Agenda

- **Deep learning on regular structures**
  - Multi-view representation
  - Volumetric Representation

- Deep learning on meshes

- Deep learning on point cloud and parametric models

Agenda

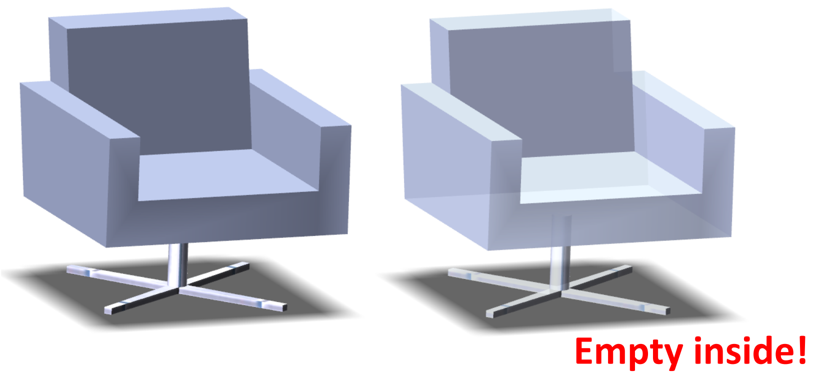
- Deep learning on regular structures
  - **Multi-view representation**
  - Volumetric Representation

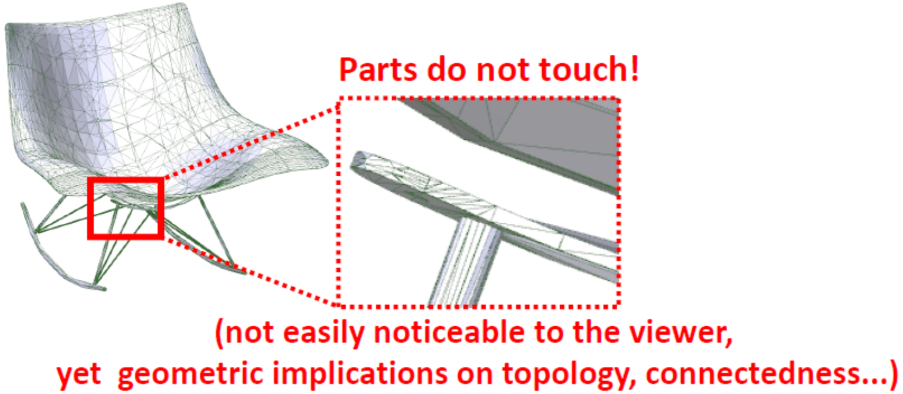
- Deep learning on meshes

- Deep learning on point cloud and parametric models

General idea

- Convert irregular (3D) to regular (images)
- Circumvent any geometric representation artifacts (non-manifold geometry, polygon soups, no interior)





- Leverage pre-trained image-based CNNs
- Similarly to humans, analyze what can be seen: combine surface information from multiple views

Agenda

- Deep Learning Review
- Overview of 3D Deep Learning
- **Deep Learning on Multi-view Representation**
  - **Classification**
  - Segmentation
  - Reconstruction

Task: 3D classification

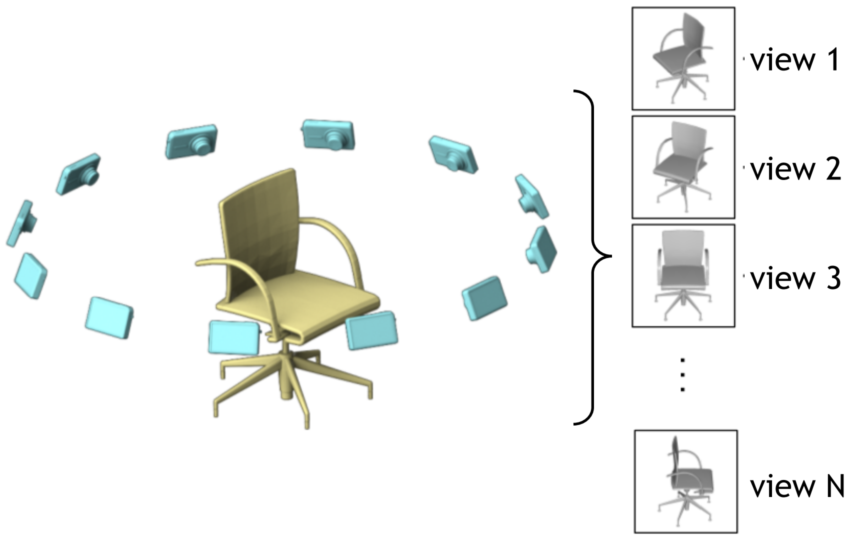
This is a chair!

Given an input shape

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller,
"**Multi-view Convolutional Neural Networks for 3D Shape Recognition**",
Proceedings of ICCV 2015

[credit: Hang Su]

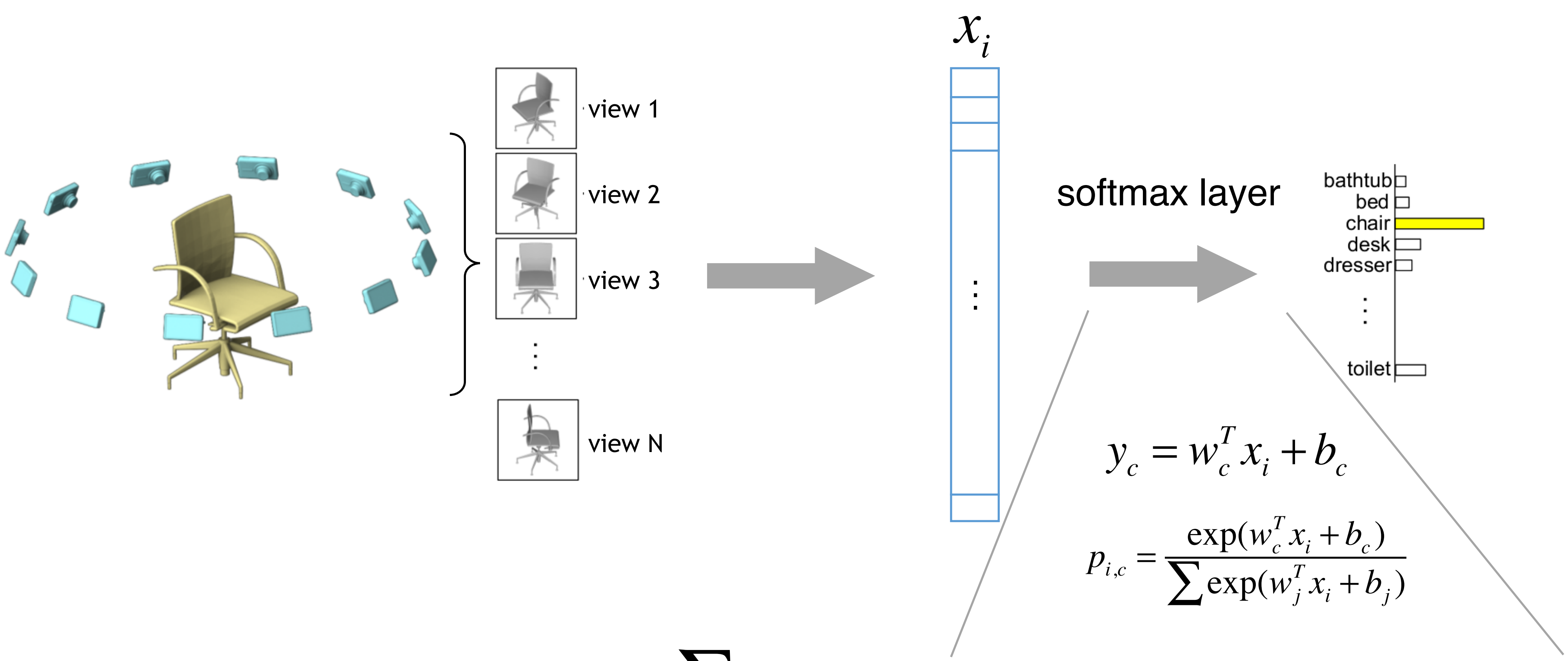
Render with multiple virtual cameras



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller,
"**Multi-view Convolutional Neural Networks for 3D Shape Recognition**",
Proceedings of ICCV 2015

 [credit: Hang Su]

Traditional approach: feature+linear classifier

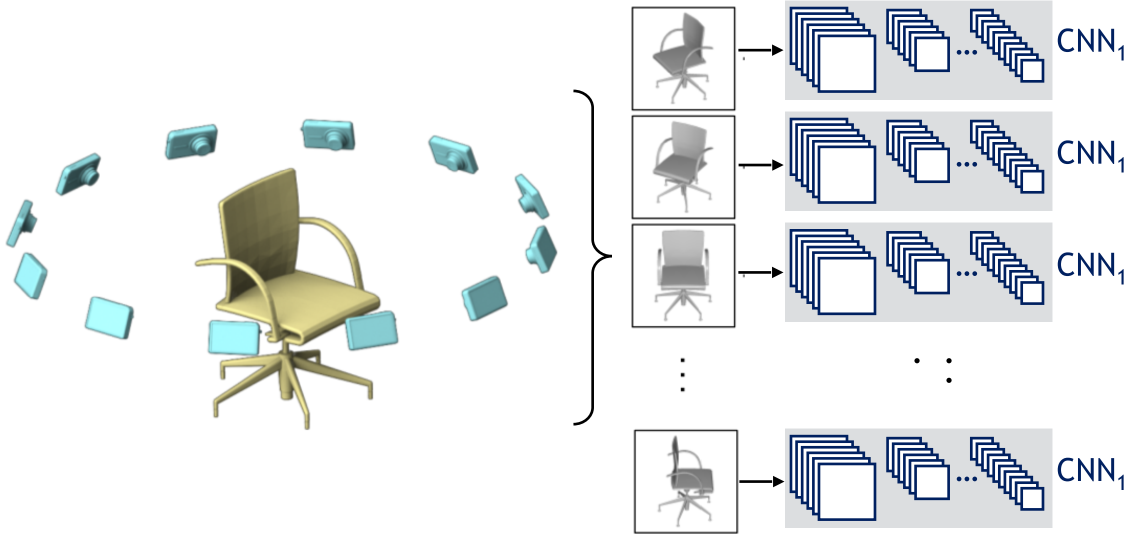


$$y_c = w_c^T x_i + b_c$$

$$p_{i,c} = \frac{\exp(w_c^T x_i + b_c)}{\sum \exp(w_j^T x_i + b_j)}$$

$$\underset{w}{\text{maximize}} \sum_i \log p_{i,l_i}$$

The rendered images are passed through CNN_1 for image features

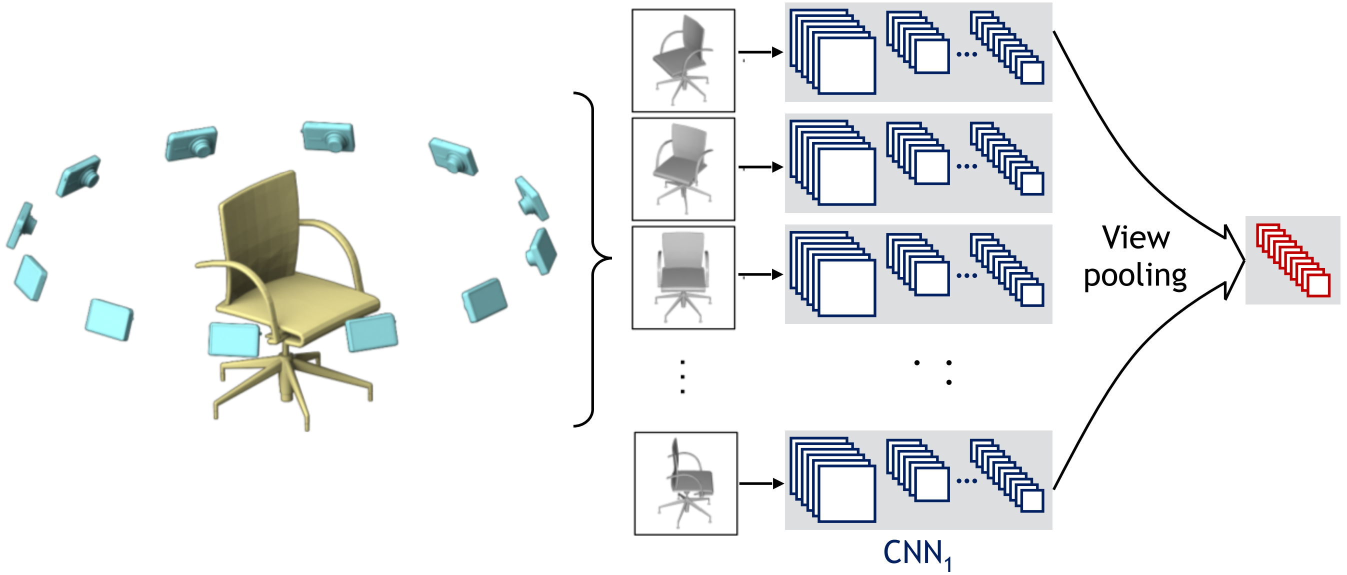


CNN_1: a ConvNet extracting image features

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

[credit: Hang Su]

All image features are combined by view pooling …

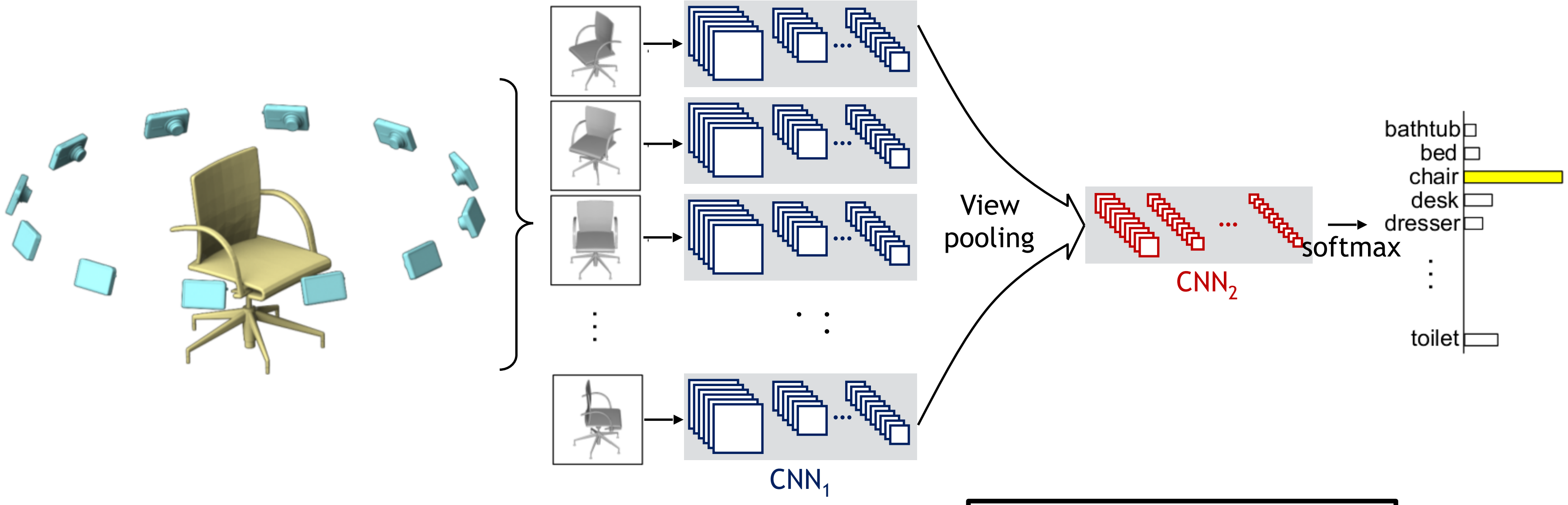


View pooling: element-wise max-pooling across all views

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

 [credit: Hang Su]

… and then passed through CNN_2 and to generate final predictions



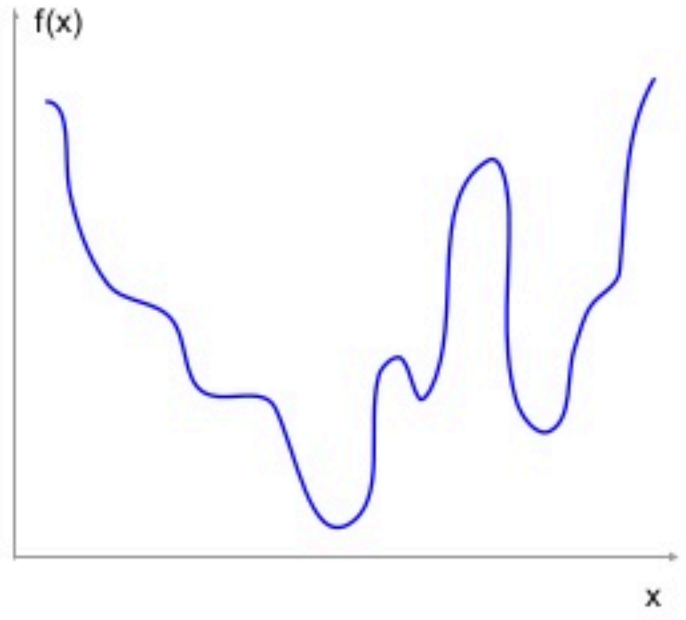
CNN_2: a second ConvNet producing shape descriptors

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

 [credit: Hang Su]

Learning by fine-tuning

- Neural network optimization is non-convex



- In general, training from more data converges at a better local minima

- However, what if your training dataset D is not big?

Learning by fine-tuning (cont.)

