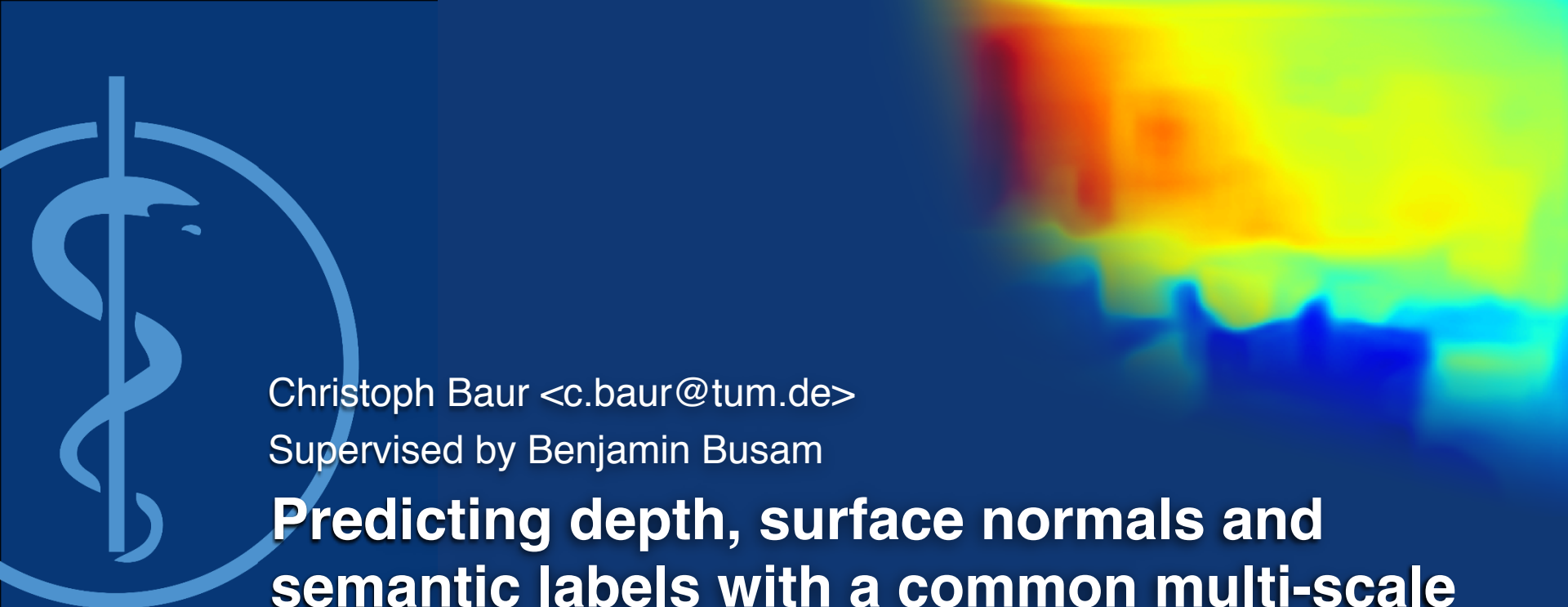


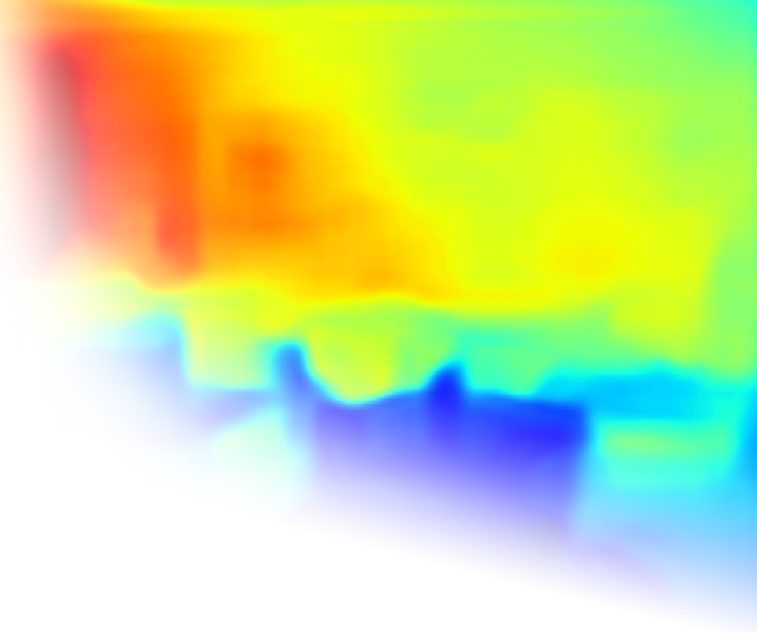
Christoph Baur <c.baur@tum.de>
Supervised by Benjamin Busam

Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture

by David Eigen and Rob Fergus (arXiv 2014)

Technische Universität München + Johns Hopkins Whiting School of Engineering

Agenda



1. Introduction and Motivation
2. Related Work
3. Contribution
    A. Architecture
    B. Predicting Depth
    C. Predicting Surface Normals
    D. Semantic Labeling of RGB-D Data
    E. Training
    F. Experiments & Performance Comparison
4. ICCV 2015
5. Conclusion

