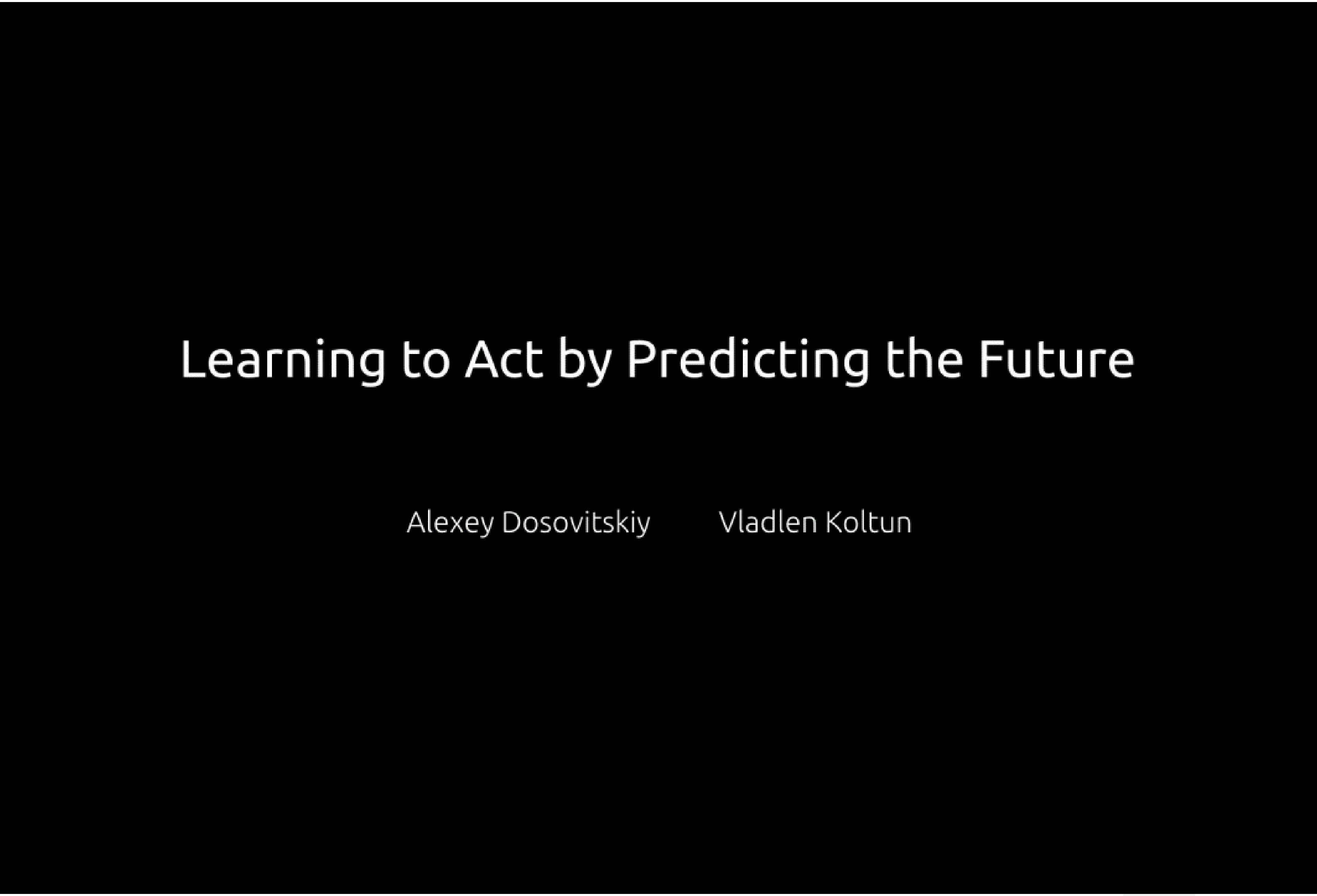
D3: Battle

D4: Battle 2

ViZDoom results



ViZDoom results



ViZDoom competition

Place	Team	1	2	3	4	5	6	7	8	9	10	11	12	Total
1	**IntelAct**	29	21	23	21	6	11	9	6	30	32	33	35	256
2	**The Terminators**	22	17	21	15	13	12	6	5	14	13	13	13	164
3	**TUHO**	8	11	13	12	0	-1	-1	-4	2	2	6	3	51
4	ColbyCS	2	4	0	1	-1	0	-1	0	3	3	4	3	18
5	5vision	3	0	4	2	1	0	1	0	0	-1	1	1	12
6	Ivomi	3	0	1	0	1	-1	-4	-4	1	1	0	0	-2
7	PotatoesArePrettyOk	0	0	2	0	-1	-3	-1	0	-2	-1	-1	-2	-9

What next?

- Deep learning and simulation
- Learning models of environments
  - Future prediction
  - Planning
  - Analysis by synthesis
- Coupled perception and control

Looking for interns! (Munich, Santa Clara)

Summary

End-to-end motion and depth estimation

Inverting ConvNets and perceptual metrics

Visualizing neurons and generating images

Sensorimotor control

Looking for interns! (Munich, Santa Clara)