Pre-training

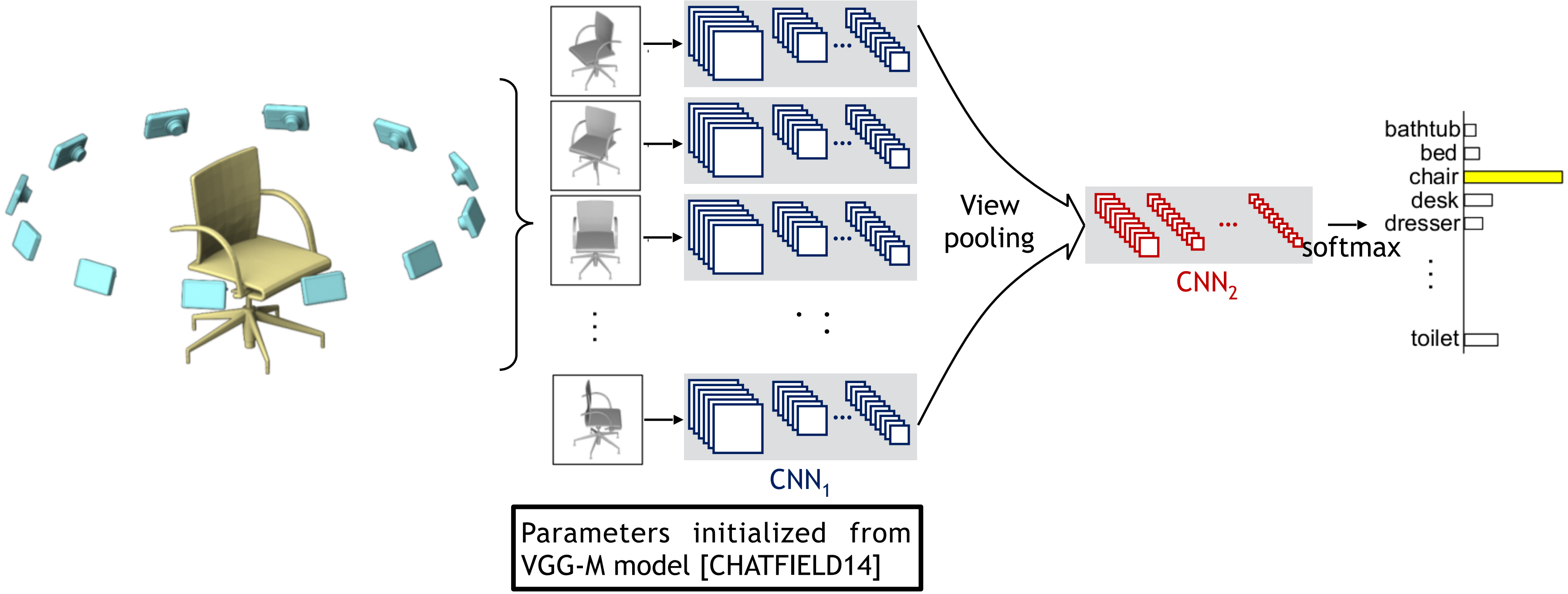
- Find a source of massive data D' with similar statistics
- Learn the network parameters from D'

Fine-tuning

- Starting from the learned parameters on D', minimize the network loss on D

A technique for *transfer learning,* quite effective in practice

Training: network parameters are pre-trained on image classification …

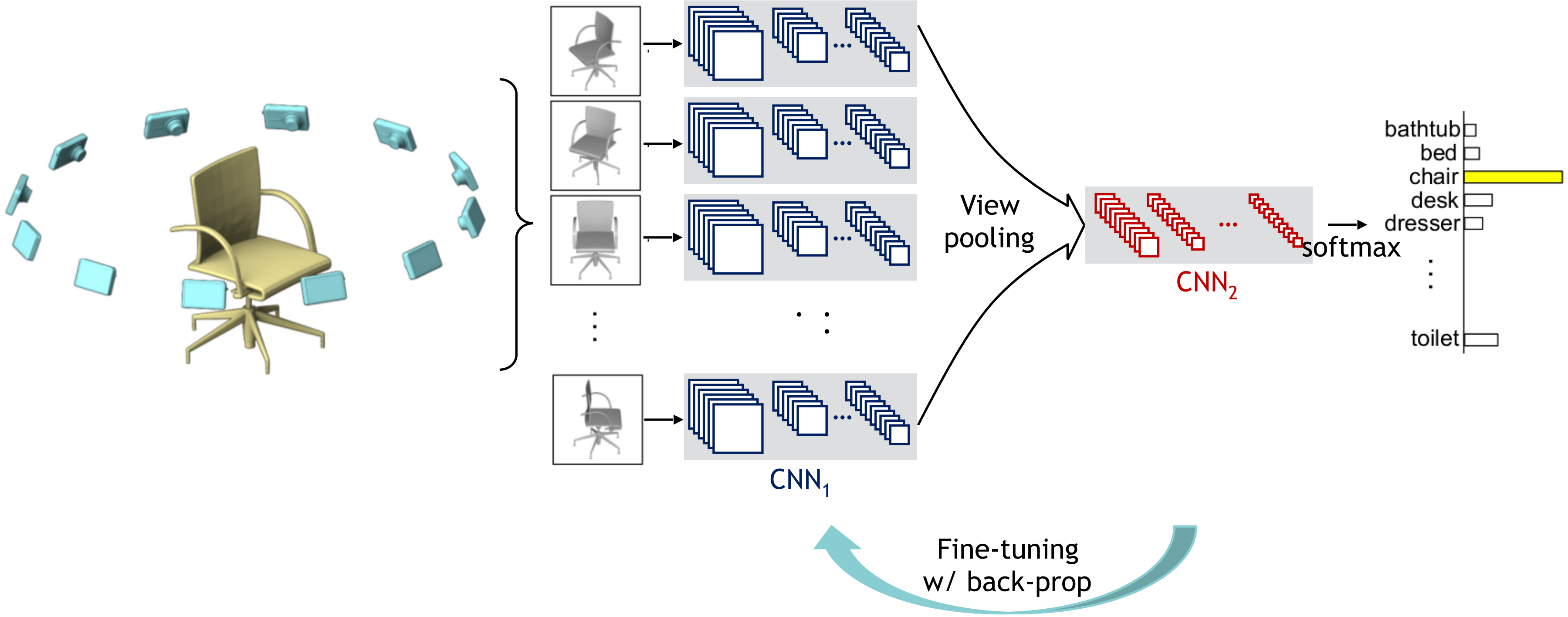


[CHATFIELD14] K. Chatfield et. al., "Return of the Devil in the Details: Delving Deep into Convolutional Nets", BMVC 2014

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller,
"**Multi-view Convolutional Neural Networks for 3D Shape Recognition**",
Proceedings of ICCV 2015

 [credit: Hang Su]

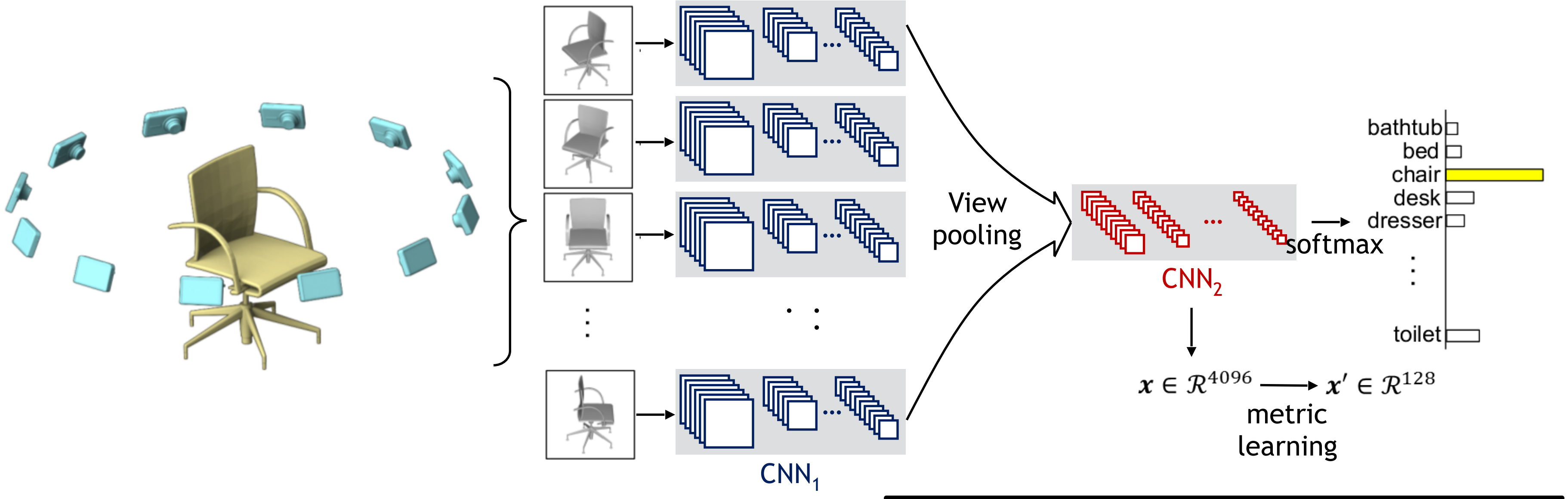
… and then fine-tuned on 3D datasets



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller,
"**Multi-view Convolutional Neural Networks for 3D Shape Recognition**",
Proceedings of ICCV 2015

 [credit: Hang Su]

Extract compact shape descriptor for other applications



Shape descriptor can be extracted from CNN_2, and a low-rank metric is learned w/ good&bad pairs

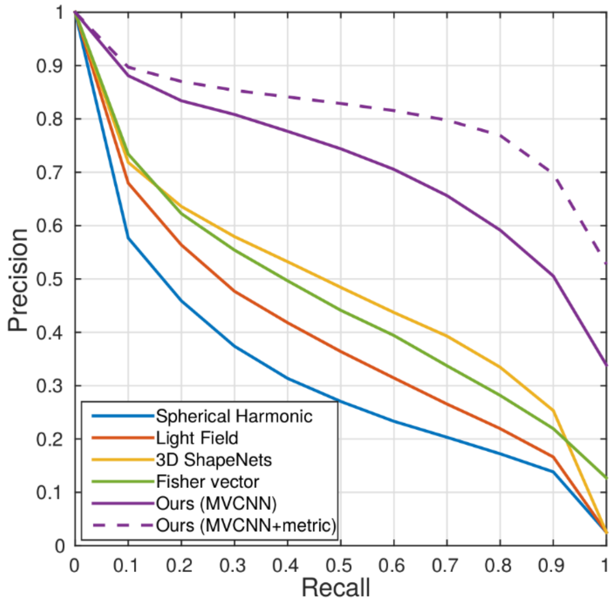
Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

 [credit: Hang Su]

Experiments – classification & retrieval

On **ModelNet40**, compared against:

- 3 existing methods:
  SPH, LFD, 3D ShapeNets
- 2 strong baselines:
  Fisher vectors, CNN

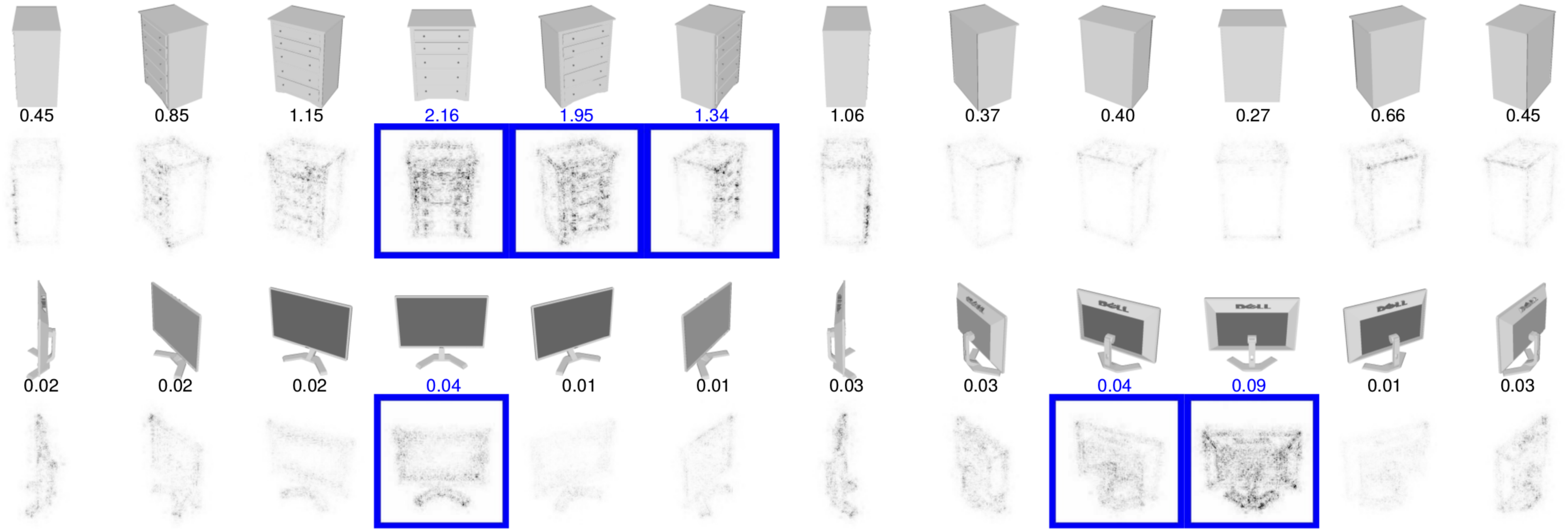


Method	Classification (Accuracy)	Retrieval (mAP)
SPH [16]	68.2%	33.3%
LFD [5]	75.5%	40.9%
3D ShapeNets [37]	77.3%	49.2%
FV, 12 views	84.8%	43.9%
CNN, 12 views	88.6%	62.8%
MVCNN, 12 views	**89.9%**	70.1%
MVCNN+metric, 12 views	89.5%	**80.2%**
MVCNN, 80 views	90.1%	70.4%
MVCNN+metric, 80 views	**90.1%**	**79.5%**

[credit: Hang Su]

Visualization of saliency across views

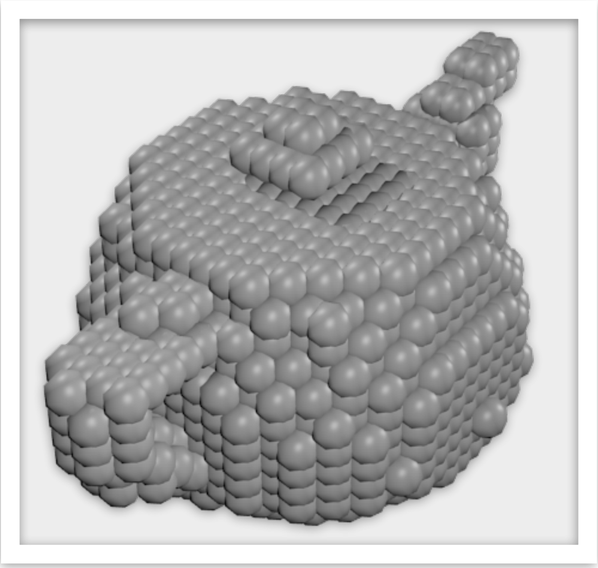
$$[\omega_1, \omega_2 \dots \omega_K] = \left[\left.\frac{\partial F_c}{\partial I_1}\right|_S, \left.\frac{\partial F_c}{\partial I_2}\right|_S \dots \left.\frac{\partial F_c}{\partial I_K}\right|_S \right]$$



[credit: Hang Su]

How do you use multi-view approach for point cloud?

Sphere Rendering ***Images***



Multi-View
Image CNN

[credit: CVPR 2016 spotlight]

Practical multi-view CNN

State-of-the-art performance for **3D mesh classification**

Issues:

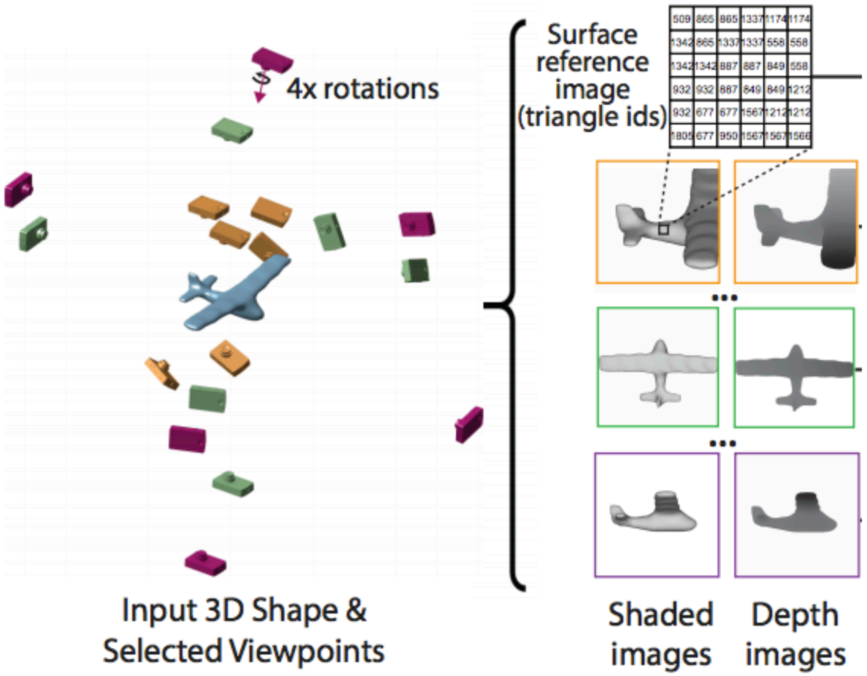
- What viewpoints to select? In particular, where shall we place the camera in a scene?
- What if the input is noisy and incomplete? e.g., point cloud

Agenda

- Deep Learning Review
- Overview of 3D Deep Learning
- **Deep Learning on Multi-view Representation**
    - Classification
    - **Segmentation**
    - Reconstruction

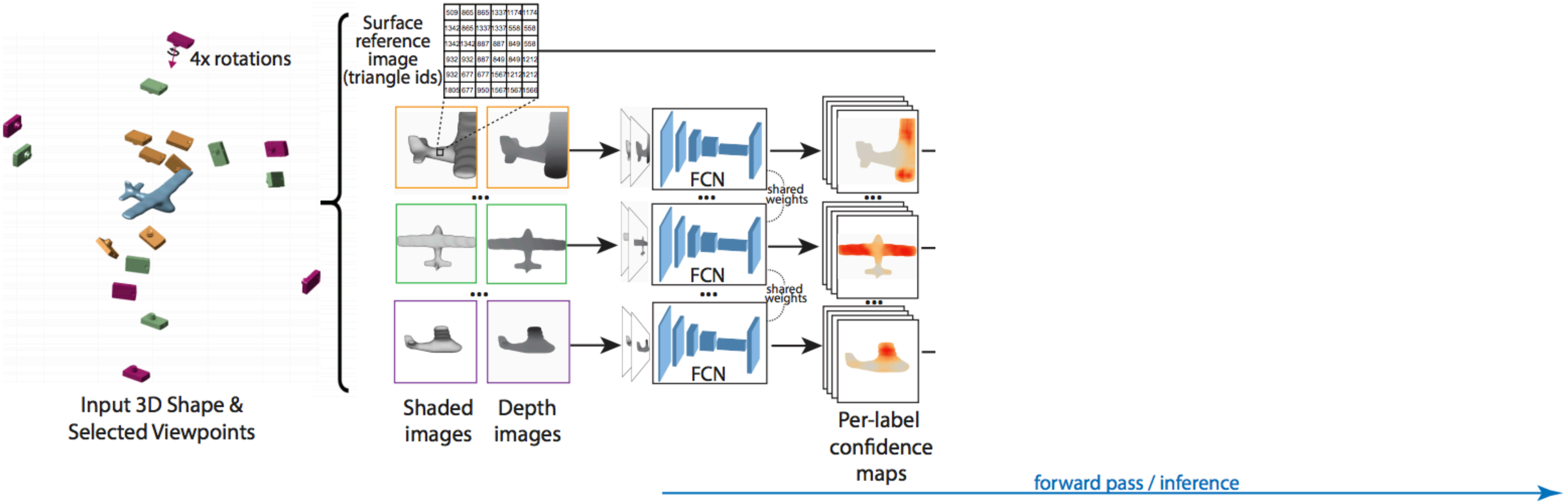
3D segmentation

Basic architecture



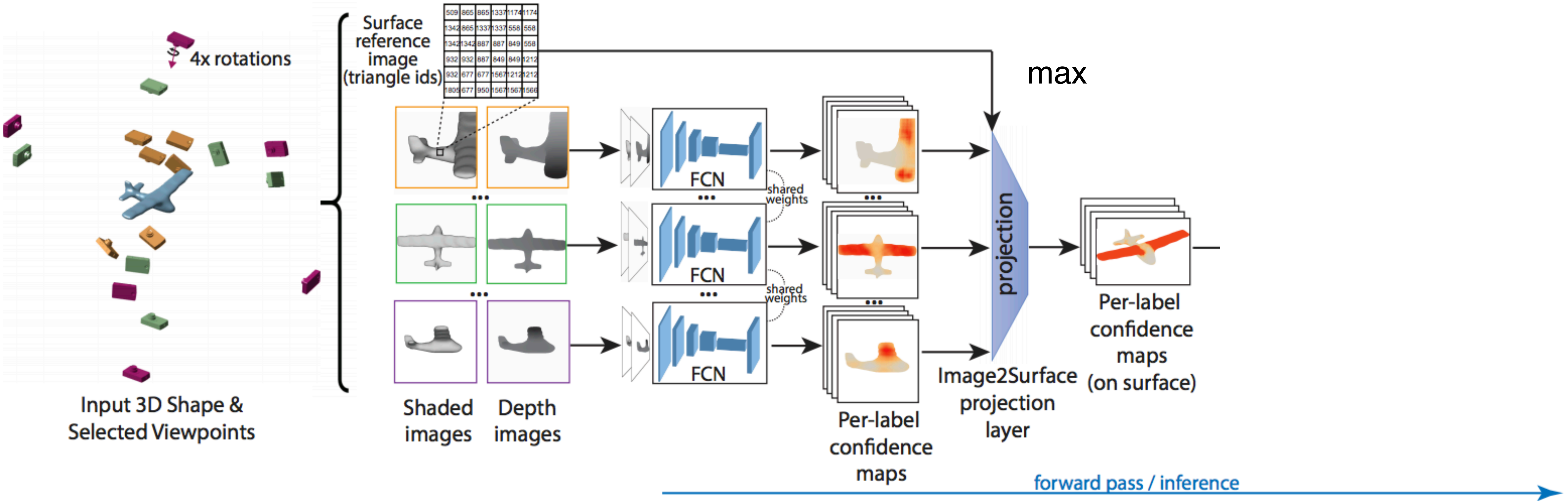
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Basic architecture



Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Basic architecture

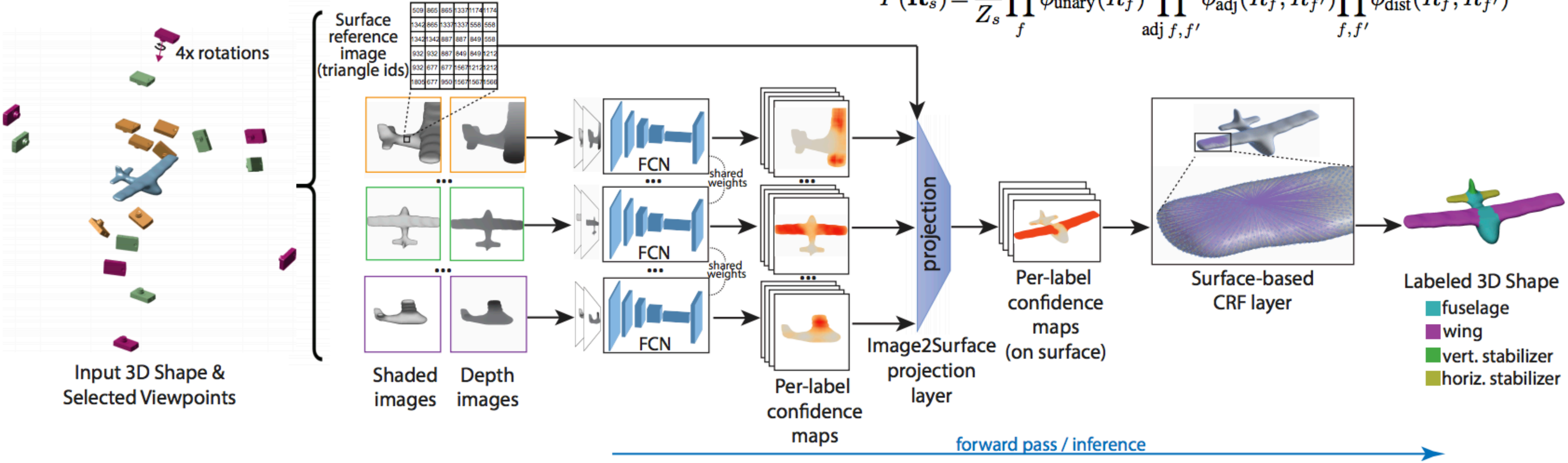


Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Basic architecture

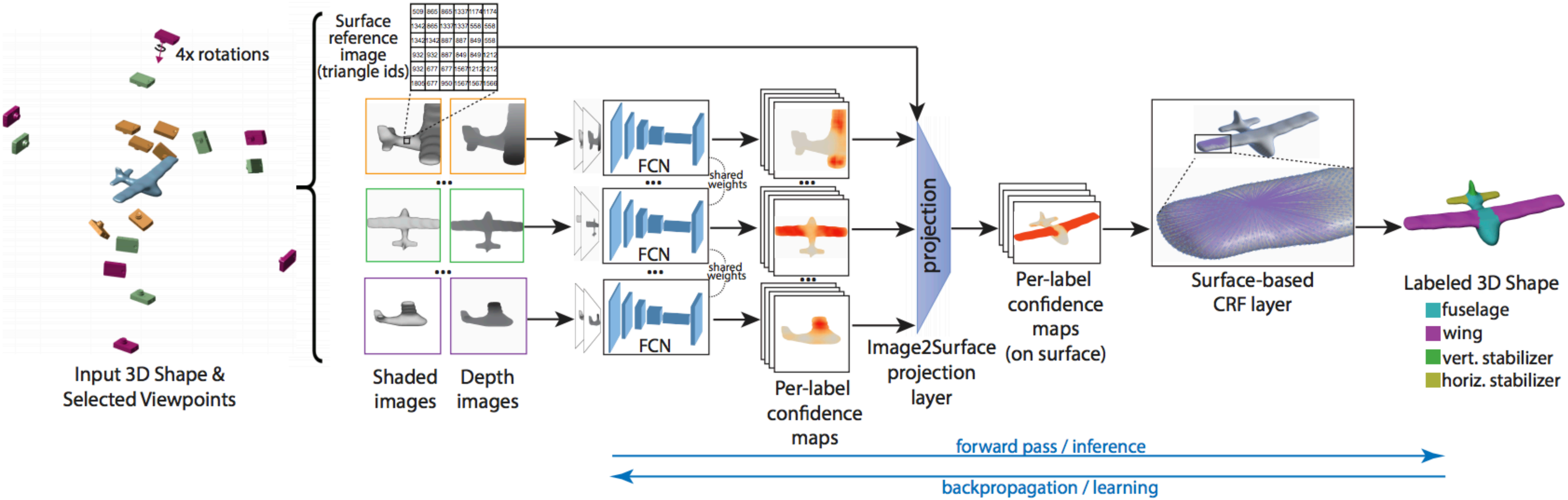
Normal Geodesic

$$P(\mathbf{R}_s) = \frac{1}{Z_s} \prod_{f} \phi_{\text{unary}}(R_f) \prod_{\text{adj } f,f'} \phi_{\text{adj}}(R_f, R_{f'}) \prod_{f,f'} \phi_{\text{dist}}(R_f, R_{f'})$$



Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

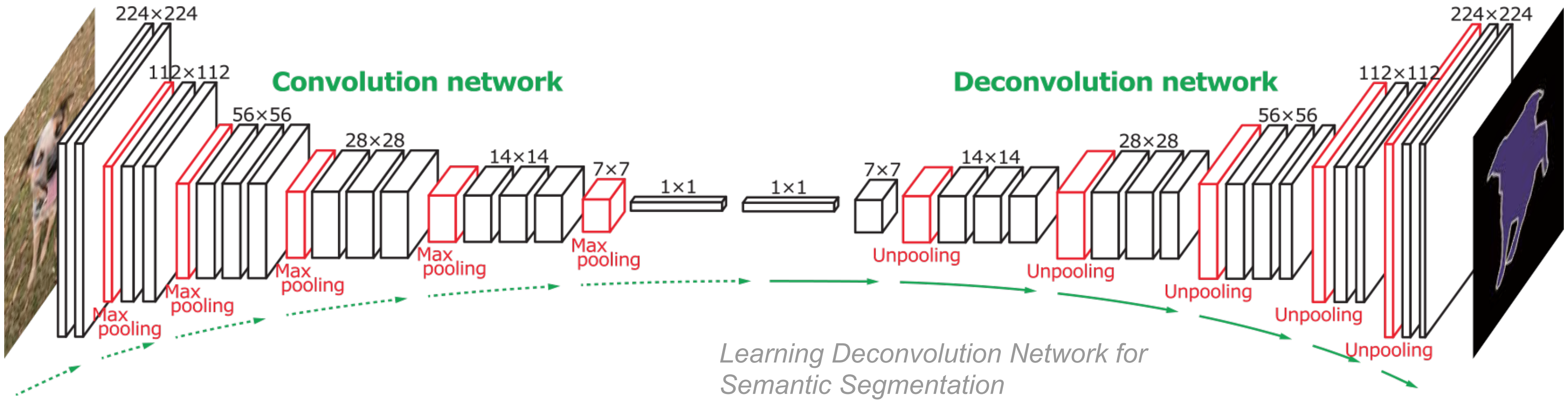
Basic architecture



Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
"3D Shape Segmentation with Projective Convolutional Networks",
CVPR2017

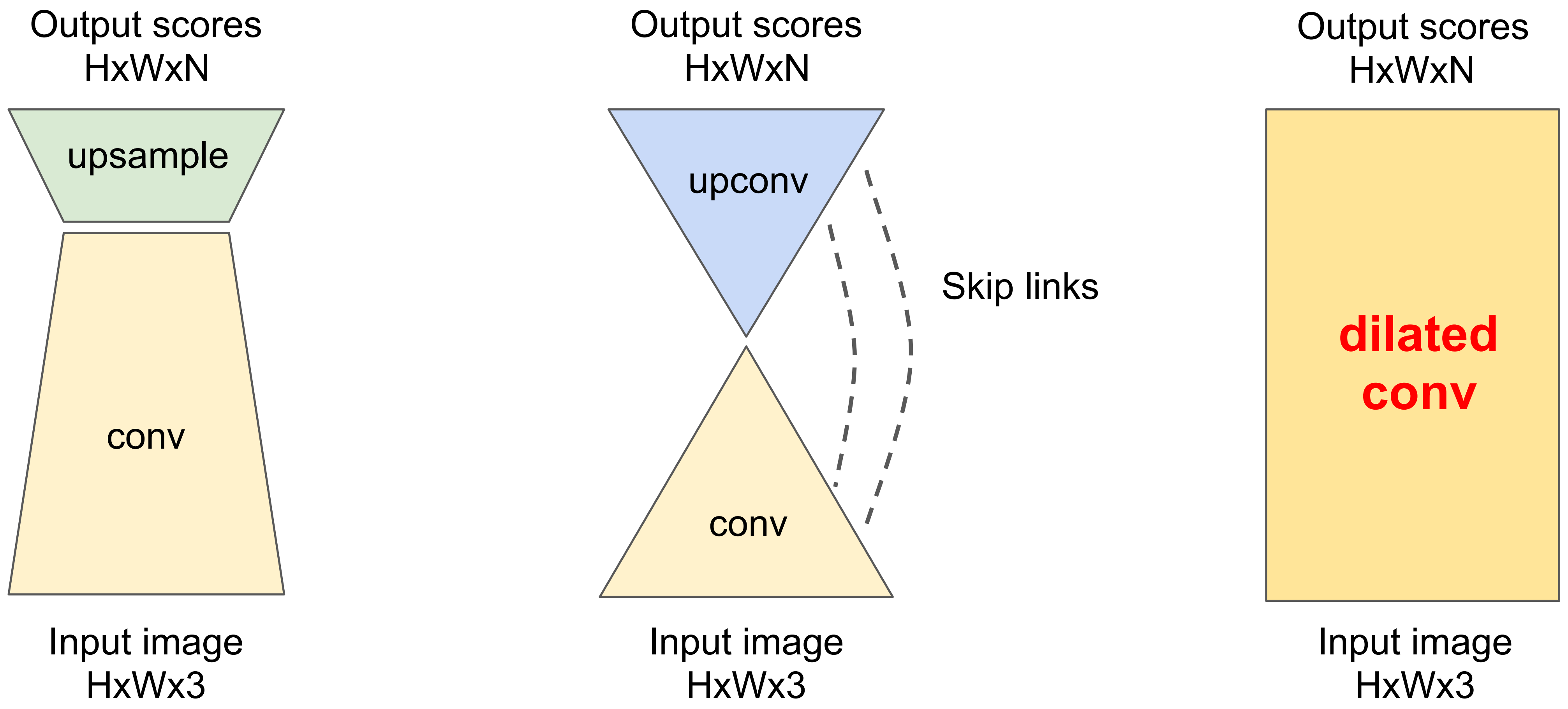
Fully Convolutional Network (FCN)

Segmentation:

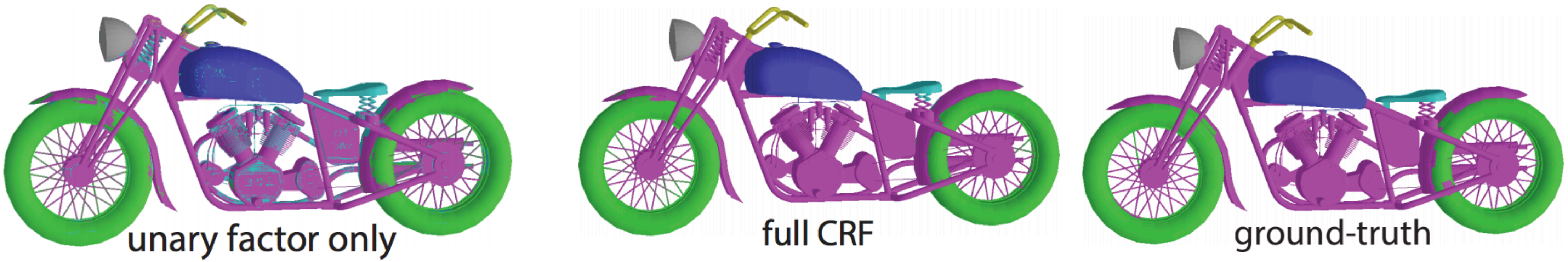


Learning Deconvolution Network for Semantic Segmentation

Fully convolutional network (FCN) variations



Performance



Performance (cont.)

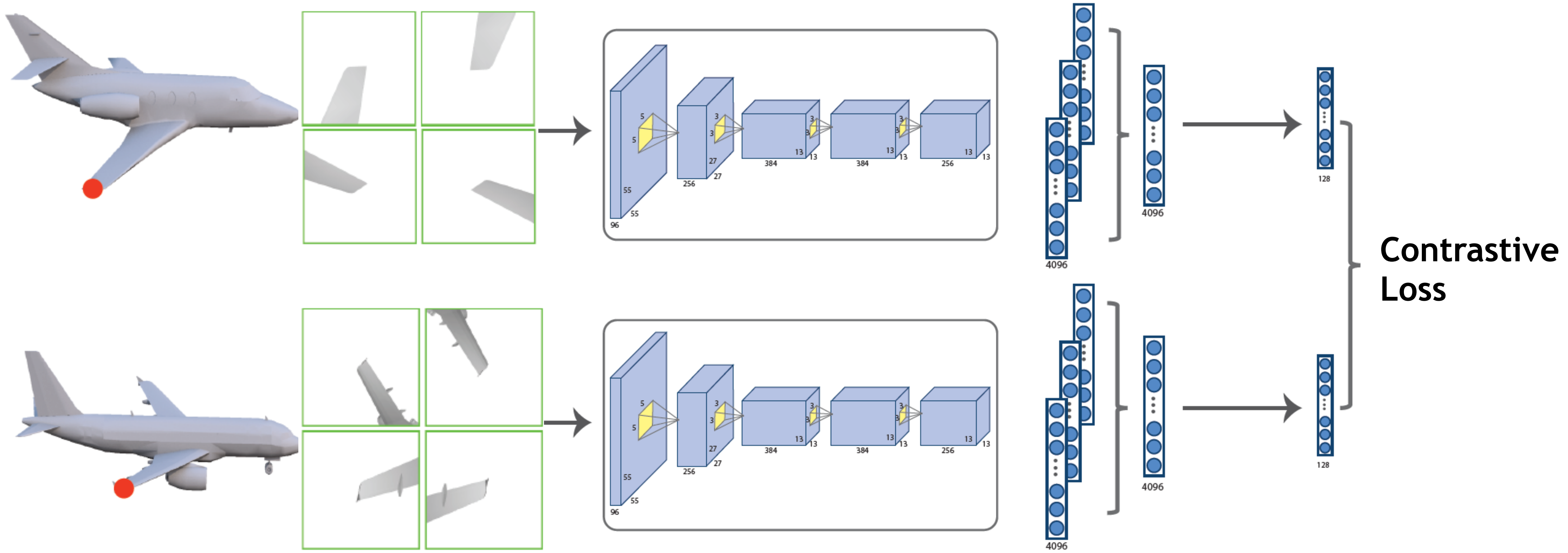
- Viewpoint selection to maximize surface coverage
- Combination of view-based network with surface-based graphical model
- ~88% labeling accuracy on ShapeNet
  (trained per category, 50%-50% split, max 250 shapes for training)

Challenges:

- View-based network does not process invisible points
- View-based representations have redundancy
- Slow to train (~week for a few hundreds of shapes)
- Aggregating view representations via max-pooling may lose information

Surface correspondences with multi-view convnets

Aggregates point-based descriptors across local views. Trained such that similar points have similar descriptors based on synthetically generated correspondences.



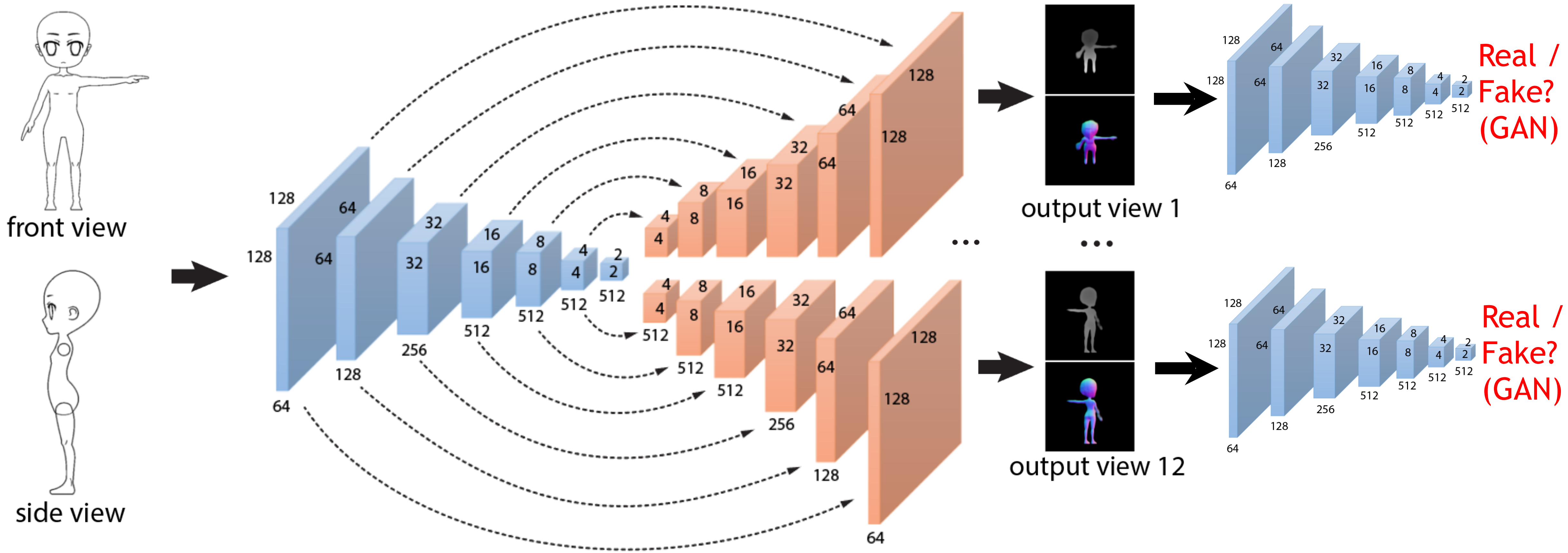
Haibin Huang, Evangelos Kalogerakis, Siddhartha Chaudhuri, Duygu Ceylan, Vladimir Kim, Ersin Yumer
Learning Local Shape Descriptors with View-Based Convolutional Neural Network, ACM TOG (to appear)

Scan-to-shape matching

shows some robustness to noise, better performance than volumetric net (3DMatch)