1. Introduction and Motivation

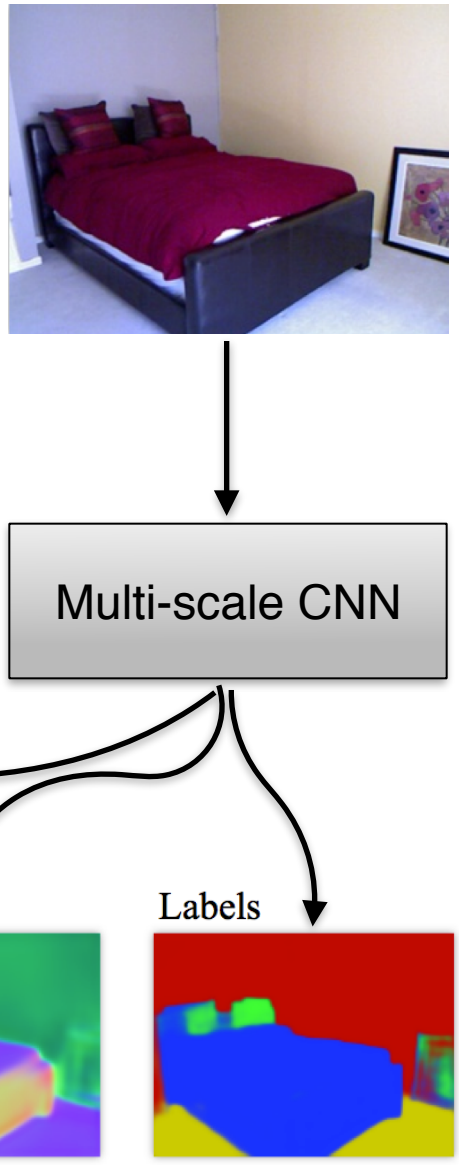
What the authors propose:

A single, multi-scale, generic CNN for

- Depth Prediction from monocular images
- Surface Normals Prediction from monocular images
- Semantic Labeling of RGB-D (and also RGB) data

Which

- requires only tiny modifications
- operates in realtime (30hz on a GPU)
- set the state-of-the-art in all of the above disciplines

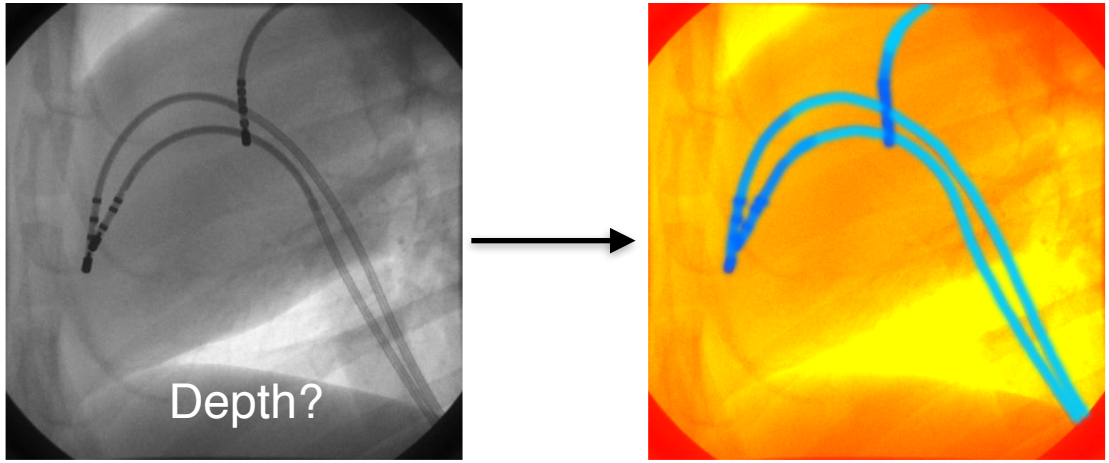


1. Introduction and Motivation

Why inferring 3D scene information on single RGB data:

