Deep Learning in Higher Dimensions

Multi-Dimensional ConvNets

- 1D ConvNets
  - Audio / Speech
  - Also Point Clouds

- 2D ConvNets
  - Images (AlexNet, VGG, ResNet -> Classification, Localization, etc..)

- 3D ConvNets
  - For videos
  - For 3D data

- 4D ConvNets
  - E.g., dynamic 3D data (Haven't seen much work there)
  - Simulations

Remember: 1D Convolutions

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

f

g	**1/3**	**1/3**	**1/3**

$f * g$		**3**								

$$4 \cdot \frac{1}{3} + 3 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} = 3$$

Remember: 1D Convolutions

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

f

g

1/3	**1/3**	**1/3**

$f * g$

	3	**0**							

$$3 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} + (-5) \cdot \frac{1}{3} = 0$$

Remember: 1D Convolutions

4	3	2	-5	3	5	2	5	5	6

f

g

1/3	1/3	1/3

$f * g$

	3	0	0						

$$2 \cdot \frac{1}{3} + (-5) \cdot \frac{1}{3} + 3 \cdot \frac{1}{3} = 0$$

Remember: 1D Convolutions

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

f

g

1/3	**1/3**	**1/3**

$f * g$

	3	**0**	**0**	**1**					

$$(-5) \cdot \frac{1}{3} + 3 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} = 1$$

Remember: 1D Convolutions

f

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

g

1/3	**1/3**	**1/3**

$f * g$

	3	**0**	**0**	**1**	**10/3**				

$$3 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} = \frac{10}{3}$$

Remember: 1D Convolutions

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

f

g

1/3	**1/3**	**1/3**

$f * g$

	3	**0**	**0**	**1**	**10/3**	**4**			

$$5 \cdot \frac{1}{3} + 2 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} = 4$$