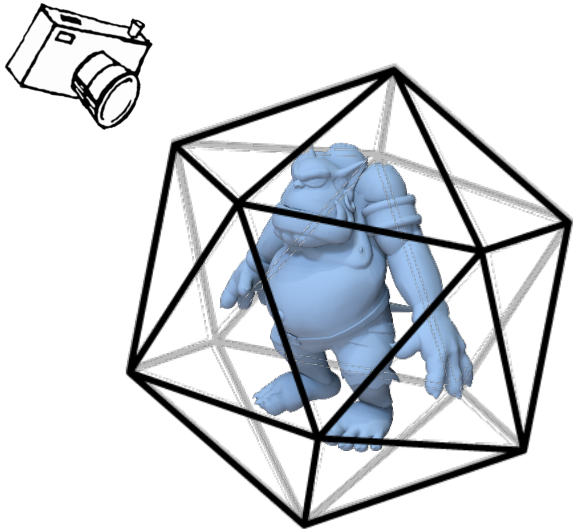
(similar colors correspond to points with similar descriptors)

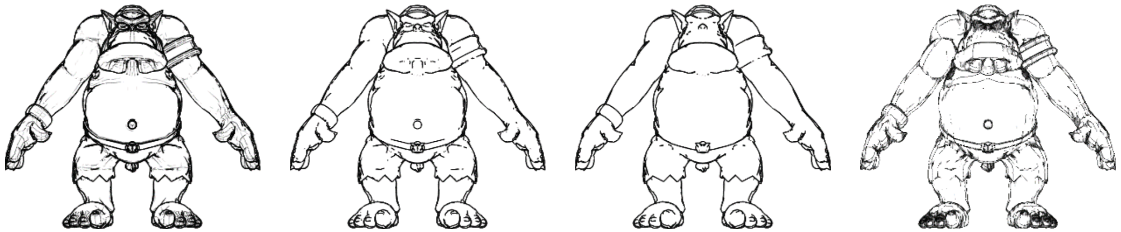
3D reconstruction by multi-view decoder branches (ShapeMVD)



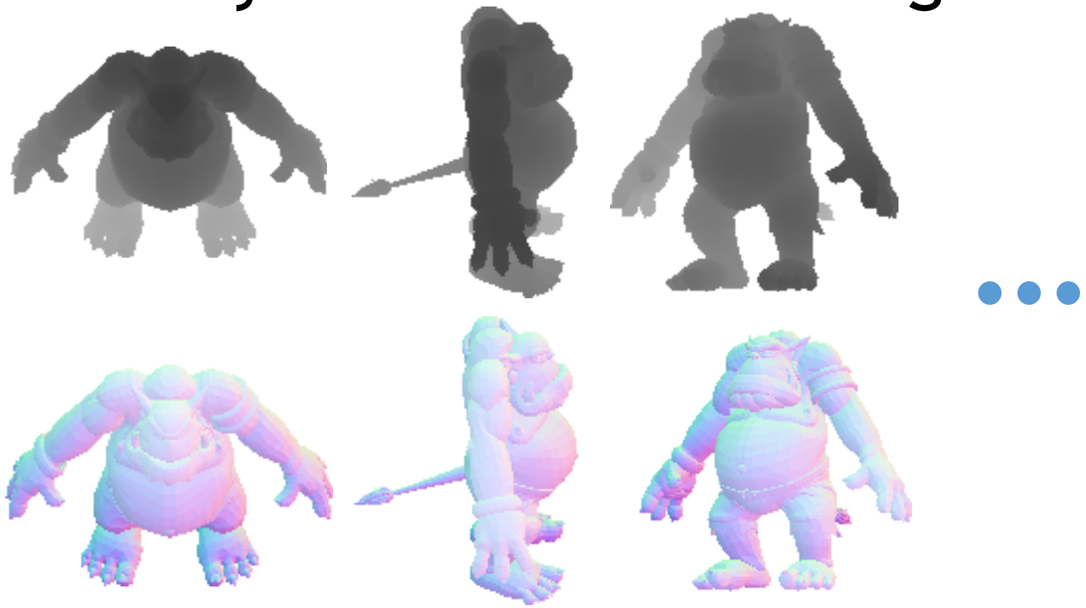
Zhaoliang Lun, Matheus Gadelha, Evangelos Kalogerakis, Subhransu Maji, Rui Wang, “3D Shape Reconstruction from Sketches via Multi-view Convolutional Networks”, arxiv 2017

Training data



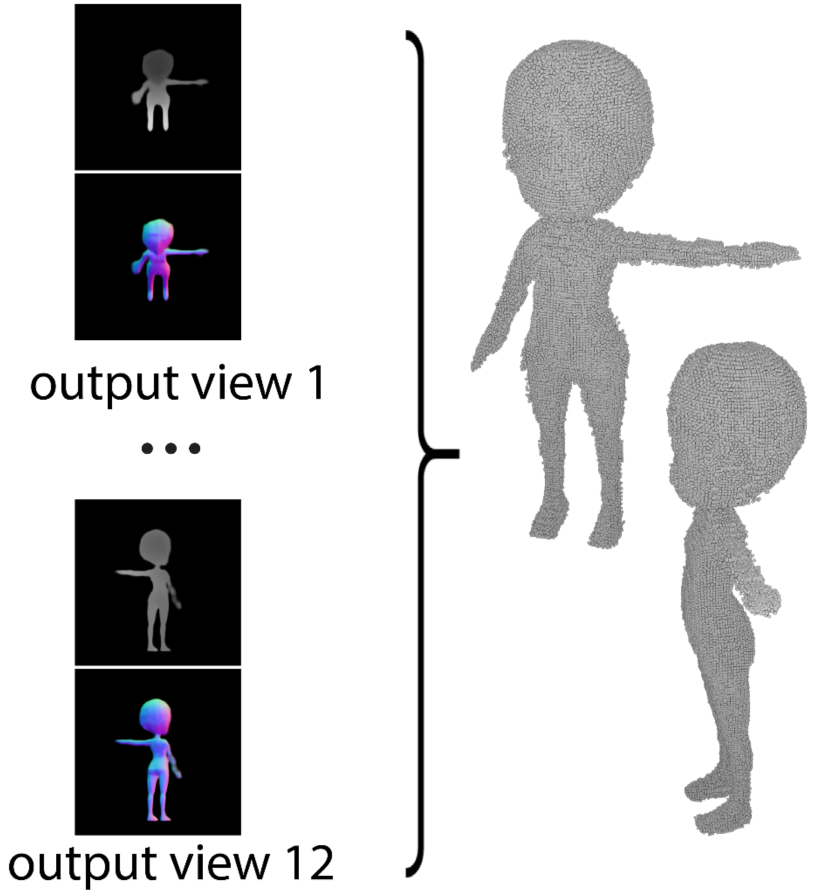


Synthetic line drawings



Training depth and normal maps

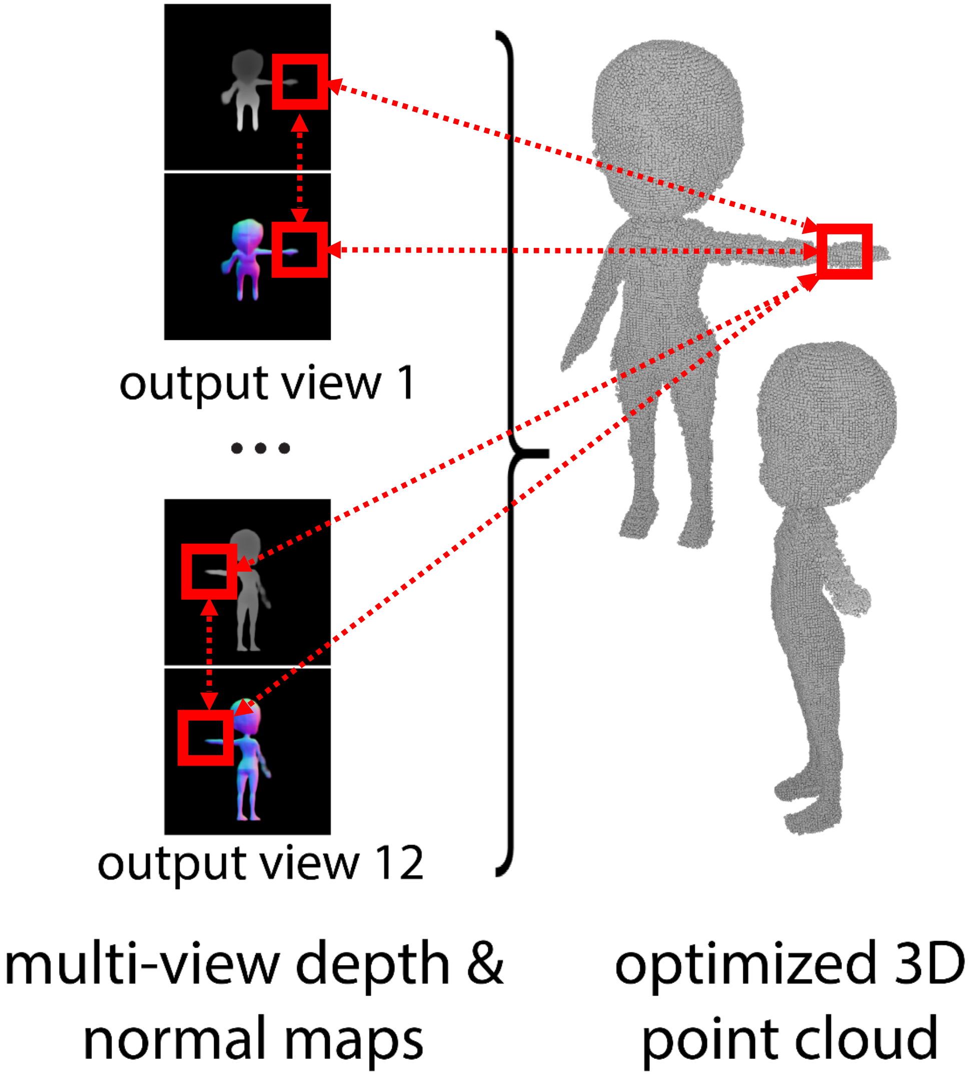
Consolidate multi-view depth and normal maps



multi-view depth & normal maps

optimized 3D point cloud

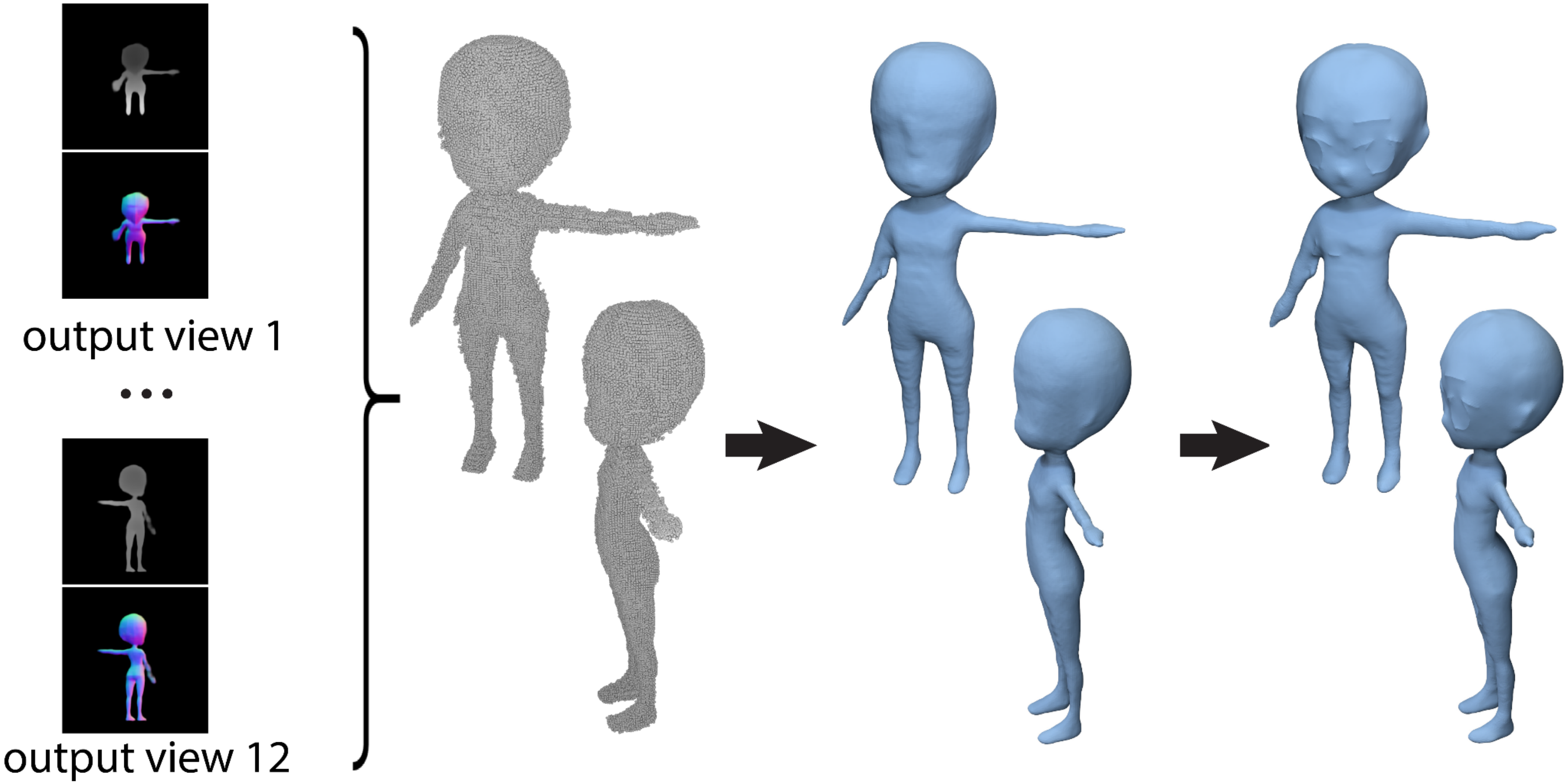
Consolidate multi-view depth and normal maps



Optimization for fusion

- Depth derivatives should be consistent with normals
- Corresponding depths and normals across different views should agree

Poisson surface reconstruction (Kazhdan et al. 2013)