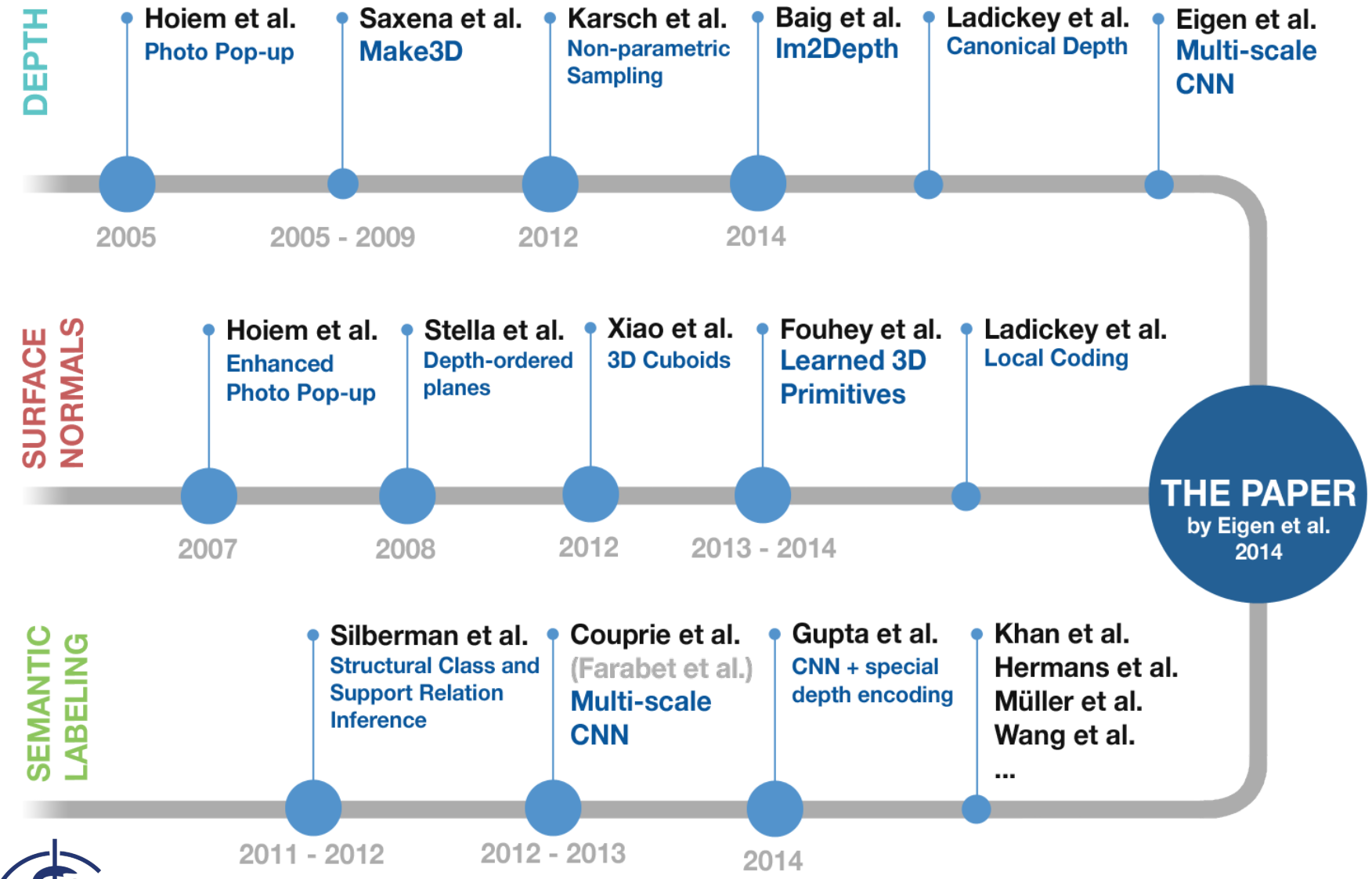
- many low cost sensors available
- generate a prior for other algorithms
- e.g. Human Pose Estimation, Robot Navigation, 3D Scene Reconstruction, Defocusing, Scene Understanding

Medical Example
Depth Prediction in X-ray images



Source: [10]

2. Related Work



3. Contribution - Architecture

Scale 1:

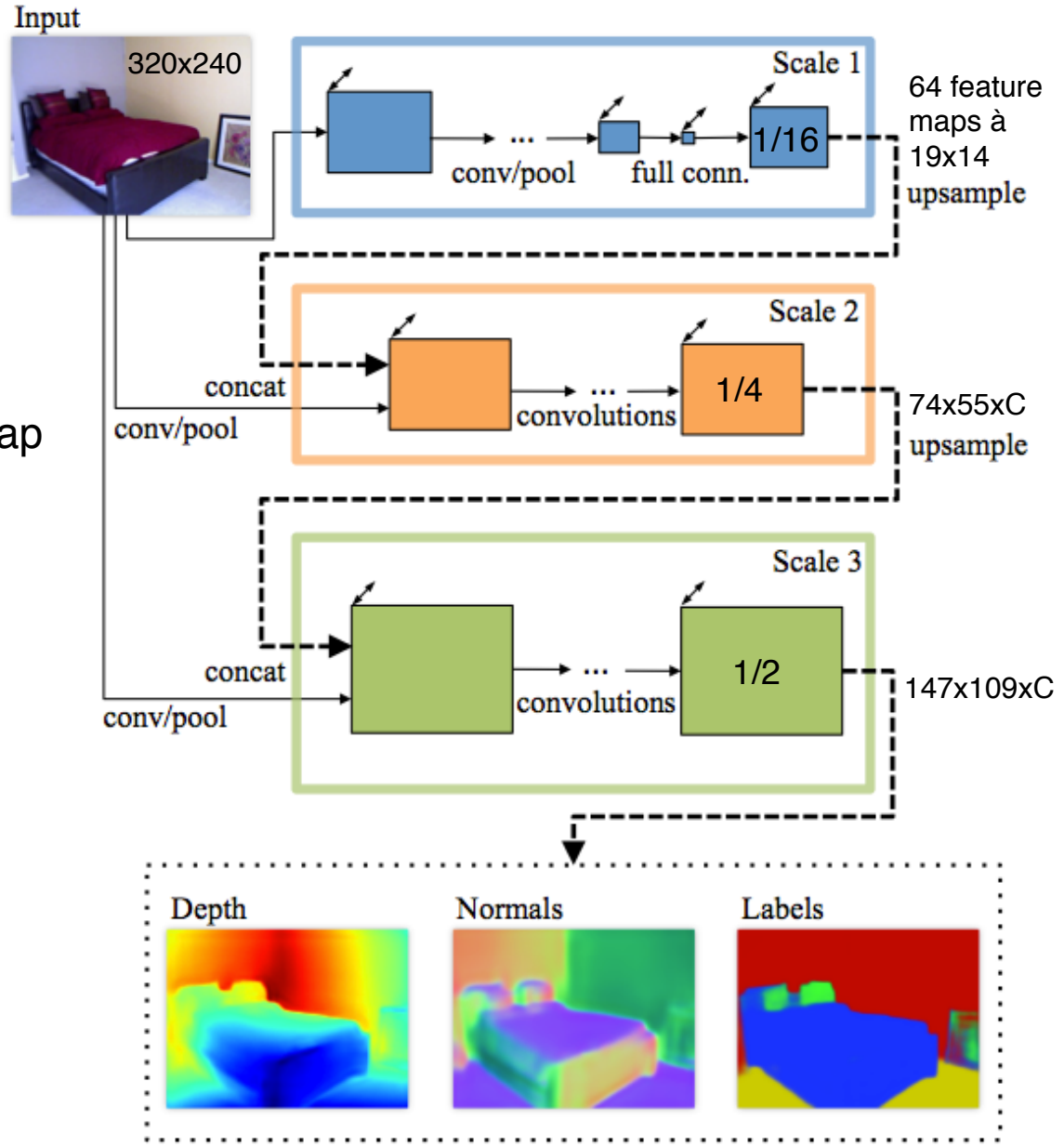
- extract features from the entire image
- obtain 19x14x64 feature map

Scale 2:

- mid-level predictions
- concat feature maps from scale 1 with scale 2 convolution+pooling output
- do more convolutions

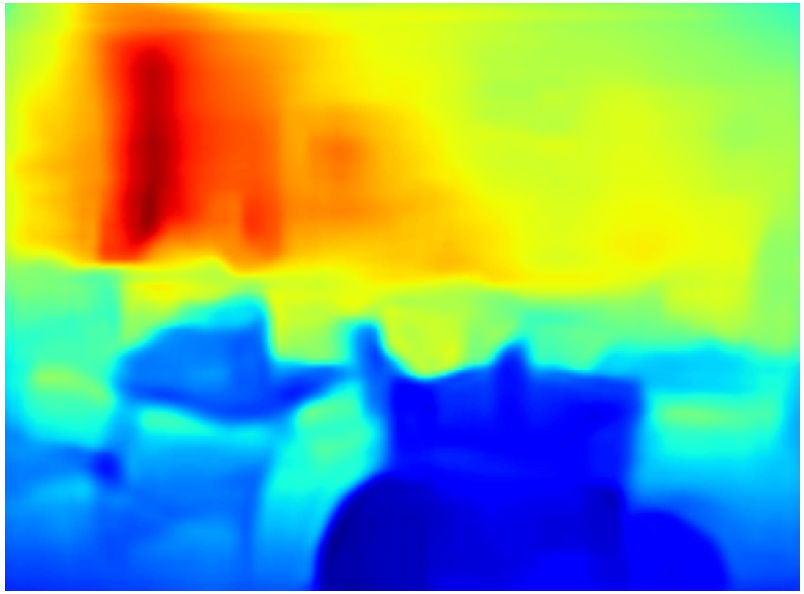
Scale 3:

- similar to scale 2
- but even finer stride for convolution+pooling



3. Contribution - Predicting Depth

Goal: Predict absolute depth at each pixel



3. Contribution - Predicting Depth

Loss Function:

$$D = \log(\text{predicted depth map})$$

$$D^* = \log(\text{groundtruth depth map})$$

$$d = D - D^* = log(\frac{\text{predicted depth map}}{\text{groundtruth depth map}})$$

$$L_{depth}(D, D^*) = \frac{1}{n}\sum_i d_i^2 - \underbrace{\frac{1}{2n^2}(\sum_i d_i)^2}_{-\frac{1}{2n^2}\sum_i d_i^2 - \frac{1}{n^2}\sum_{i \neq j} d_i d_j} + \frac{1}{n}\sum_i [(\nabla_x d_i)^2 + (\nabla_y d_i)^2]$$