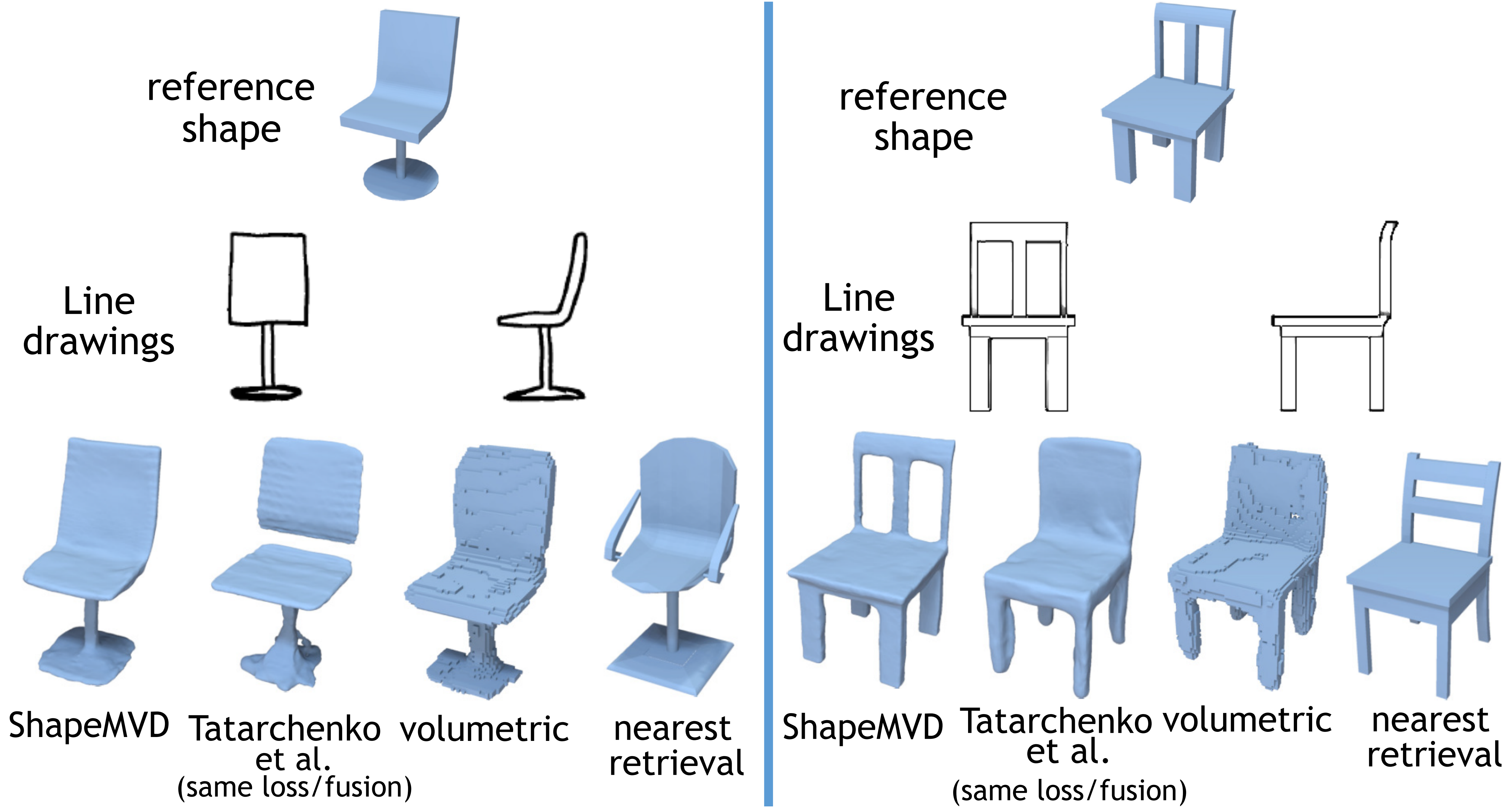


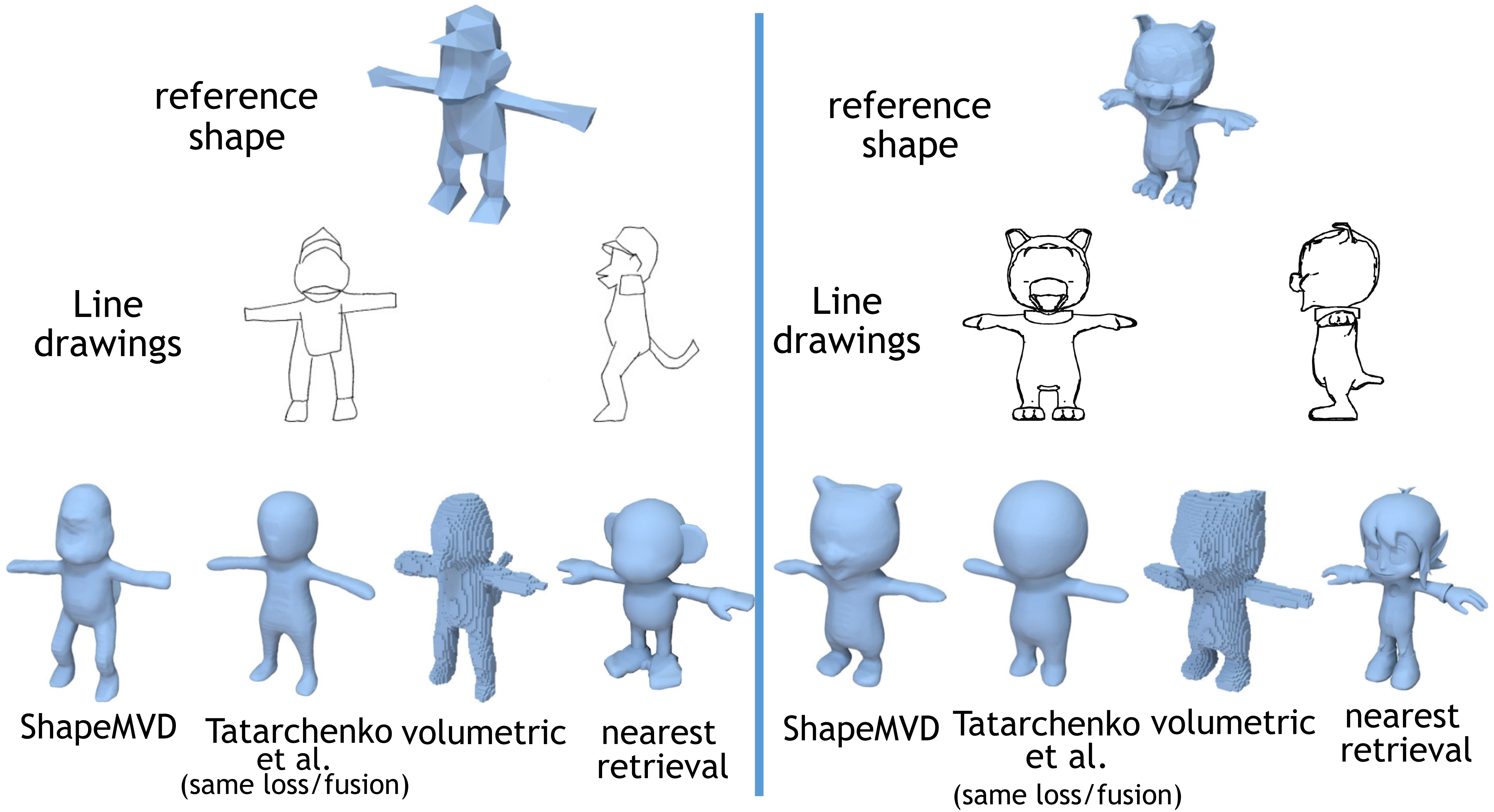
multi-view depth & normal maps

optimized 3D point cloud

surface reconstruction

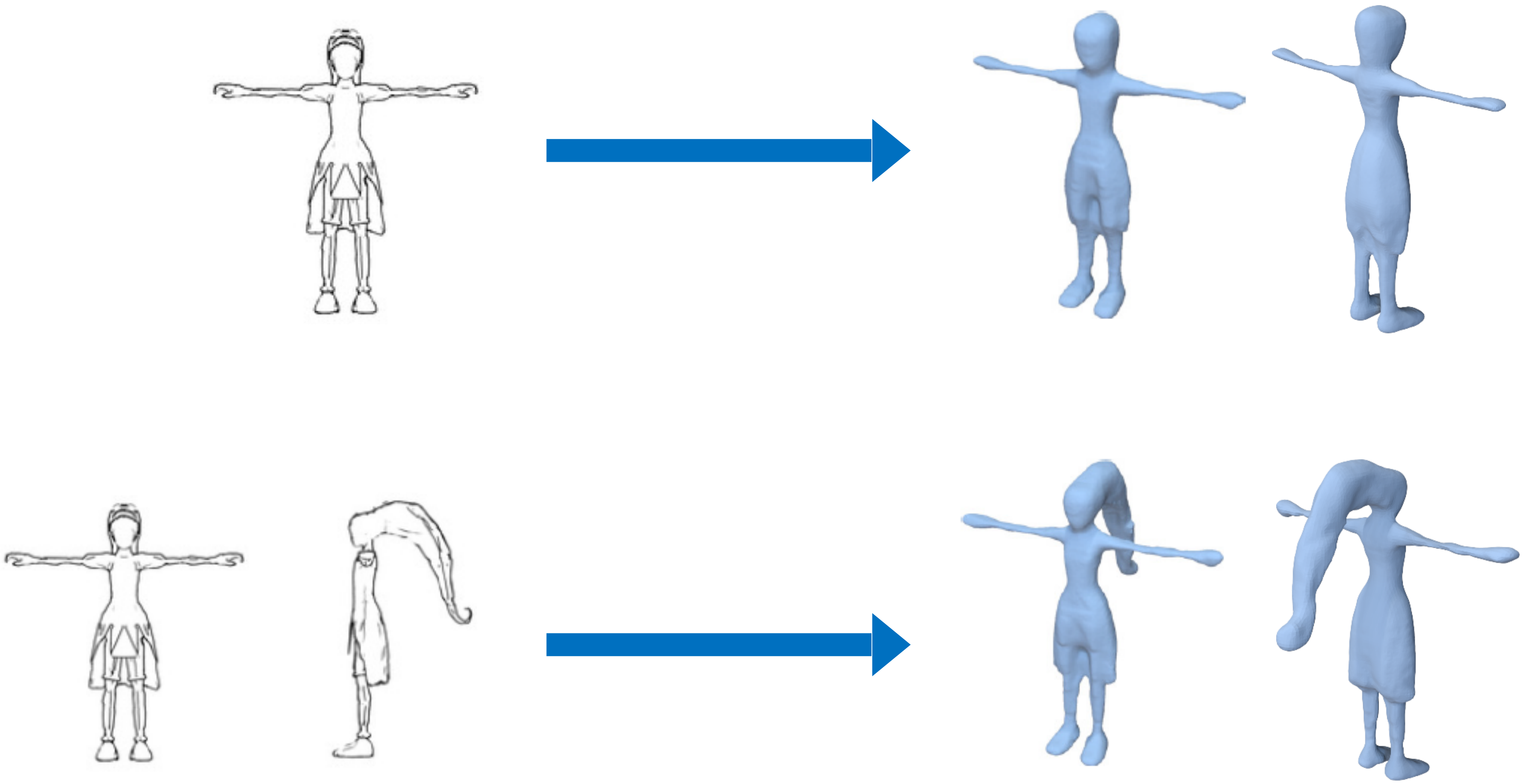
surface fine-tuning


reference shape
Line drawings
ShapeMVD
Tatarchenko et al.
(same loss/fusion)
volumetric
nearest retrieval
reference shape
Line drawings
ShapeMVD
Tatarchenko et al.
(same loss/fusion)
volumetric
nearest retrieval



reference shape
Line drawings
ShapeMVD
Tatarchenko et al.
(same loss/fusion)
volumetric
nearest retrieval
reference shape
Line drawings
ShapeMVD
Tatarchenko et al.
(same loss/fusion)
volumetric
nearest retrieval

Single vs two input line drawings



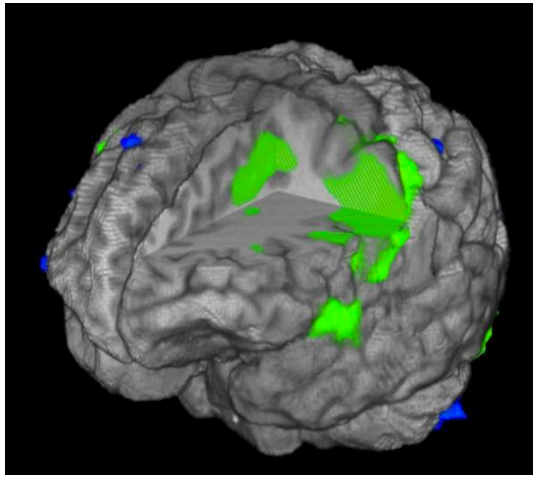
Key challenges for multi-view representation

- Fusing information across viewpoints is not incorporated in the network (not trivial)
- "Cannot see through the surface"
- Less redundancy than producing a surface for every possible continuous viewing angle, yet surfaces across different viewpoints may not be consistent.

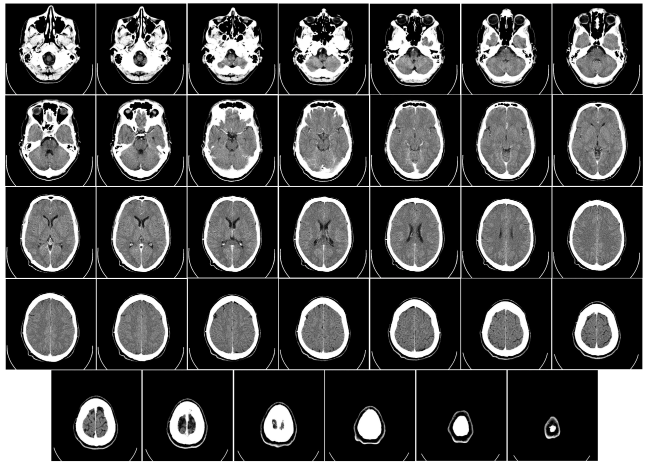
Agenda

- Deep learning on regular structures
  - Multi-view representation
  - **Volumetric representation**
- Deep learning on meshes
- Deep learning on point cloud and parametric models

Popular 3D volumetric data

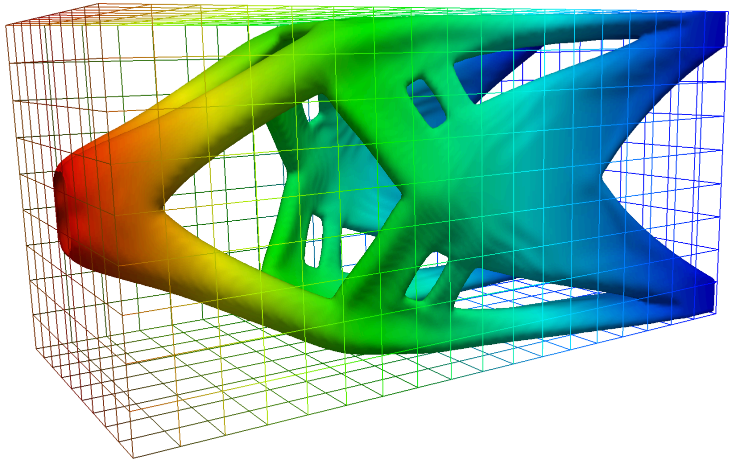


fMRI

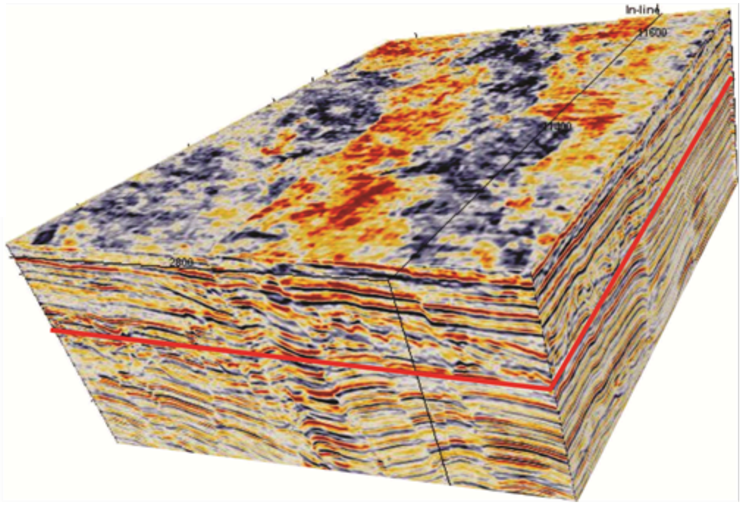


CT

Voxelized
CAD models



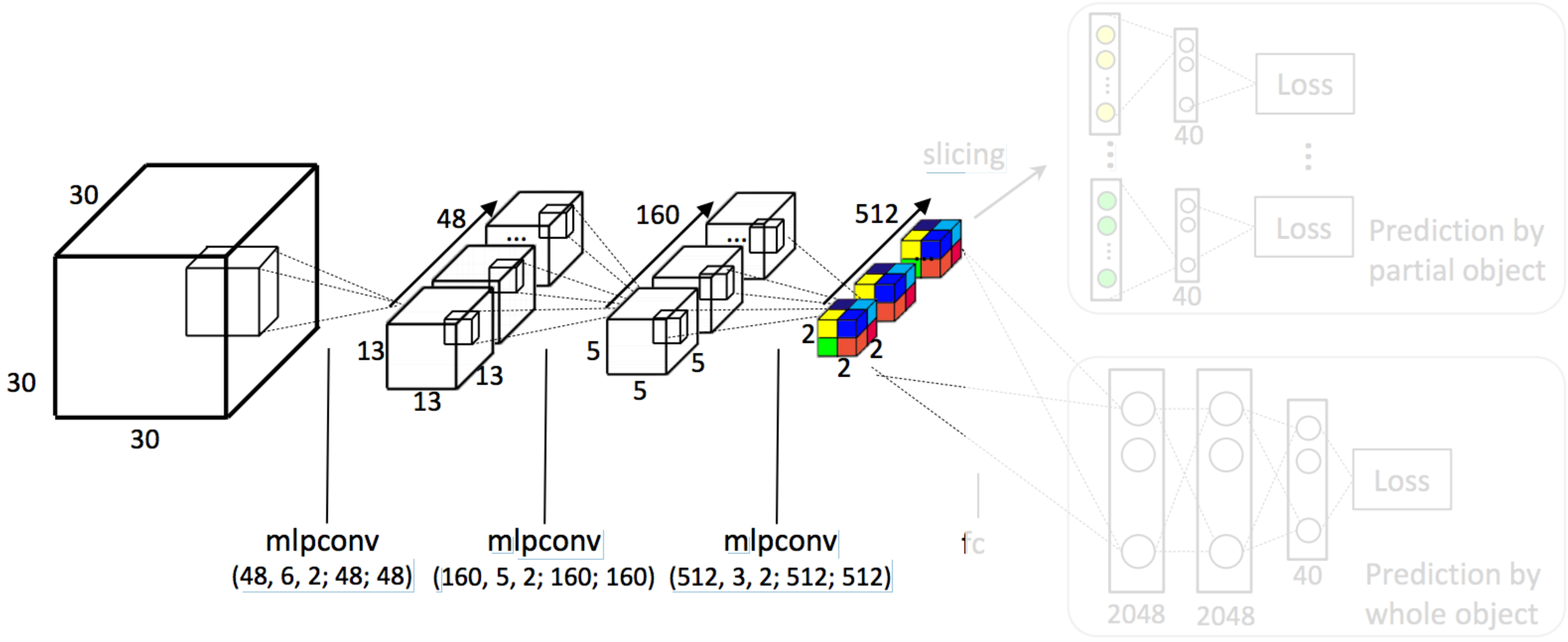
Manufacturing
(finite-element analysis)



Geology

3D CNN on volumetric data

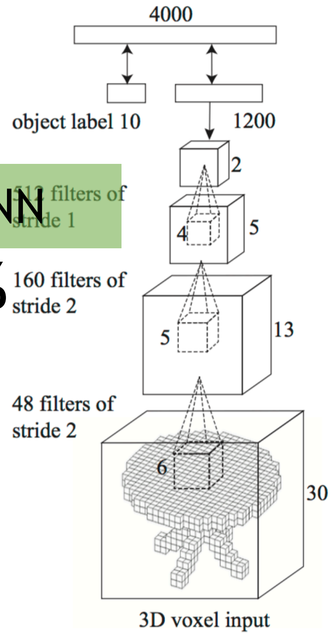
3D convolution uses 4D kernels



[Credit: Su et al. CVPR 2016]