l2-Error

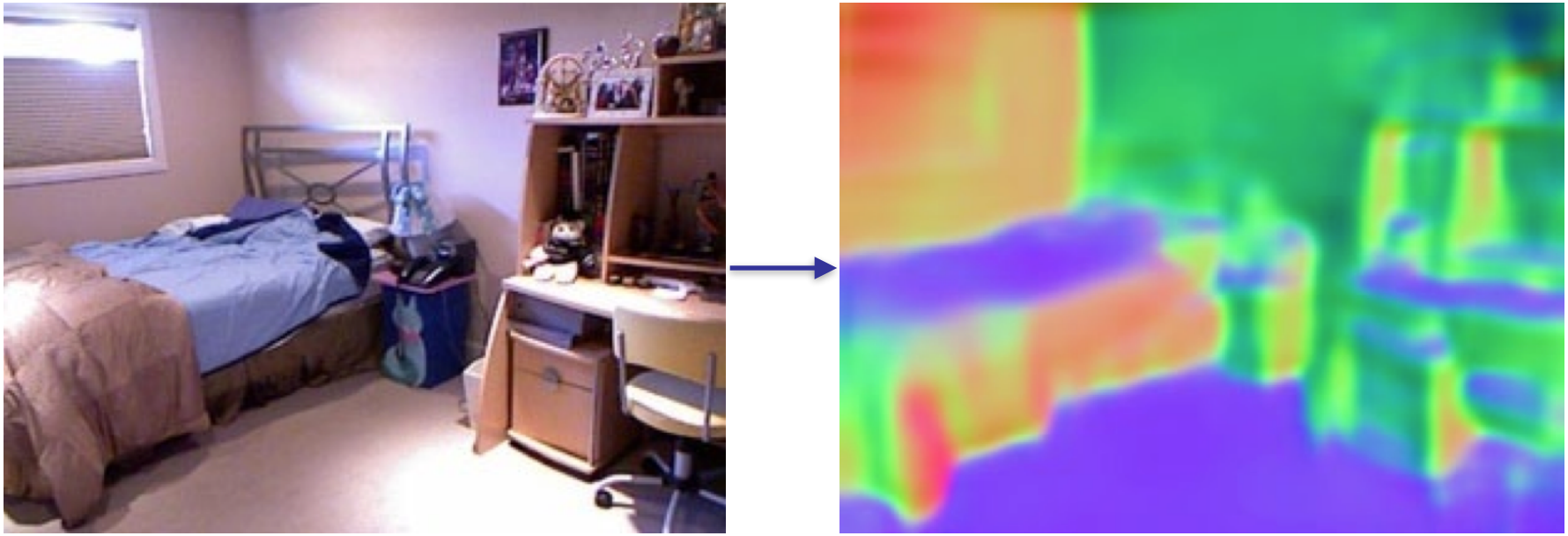
2nd-Order l2

Scale Consistency

Spatial Consistency

3. Contribution - Predicting Surface Normals

Goal: Predict x,y,z components of the surface normal vector at each pixel



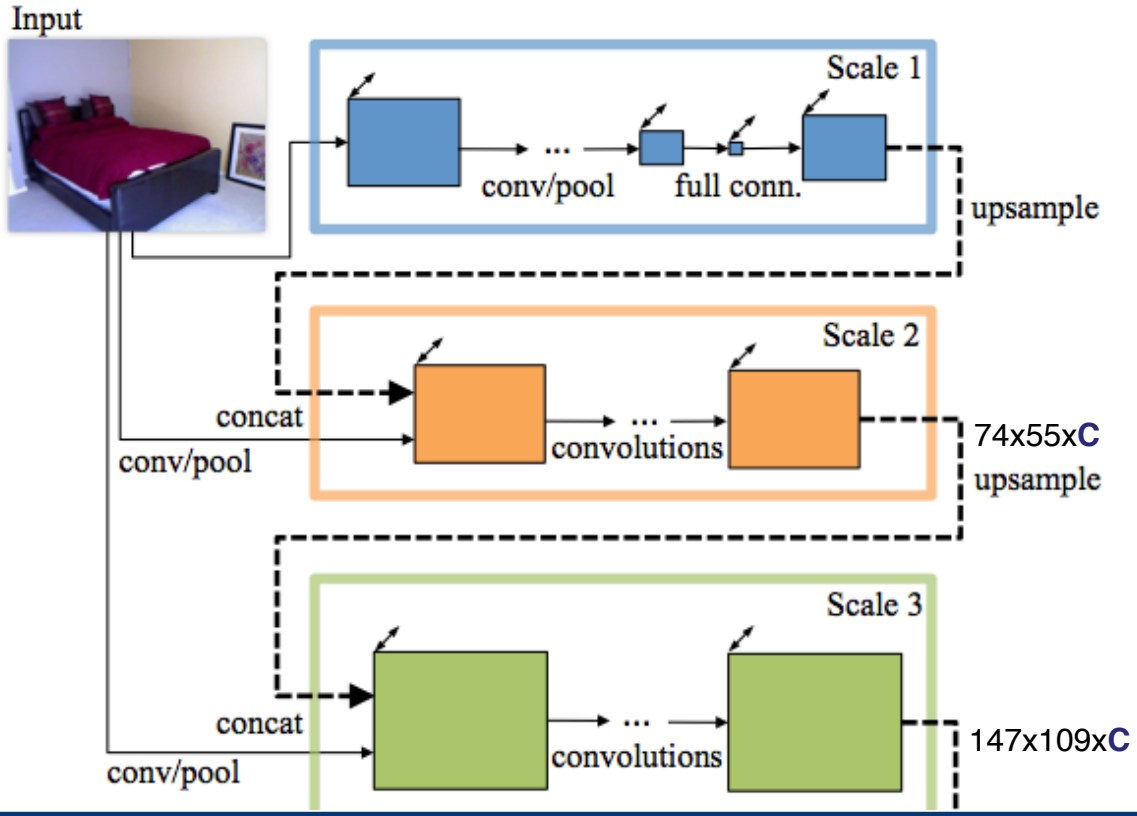
3. Contribution - Predicting Surface Normals

Loss Function: Dot product of predicted and ground truth normal

$$L_{normals}(N, N^*) = -\frac{1}{n}\sum_i N_i \cdot N_i^* = -\frac{1}{n} N \cdot N^*$$

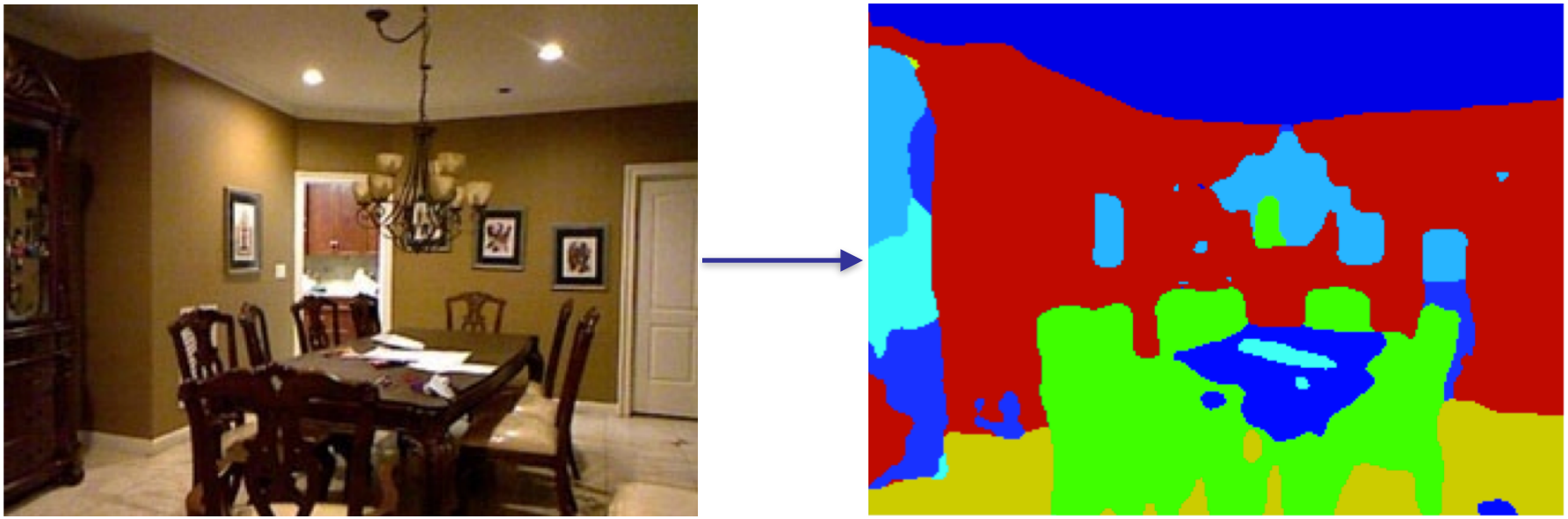
Modification:

- #Channels = 3



3. Contribution - Predicting Semantic Labels in RGB-D data

Goal: Predict class label at each pixel



3. Contribution - Predicting Semantic Labels in RGB-D data

Loss Function: Cross-entropy loss

$$L_{semantic}(C, C^*) = -\frac{1}{n} \sum_i C_i^* \log(C_i)$$

$$C_i = \frac{\exp(z_i)}{\sum_c \exp(z_{i,c})}$$

Modification:

- #Channels = #Classes
- Scale 2 and 3:
  - Use RGB, Depth and computed Normals as input
  - Enforce Independence of input types:
    Convolve each input type separately with a different set of 32x9x9 filters
- Concat resulting 3 feature sets with up sampled output from previous scale network

3. Contribution - Training

Data:

- NYUDepth v2[2] & SiftFlow datasets[3]
- Standard augmentation: random scaling, translation, flips etc…

Init:

- ImageNet trained weights for convolutional layers of scale 1
- random weights for fully connected layers and all layers of the other scales

Phase 1:

- Attach Loss-function to Scale 2 (Scale 1 simply yields feature maps)
- Jointly train Scale 1 and 2 on 5M samples

Phase 2:

- Fix params of Scale 1 and 2
- Attach Loss-function to Scale 3
- Train weights of Scale 3 also on 5M samples

3. Contribution - Experiments & Performance Comparison

Depth prediction on NYUDepth v2 [2]

Metric	Karsch[4]	Baig[5]	EigenDepth[6]	This	
RMSE	1.2	1.0	0.877	**0.753**	**Less is better**

Surface Normal prediction on NYUDepth v2

Metric	Ladickey[7]	Fouhey 1[8]	Fouhey 2[9]	This	
Mean Angle Distance	32.5	34.2	35.1	**23.1**	**Less is better**
Median Angle Distance	22.3	30.0	19.2	**15.1**	**Less is better**
% Within 11.25° Degrees	27.4	18.5	37.6	**39.4**	**More is better**

Comparing to ground truth generated as described by Fouhey 1's method

3. Contribution - Experiments & Performance Comparison

Semantic Labeling on NYUDepth v2 (4, 13 and 40 classes)

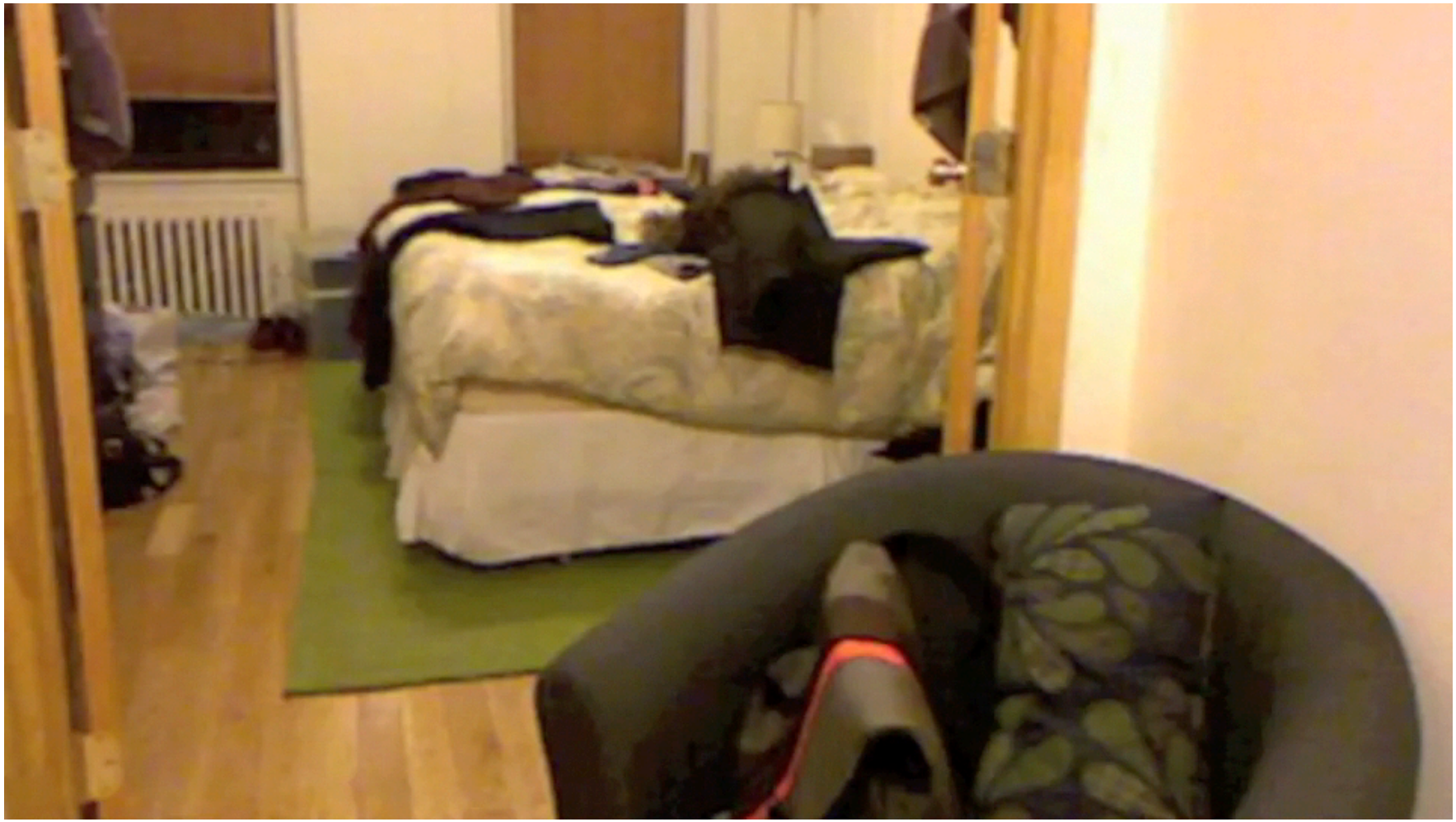
Metric Pixel Accuracy	Couprie[11]	Khan[12]	Gupta 13[13]	Gupta 14[14]	This	
4 classes	64.5	69.2	78.0	-	**80.6**	**More is better**
13 classes	52.4	58.3	-	-	**70.5**	**More is better**
40 classes	-	-	59.1	60.3	**62.9**	**More is better**

Semantic Labeling on the SiftFlow Dataset (33 classes) [3]

Metric	Farabet 1[15]	Farabet 2[15]	Tighe[16]	This 1	This 2	
Pixel Accuracy	78.5	74.2	78.6	**84.0**	81.6	**More is better**
Per-Class Accuracy	29.6	46.0	39.2	42.0	**48.2**	**More is better**

***This 2: class balanced model, This 1: regular class distribution model**

Examples



3. Contribution - Experiments & Performance Comparison

Contribution of scales:

- progressive improvements if scales are added
- largest single contribution:
  - depth and normals: Scale 1
  - semantic labels: Scale 2 (because of depth & normals input)

Importance of depth and normals for semantic labeling

- Scale 2 with RGB only: baseline
- Scale 1+2 with RGB only: +
- Scale 2 with RGB + predicted depth/normals: +
- Scale 1+2 with RGB + predicted depth/normals: +
- Scale 2 with RGB + ground truth depth/normals: ++++
- Scale 1+2 with RGB + ground truth depth/normals: +++++

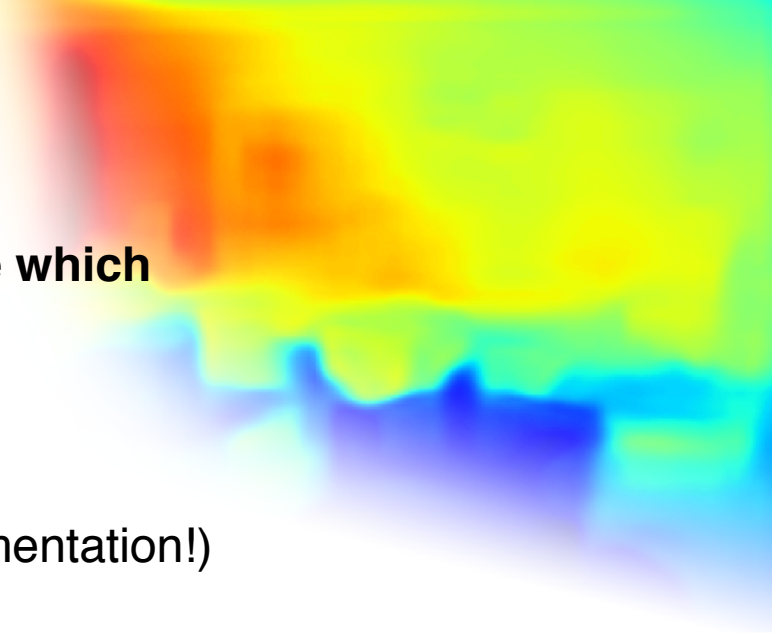
4. ICCV 2015

Eigen and Fergus submit refined work to ICCV 2015 [18]

Changes:

- Further improvement: 15% gain in depth prediction
- Increased receptive field size of the scale 1 CNN
- Initialized weights of convolutional layers from VGG-Net[17]
- Also evaluated on the Pascal VOC[18] dataset
- Combined Depth and Normals: Share Scale 1 (x1.6 speed-up)

5. Conclusion



A single, multi-scale, generic CNN architecture which

- sets the state-of-the-art in 3 different tasks
- probably suited for other tasks as well
- realtime (30Hz) on a GPU!
- requires no pre- or post processing (no segmentation!)
- End-to-end training and prediction

Future work

- Use sparsely labeled data for faster training
- Could also be applied to other tasks such as instance labeling
- I believe it could be applied to depth prediction on X-ray images