Early 3D CNNs for shape classification

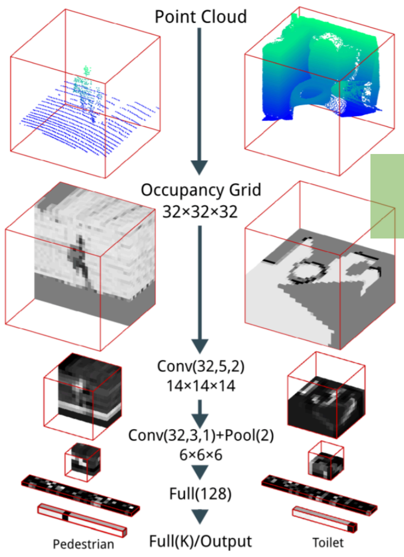
3DShapeNets from Princeton
CVPR 2015



3D CNN

77.3%

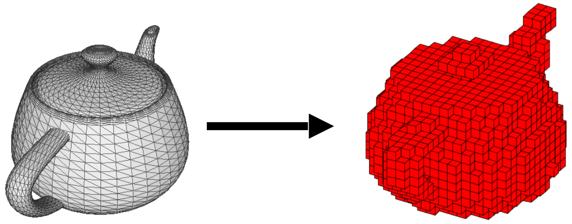
VoxNet from CMU Robotics
IEEE/RSJ 2015



3D CNN

83.0%

Information loss in voxelization



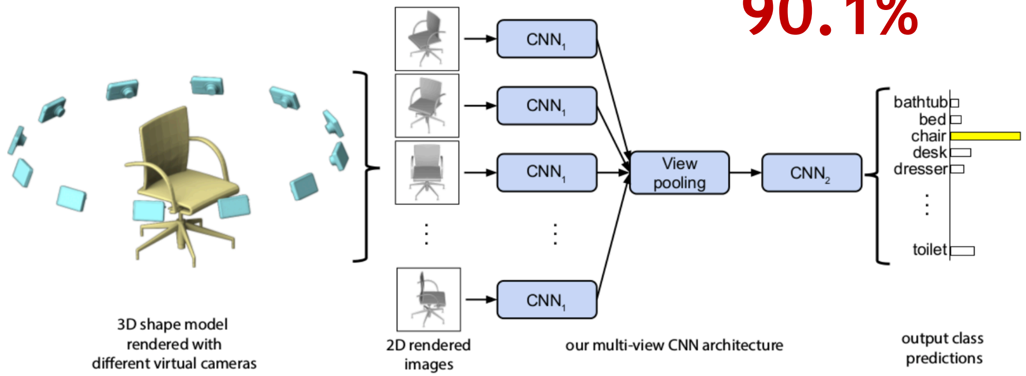
CAD model

Occupancy Grid
30x30x30

Rendering +
2D CNN

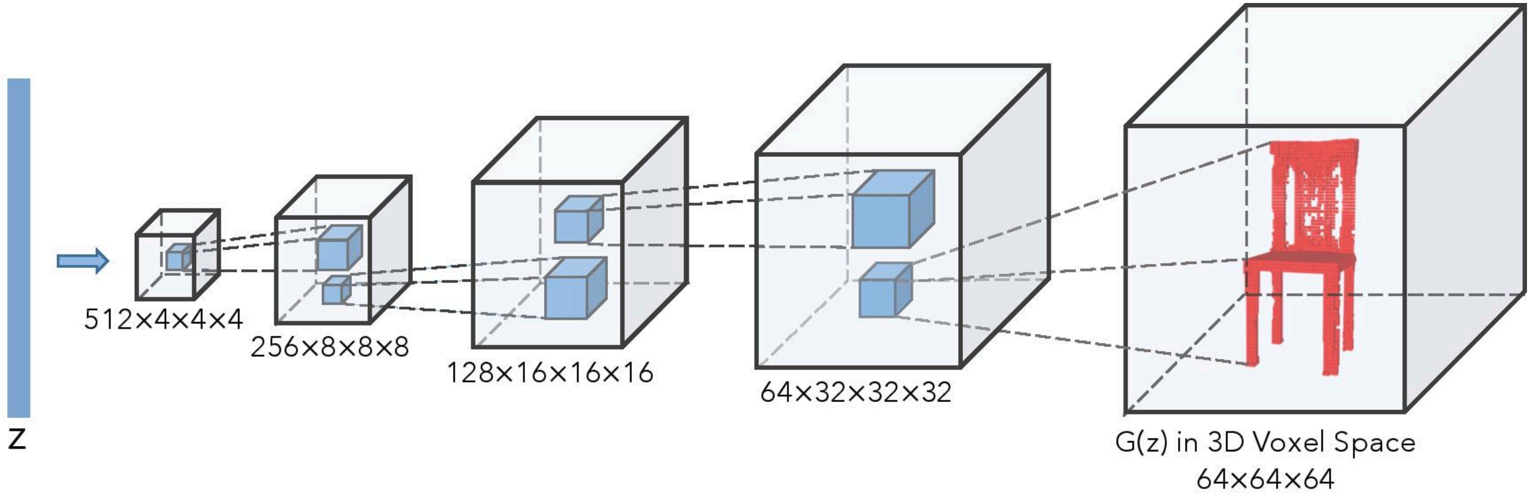
90.1%

MVCNN from UMass
ICCV 2015



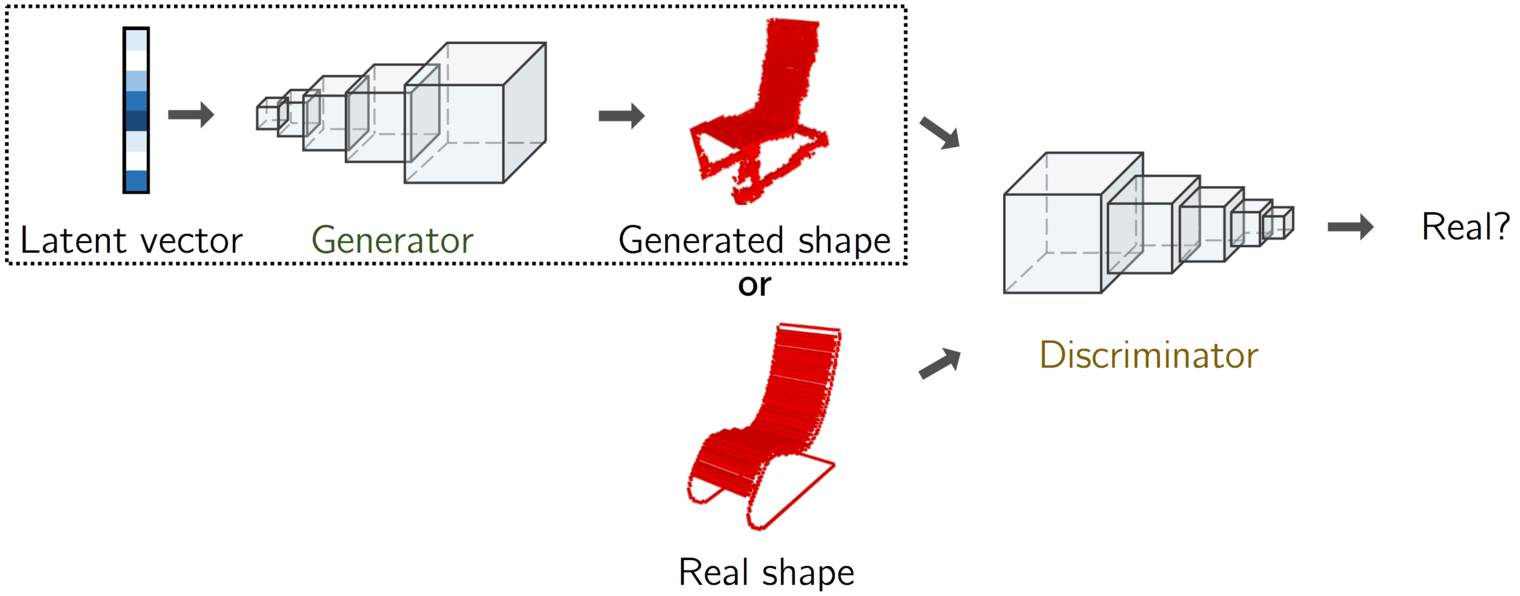
3D CNN for volumetric data

3D deconvolution uses 4D kernels



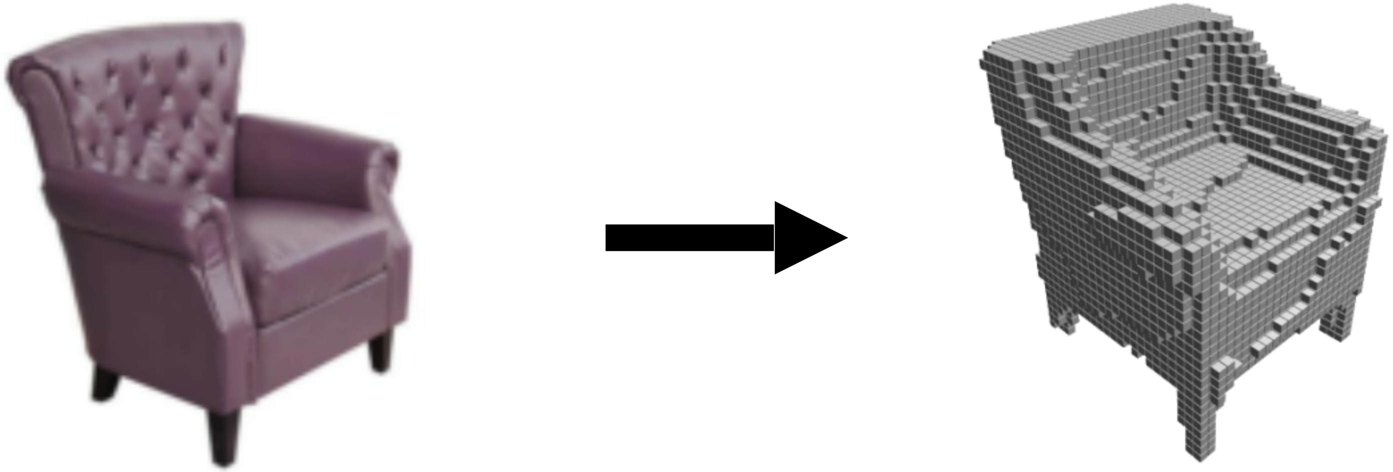
[Credit: Wu&Zhang, et al. NIPS 2016]

Volumetric Generative Adversarial Networks



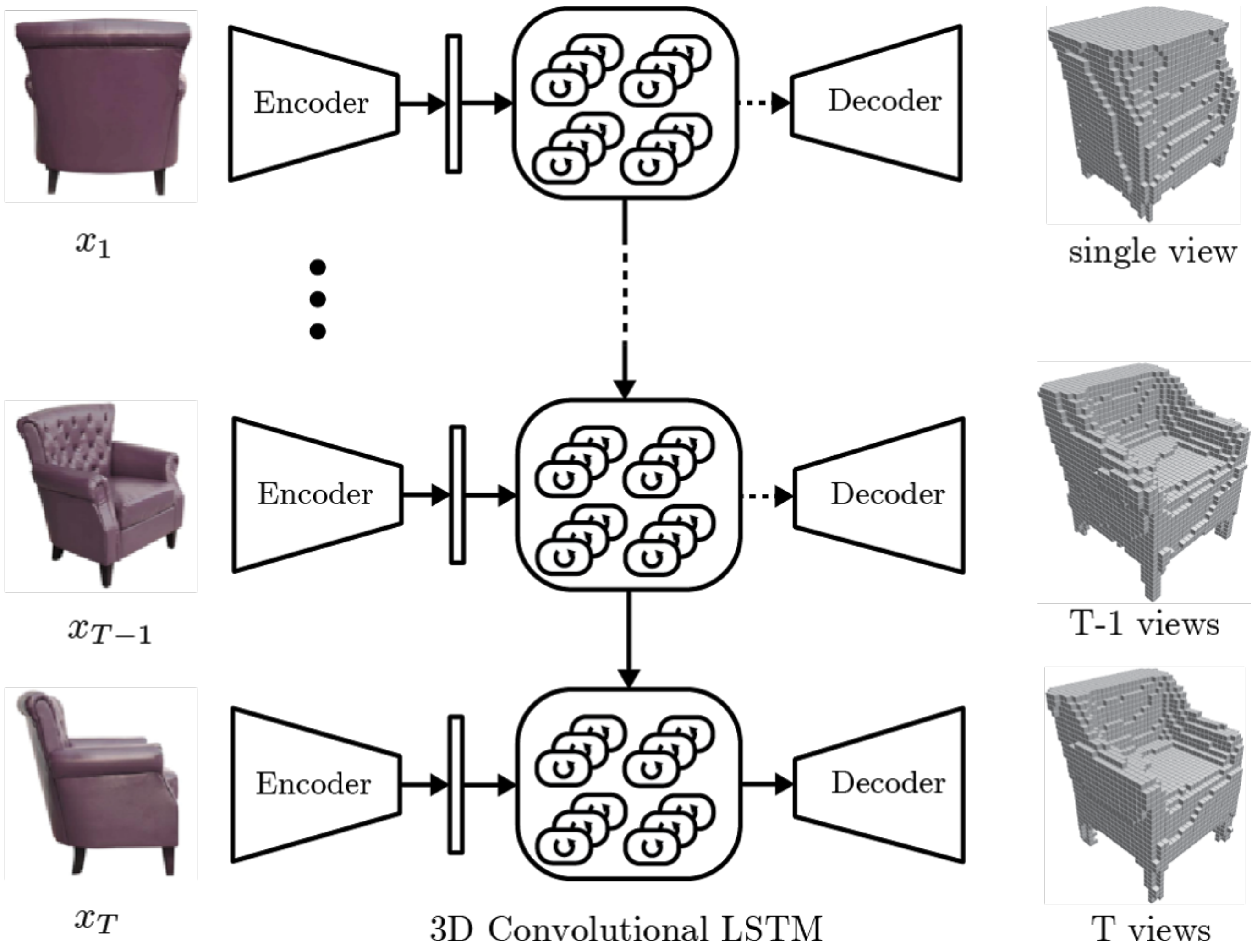
Jiajun Wu, Chengkai Zhang, et al. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. NIPS 2016

Learning 3D reconstruction from single-view



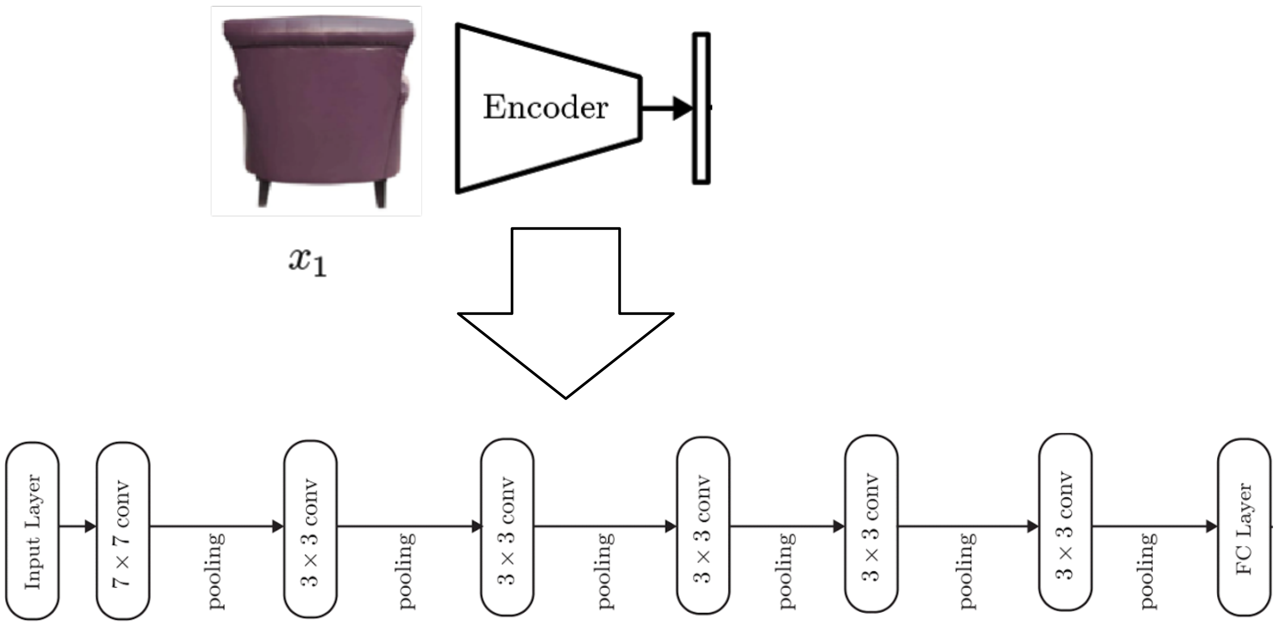
- Depth based methods [Eigen et al., Saxena et al., etc]
- Model based methods [Su et al., Kar et al., Aubry et al., Choy et al., etc]

Recurrent 2D-3D CNN for volumetric reconstruction

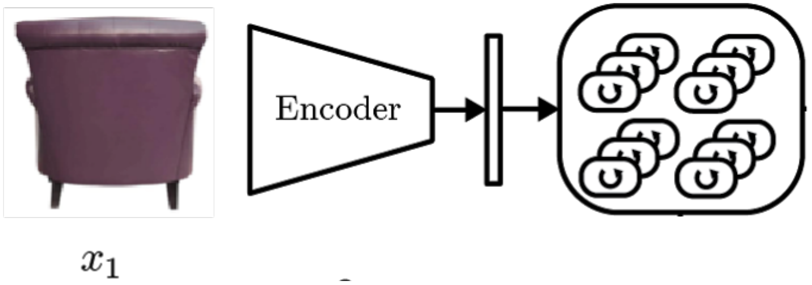


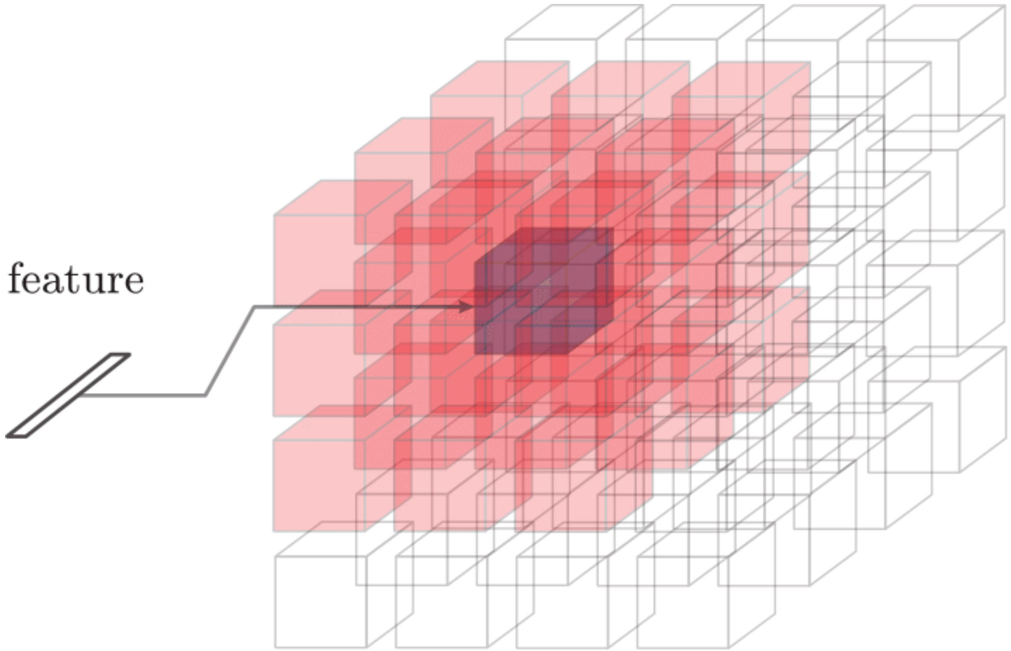
Christopher B. Choy, Danfei Xu*, JunYoung Gwak*, Kevin Chen, Silvio Savarese,
3D-R^2N^2: A unified approach for single and multi-view 3D object reconstruction
ECCV2016