References

[1] Eigen, David, and Rob Fergus. "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture." arXiv preprint arXiv:1411.4734 (2014).
[2] Silberman, Nathan, et al. "Indoor segmentation and support inference from RGBD images." Computer Vision–ECCV 2012. Springer Berlin Heidelberg, 2012. 746-760.
[3] Liu, Ce, et al. "Sift flow: Dense correspondence across different scenes." Computer Vision–ECCV 2008. Springer Berlin Heidelberg, 2008. 28-42.
[4] Karsch, Kevin, Ce Liu, and Sing Bing Kang. "Depth extraction from video using non-parametric sampling." Computer Vision–ECCV 2012. Springer Berlin Heidelberg, 2012. 775-788.
[5] Baig, Mohammad Haris, et al. "Im2depth: Scalable exemplar based depth transfer." Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on. IEEE, 2014.
[6] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." Advances in Neural Information Processing Systems. 2014.
[7] L'ubor Ladický, Bernhard Zeisl, and Marc Pollefeys. "Discriminatively Trained Dense Surface Normal Estimation."
[8] Fouhey, David F., Arpan Gupta, and Martial Hebert. "Data-driven 3D primitives for single image understanding." Computer Vision (ICCV), 2013 IEEE International Conference on. IEEE, 2013.
[9] Fouhey, David Ford, Abhinav Gupta, and Martial Hebert. "Unfolding an indoor origami world." Computer Vision–ECCV 2014. Springer International Publishing, 2014. 687-702.
[10] Milletari, Fausto, Nassir Navab, and Pascal Fallavollita. "Automatic detection of multiple and overlapping EP catheters in fluoroscopic sequences." Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013. Springer Berlin Heidelberg, 2013. 371-379.

References (Continued)

[11] Couprie, Camille, et al. "Indoor semantic segmentation using depth information." arXiv preprint arXiv: 1301.3572 (2013).
[12] Khan, Salman Hameed, et al. "Geometry driven semantic labeling of indoor scenes." Computer Vision–ECCV 2014. Springer International Publishing, 2014. 679-694.
[13] Gupta, Swastik, Pablo Arbelaez, and Jagannath Malik. "Perceptual organization and recognition of indoor scenes from RGB-D images." Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013.
[14] Gupta, Saurabh, et al. "Learning rich features from RGB-D images for object detection and segmentation." Computer Vision–ECCV 2014. Springer International Publishing, 2014. 345-360.
[15] Farabet, Clement, et al. "Scene parsing with multiscale feature learning, purity trees, and optimal covers." arXiv preprint arXiv:1202.2160 (2012).
[16] Tighe, Joseph, and Svetlana Lazebnik. "Finding things: Image parsing with regions and per-exemplar detectors." Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013.
[17] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
[18] Everingham, Mark, et al. "The pascal visual object classes challenge: A retrospective." International Journal of Computer Vision 111.1 (2014): 98-136.

Thanks for your attention!

Any Questions?

Appendix: Depth Prediction Loss Function

Formulation in the NIPS (two-scale architecture) paper

Average Depth Error

$$L_{depth}(D, D^*) = \frac{1}{2n} \sum_i [d_i - \frac{1}{n} \sum_i (d_i)]^2$$

leads to (my calculus):

$$= \frac{1}{2n} \sum_i d_i^2 - \frac{1}{n^2} \sum_{i,j} (d_i d_j) + \frac{1}{2n^2} (\sum_i d_i)^2$$

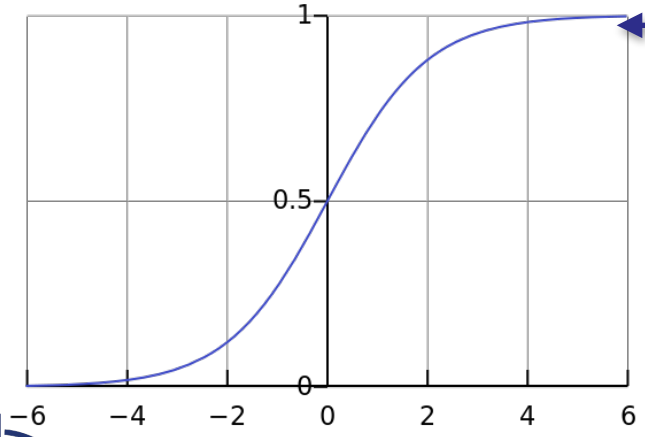
leads to (their calculus):

$$= \frac{1}{n} \sum_i d_i^2 - \frac{1}{n^2} \sum_{i,j} d_i d_j = \frac{1}{n} \sum_i d_i^2 - \frac{1}{n^2} \left(\sum_i d_i \right)^2$$

Appendix: Cross Entropy

Cross Entropy

- Consider the Mean Squared Error:
- Minimizing the MSE is the MLE of a (multivariate) normal distribution
- But is the output of the Softmax activation normally distributed?
- Probably not! Look at those exponentials everywhere!
- Cross Entropy is a better choice for sigmoid activations!
- It introduces no learning slowdown!



Gradient of a sigmoid vanishes near 1 or 0 —> Slow learning!

Using Cross-Entropy, this is not the case

Appendix: Training - Phase 2 Remarks

At Scale 3:

- don’t train on entire images as it is computationally heavy
- instead, use random crops of size 74x55 of up sampled scale 2 output concatenated with actual scale 3 input
- x3 speed-up!

Appendix: Jaccard Index, Frequency-Weighted Average Jaccard Index

- Measure overlap between predictions and ground truth:

$$J(\text{predictions}, \text{ground truth}) = \frac{|\text{predictions} \cap \text{ground truth}|}{|\text{predictions} \cup \text{ground truth}|}$$

Example: label predictions = {chair, bed, sofa}, ground truth = {chair, bed, sofa, tv}

$$J(...) = \frac{3}{4}$$

- Frequency Weighted: Average of the Jaccard-index of each class, where each index contributes to the average by the amount of pixels it comprises relative to all pixels