Recurrent 3D CNN



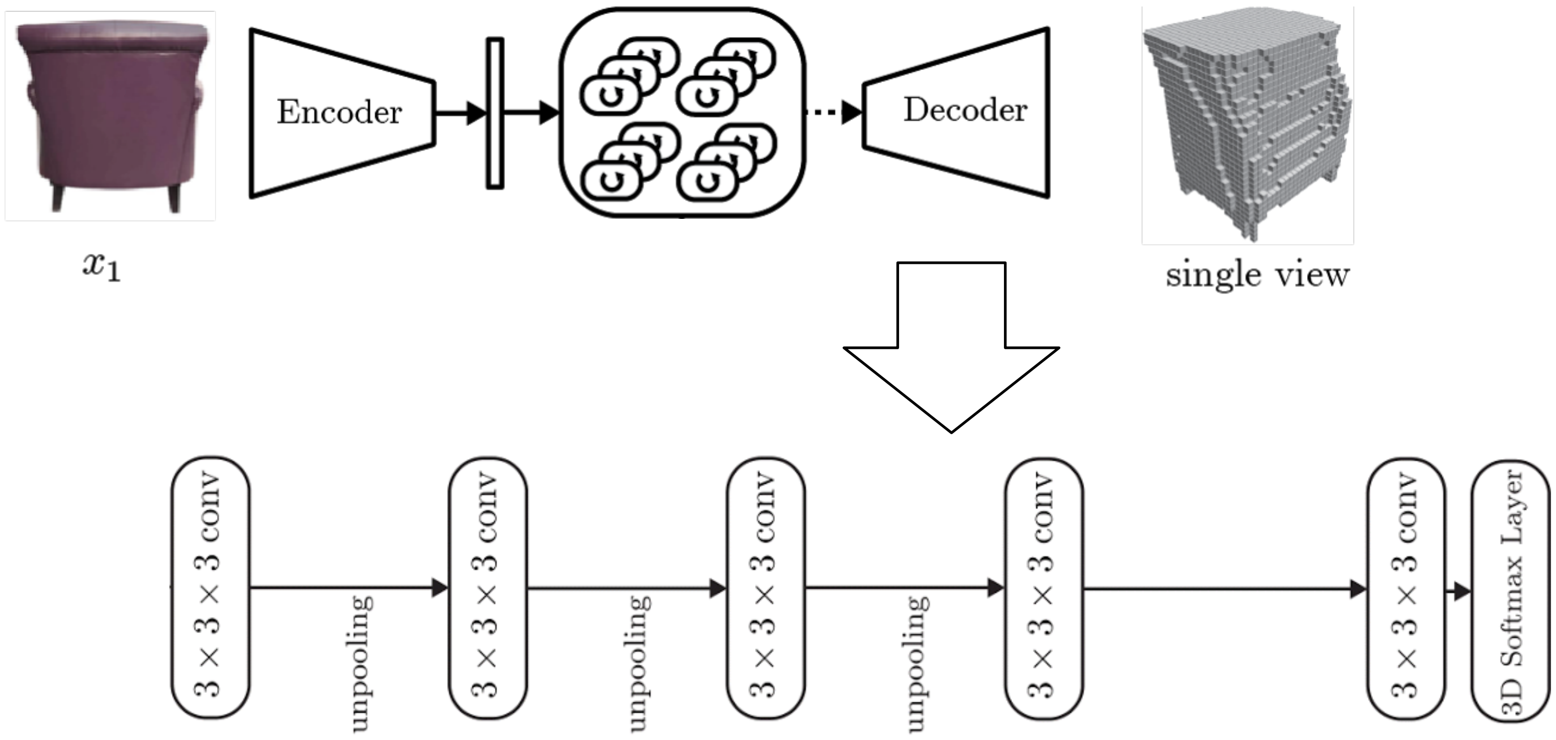
Recurrent 3D CNN





3D Convolutional LSTM

Recurrent 3D CNN



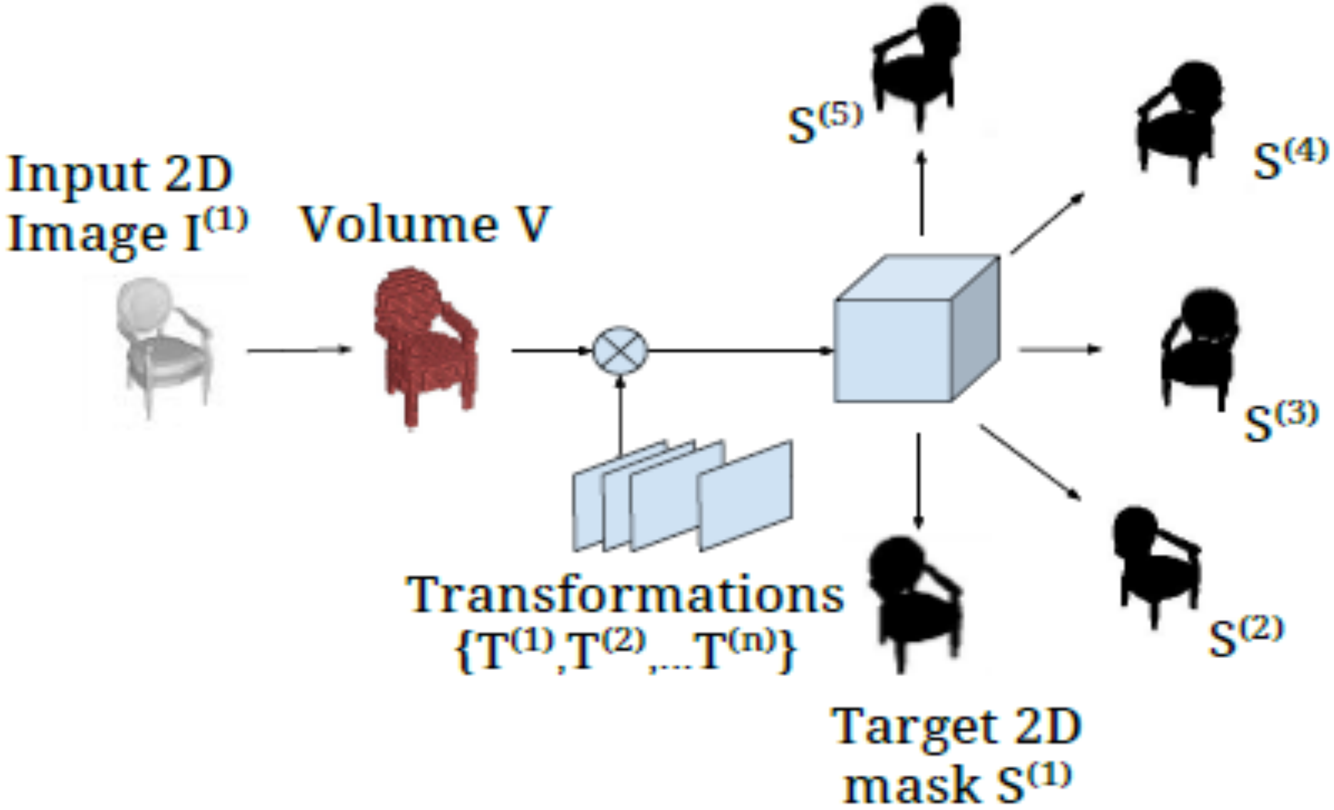
Supervised learning with ground truth 3D volumes

- Voxel-wise cross entropy loss

$$L(\mathcal{X}, y) = \sum_{i,j,k} y_{(i,j,k)} \log(p_{(i,j,k)}) + (1 - y_{(i,j,k)}) \log(1 - p_{(i,j,k)})$$

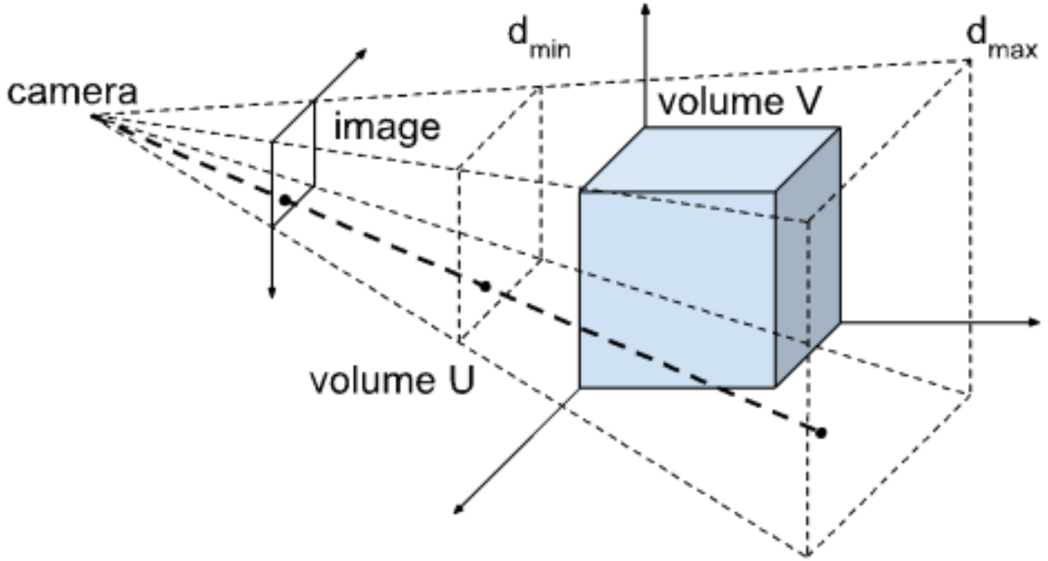
- ShapeNet
    - 50k CAD models
    - Render from arbitrary views
    - Random number of images w/ random order
    - Random background, translation

Learning volumetric reconstruction by multi-view supervision



Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision
Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, Honglak Lee, NIPS 2016

Perspective Transformer layer (Projecting 3D volume to 2D masks)



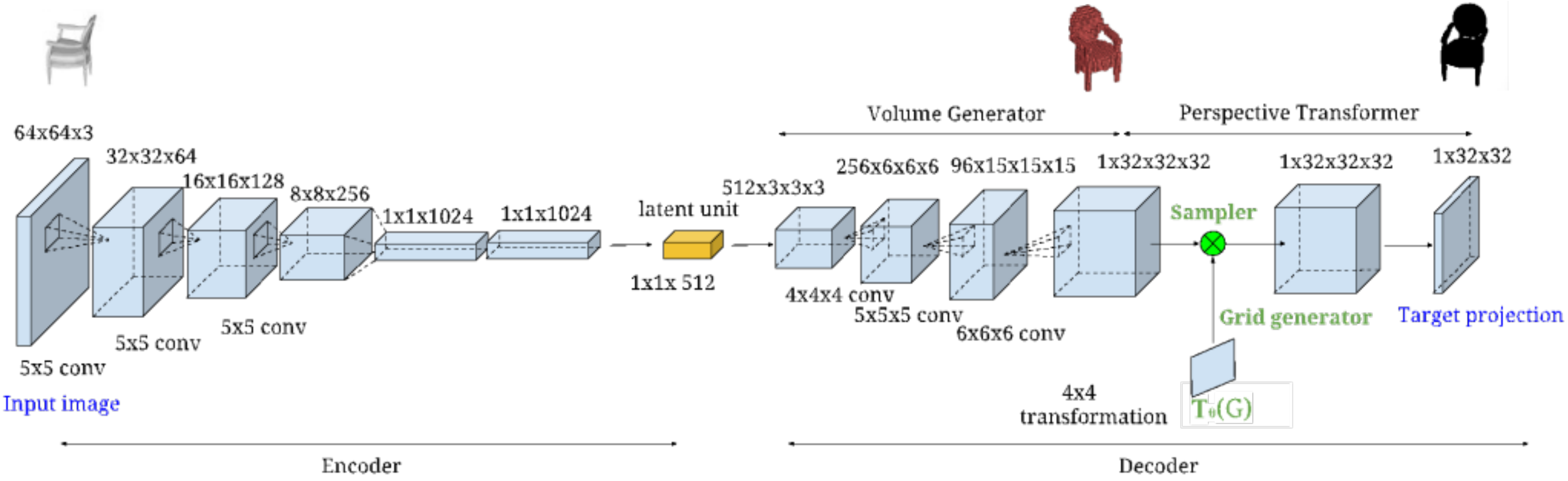
$$U_i = \sum_n^H \sum_m^W \sum_l^D V_{nml} \max(0, 1 - |x_i^s - m|) \max(0, 1 - |y_i^s - n|) \max(0, 1 - |z_i^s - l|)$$

$$S_{n'm'} = \max_{l'} U_{n'm'l'}$$

For each pixel on a mask, find the intersection of its corresponding ray and the input volume

1. Sample points p = [x, y, 1, d] given the range of disparity d in [d_min, d_max]
    1. p = [x/d, y/d, 1/d, 1]
2. Given a perspective transform matrix T, generate sampling points on the input volume V by q = T^-1 p (ray sampling)
3. Generate the output volume U by bilinear sampling on the input volume V
4. Generate the mask S by max pooling over the depth dimension on U

Perspective Transformer Nets

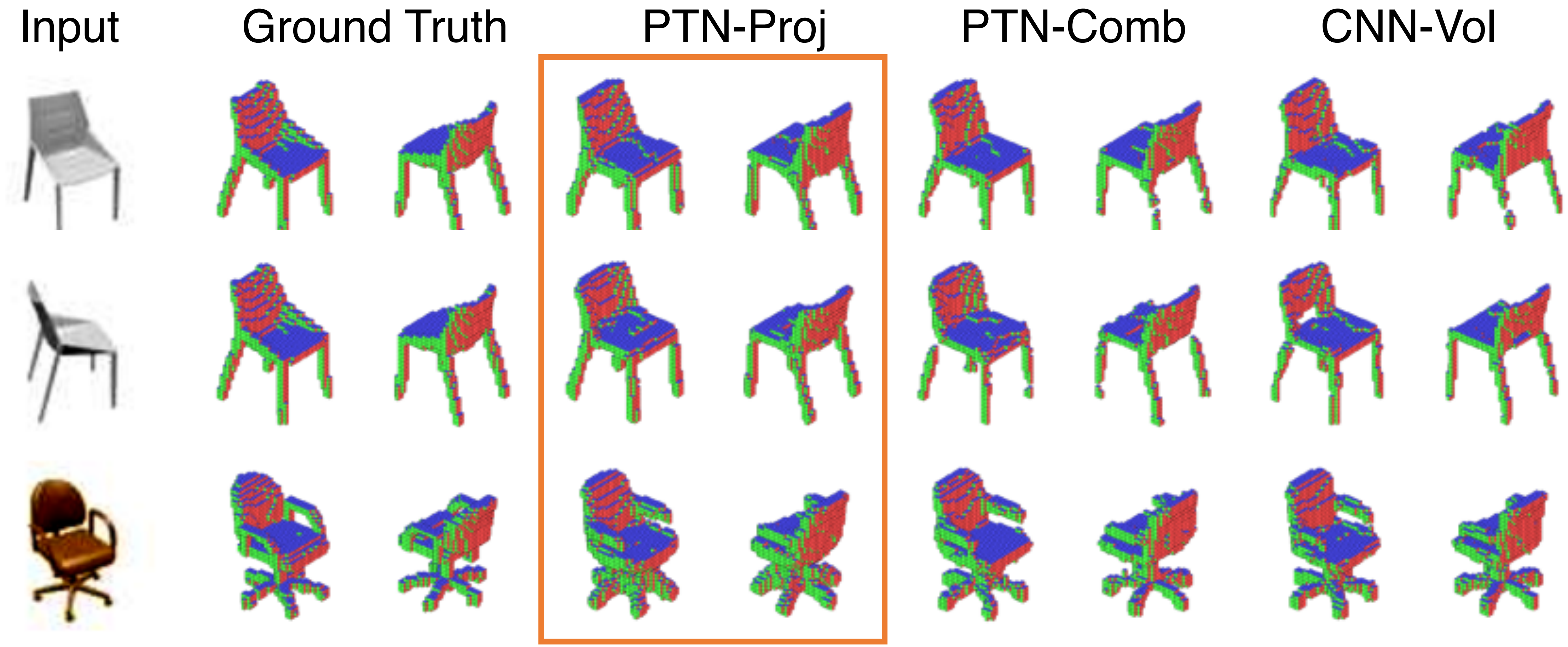


$$\mathcal{L}_{comb}(I^{(k)}) = \lambda_{proj}\mathcal{L}_{proj}(I^{(k)}) + \lambda_{vol}\mathcal{L}_{vol}(I^{(k)})$$

$$\mathcal{L}_{vol}(I^{(k)}) = ||f(I^{(k)}) - \mathbf{V}||_2^2$$

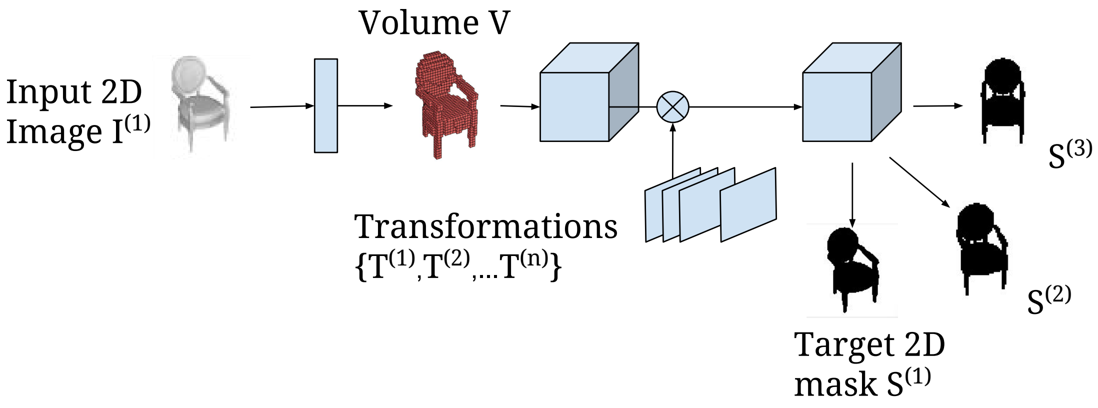
$$\mathcal{L}_{proj}(I^{(k)}) = \sum_{j=1}^{n} \mathcal{L}_{proj}^{(j)}(I^{(k)}; S^{(j)}, \alpha^{(j)}) = \frac{1}{n}\sum_{j=1}^{n} ||P(f(I^{(k)}); \alpha^{(j)}) - S^{(j)}||_2^2$$

Results

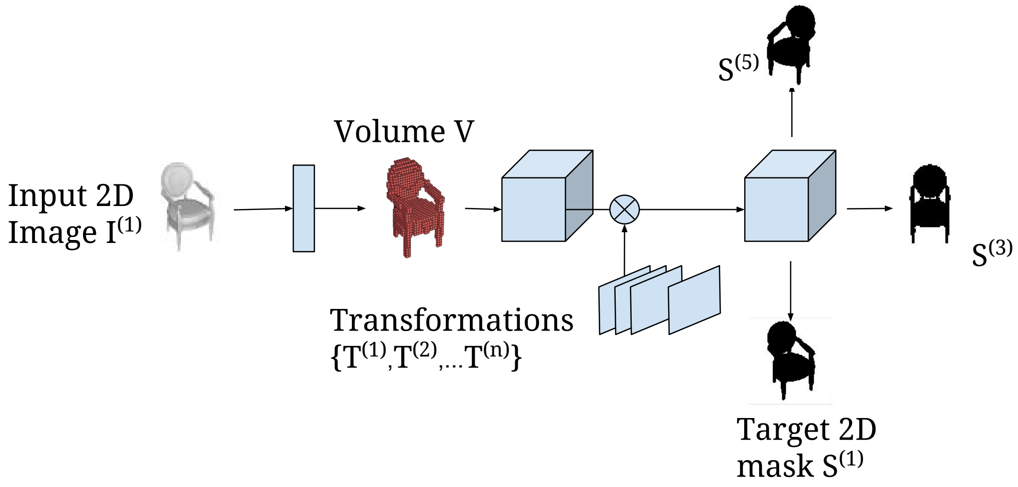


Learning from partial views

A Narrow range of views

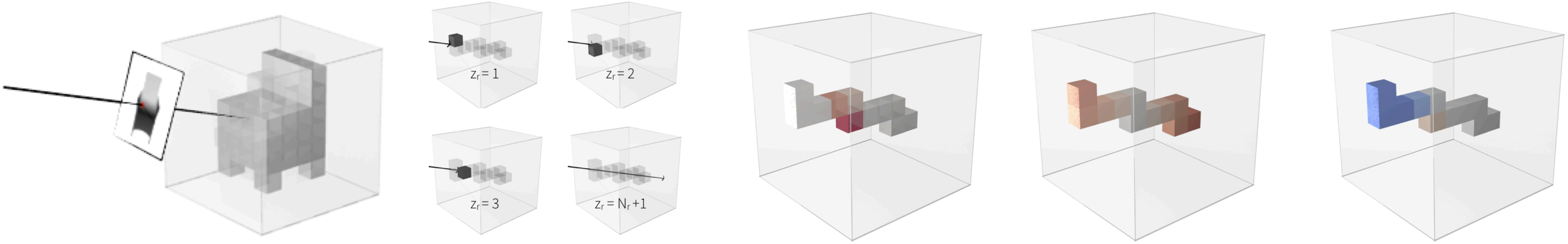


A set of Sparsely sampled views



Method / Evaluation Set	chair		chair-N		chair-S	
	training	test	training	test	training	testing
PTN-Proj:single	0.5712	0.5027	0.4882	**0.4583**	0.5201	**0.4869**
PTN-Comb:single	**0.6435**	**0.5067**	**0.5564**	0.4429	**0.6037**	0.4682
CNN-Vol:single	0.6390	0.4983	0.5518	0.4380	0.5712	0.4646
NN search (vol. supervision)	—	0.3557	—	0.3073	—	0.3084
	24 views (360 degree)		8 views (90 degree)		8 views (evenly sampled)	

Differentiable ray consistency

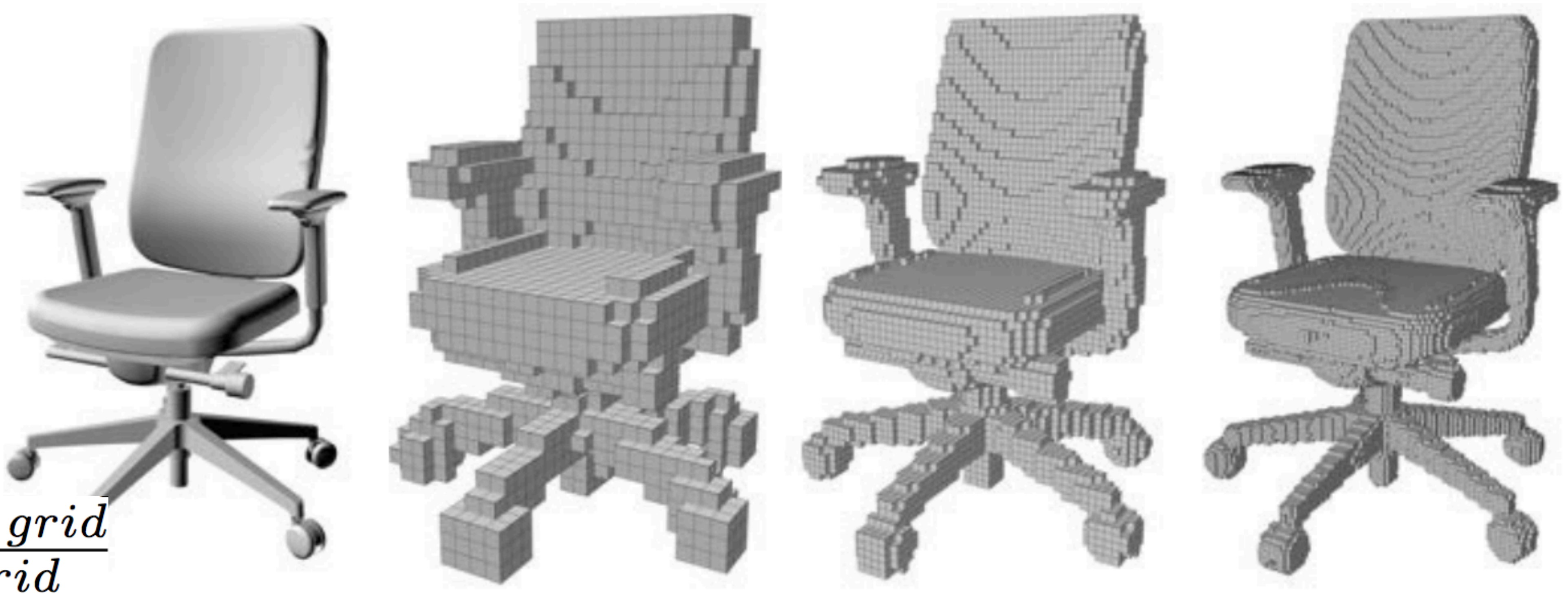


a) Observation Image and Predicted Shape
b) Ray Termination Events
c) Event Probabilities
d) Event Costs
e) Gradients

1. Given a pair of observation and camera, trace the voxels for each pixel along the ray (Nearest neighbor sampling)
2. Define ray termination probability to determine the relationship between a pixel and voxel occupancy likelihood (Differentiable)
3. Different types of multi-view observations e.g. foreground masks, depth, color images semantics etc. as supervision.

Multi-view Supervision for Single-view Reconstruction via Differentiable Ray Consistency. S. Tulsiani, T. Zhou, A. A. Efros, J. Malik. In CVPR, 2017

The sparsity characteristic of volumetric data



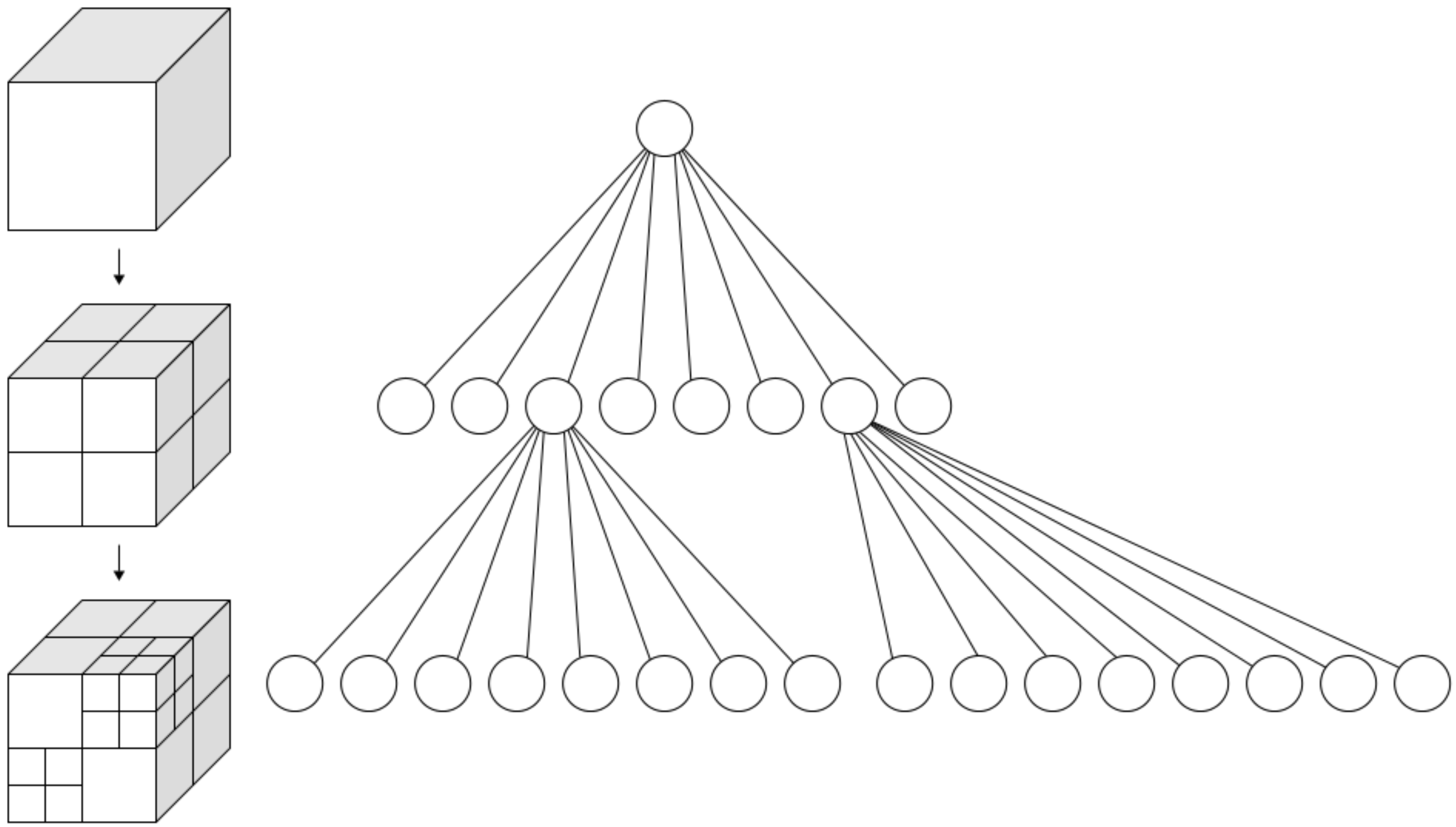
$$\frac{\#occupied\ grid}{\#total\ grid}$$

Occupancy:	10.41%	5.09%	2.41%
Resolution:	32	64	128

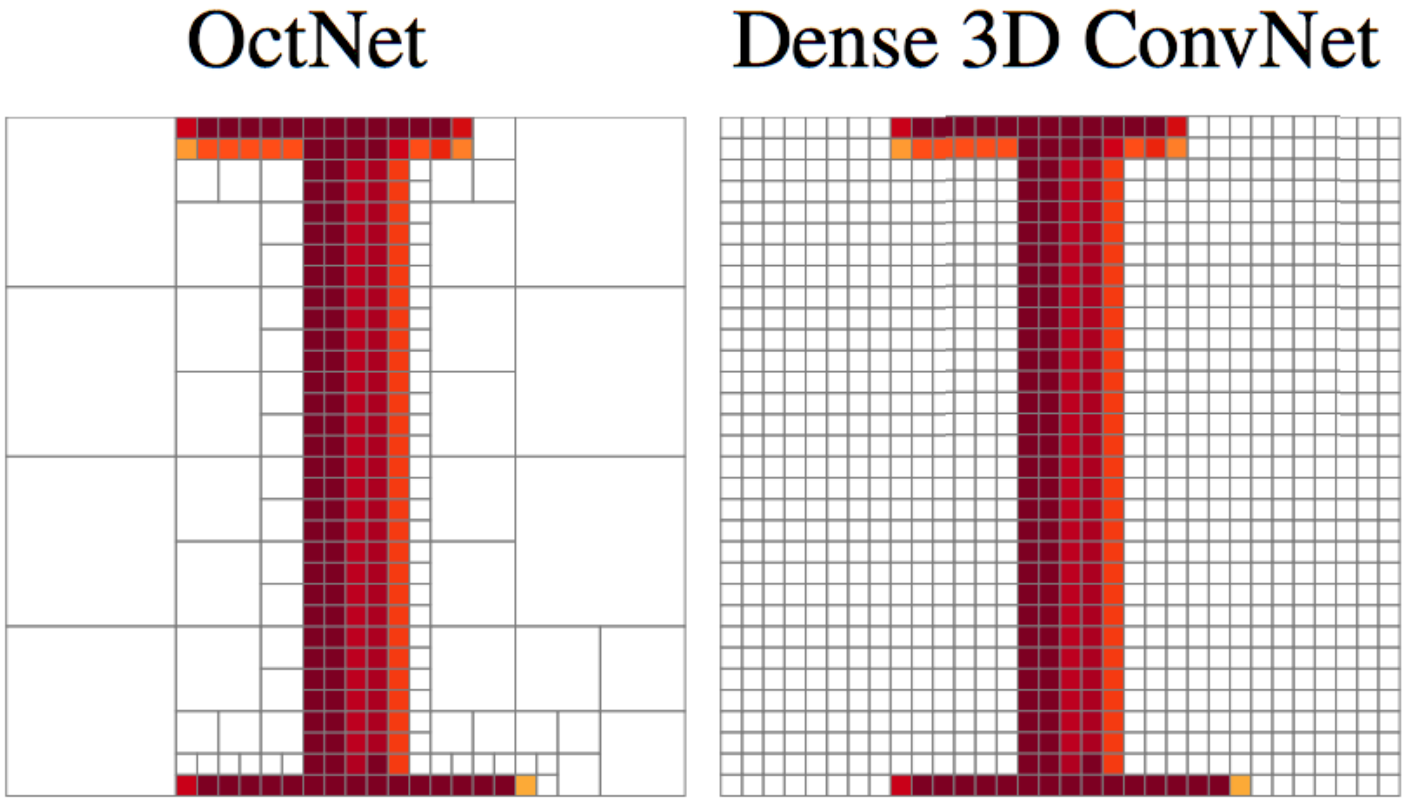
Store only the occupied grids

Octree: recursively partition the space

Each internal node has exactly eight children



Skip the computation of empty cells

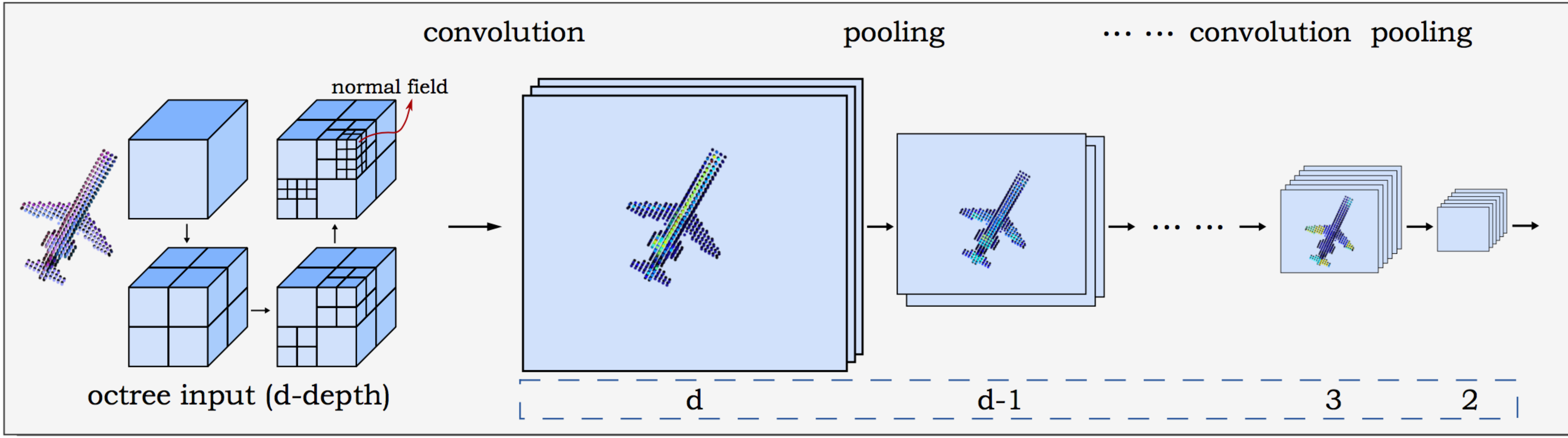


Gernot Riegler, Ali Osman Ulusoy, Andreas Geiger
"OctNet: Learning Deep 3D Representations at High Resolutions"
CVPR2017

Pengshuai Wang, Yang Liu, Yuxiao Guo, Chunyu Sun, Xin Tong
"O-CNN: Octree-based Convolutional Neural Network for Understanding 3D Shapes"
SIGGRAPH2017

Octree-based Convolutional Neural Network

Define convolution and pooling along the octree



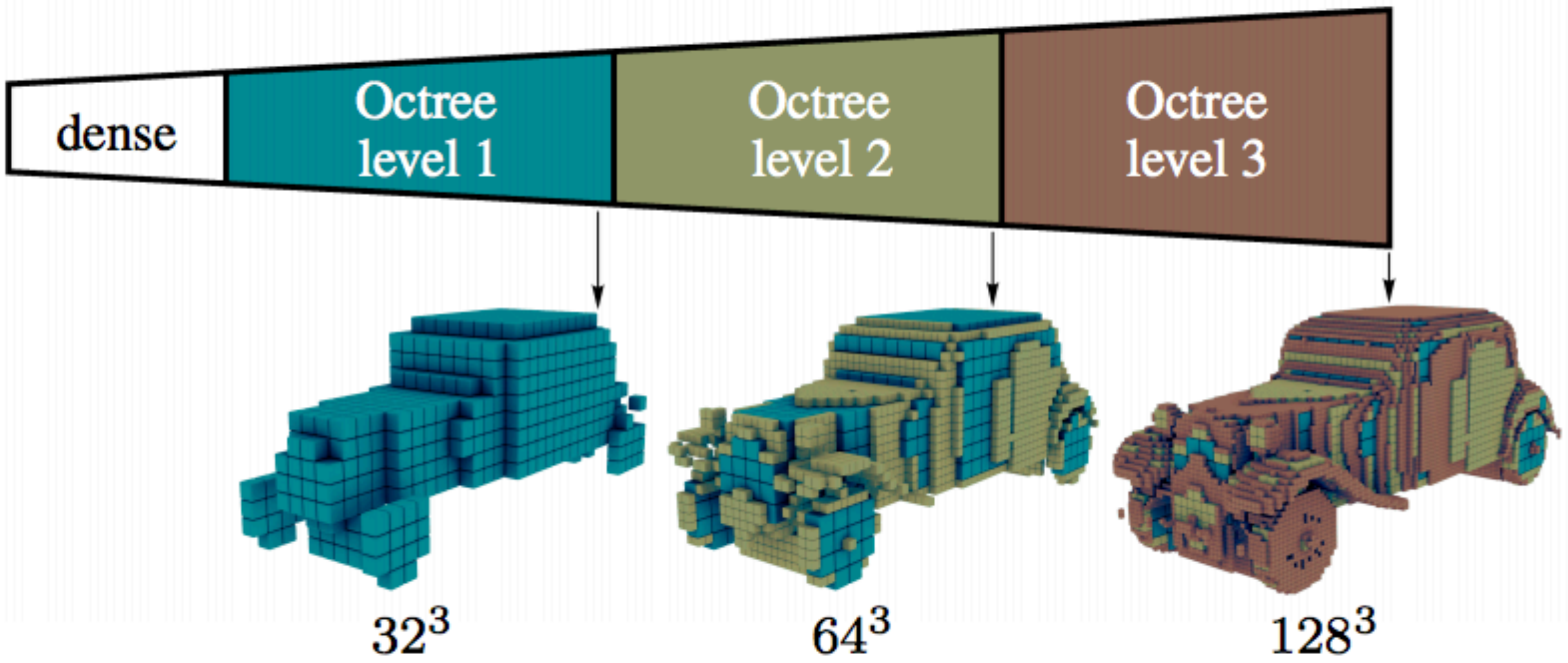
The challenge is how to implement efficiently — build a hash table to index the neighborhood

Restrict the convolution stride to be 2

Performance

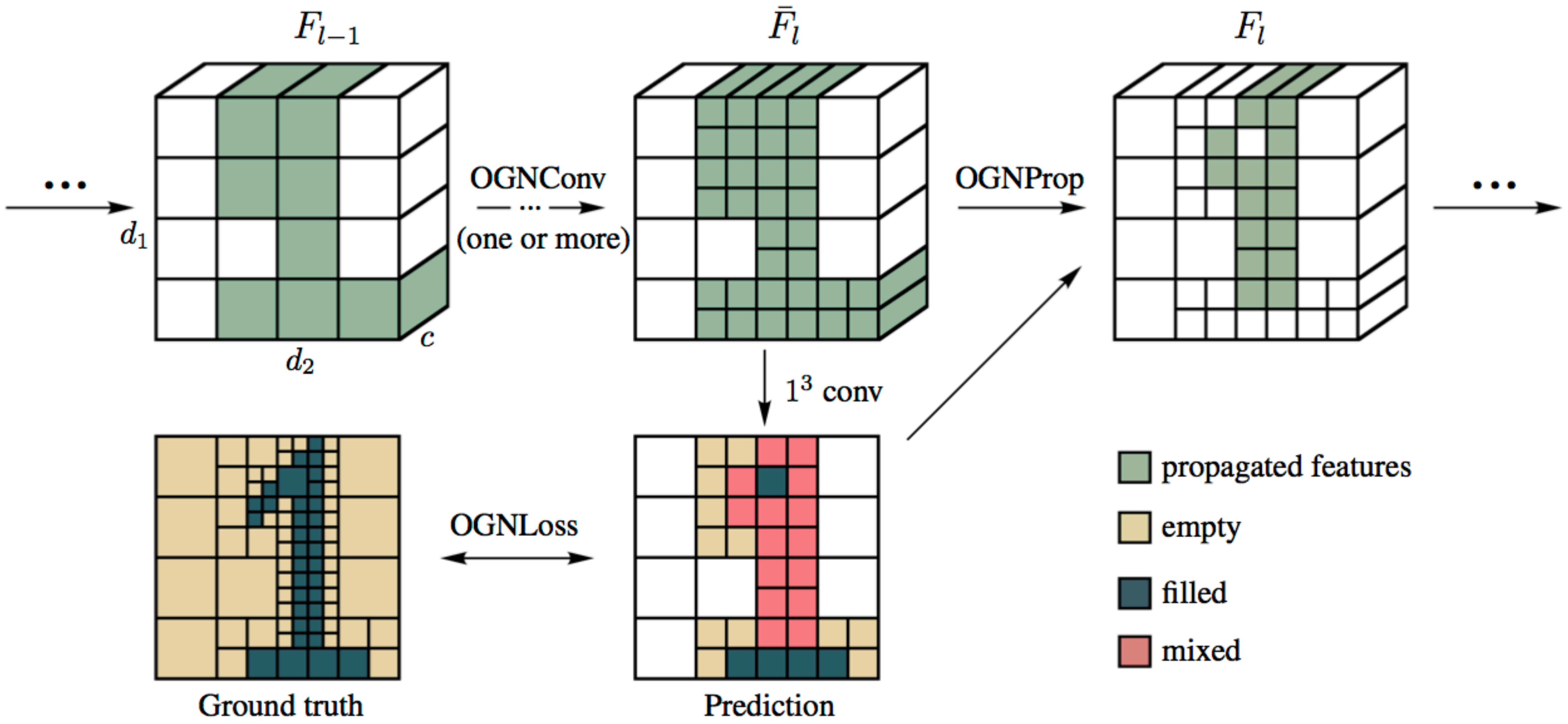
Network	non-voting	voting
VoxNet(32^3)	82.0%	83.0%
GIFT	83.1%	-
Geometry image	83.9%	-
SubVolSup	87.2%	89.2%
FPNN(16^3)	87.3%	-
FPNN(32^3)	87.3%	-
FPNN(64^3)	87.5%	-
FPNN+normal(64^3)	88.4%	-
PointNet	89.2%	-
O-CNN(3)	85.5%	87.1%
O-CNN(4)	88.3%	89.3%
O-CNN(5)	89.6%	90.4%
O-CNN(6)	**89.9%**	**90.6%**
O-CNN(7)	89.5%	90.1%
O-CNN(8)	89.6%	90.2%

Towards higher spatial resolution



Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox
“Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs”
arxiv (March, 2017)

Progressive voxel refinement



3-way classification

Results