Remember: 1D Convolutions

4	3	2	-5	3	5	2	5	5	6

f

g

1/3	1/3	1/3

$f * g$

	3	0	0	1	10/3	4	4		

$$2 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} = 4$$

Remember: 1D Convolutions

4	**3**	**2**	**-5**	**3**	**5**	**2**	**5**	**5**	**6**

f

g

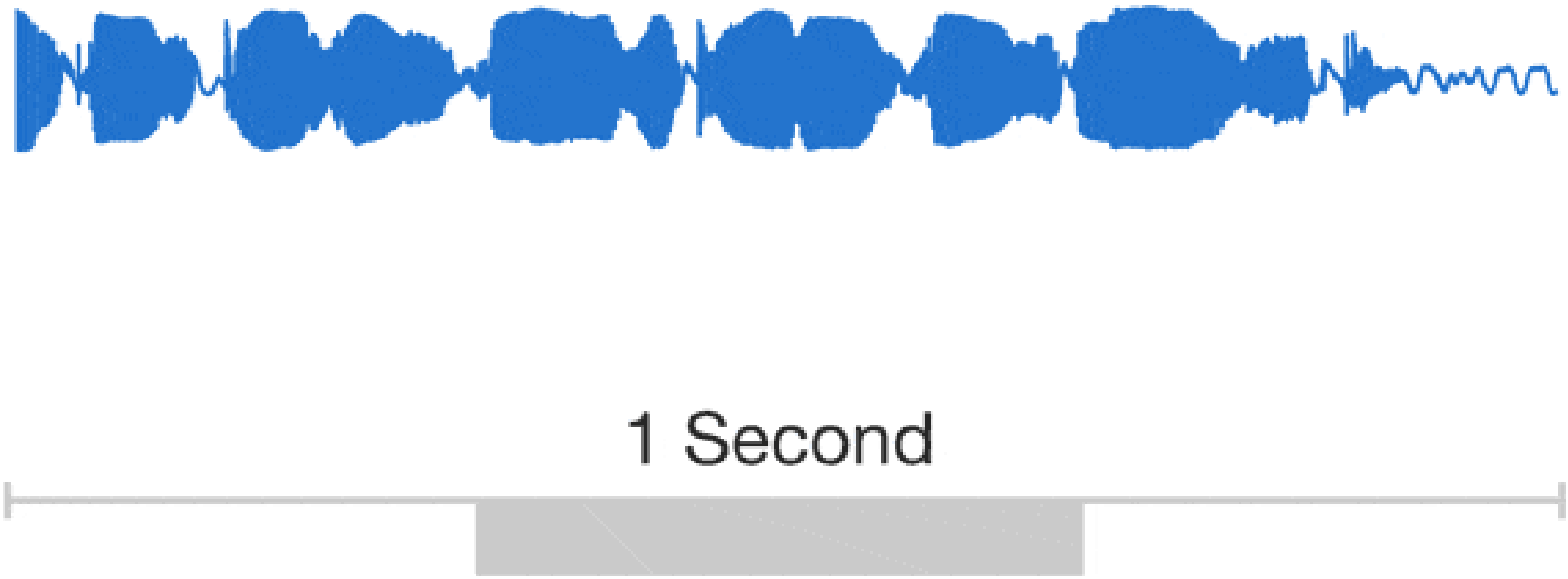
1/3	**1/3**	**1/3**

$f * g$

	3	**0**	**0**	**1**	**10/3**	**4**	**4**	**16/3**	

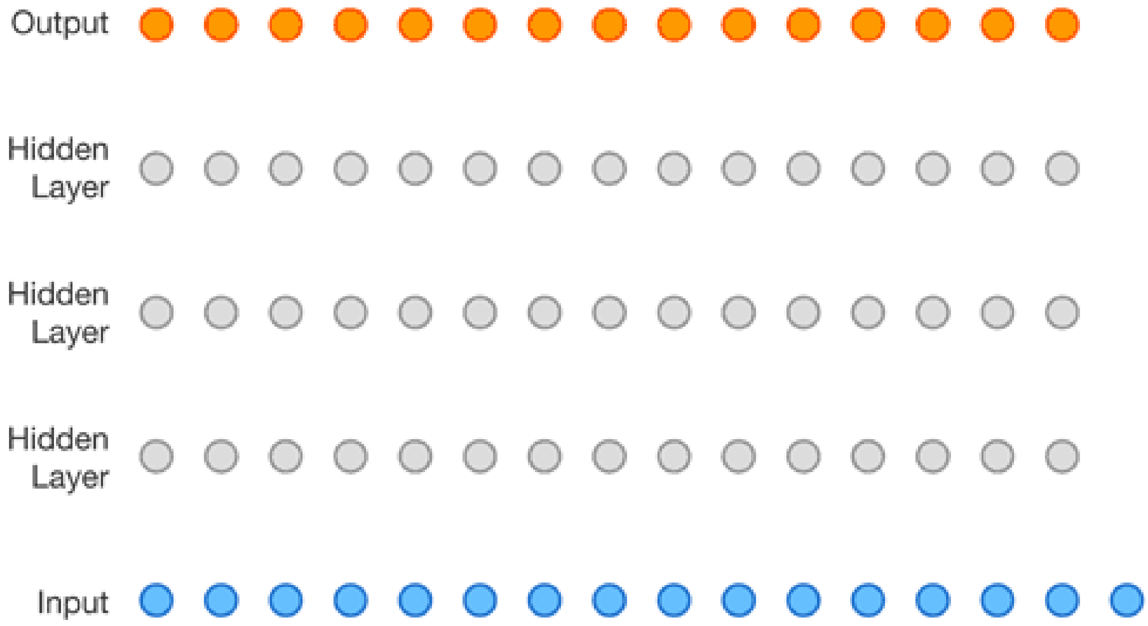
$$5 \cdot \frac{1}{3} + 5 \cdot \frac{1}{3} + 6 \cdot \frac{1}{3} = \frac{16}{3}$$

1D ConvNets: WaveNet



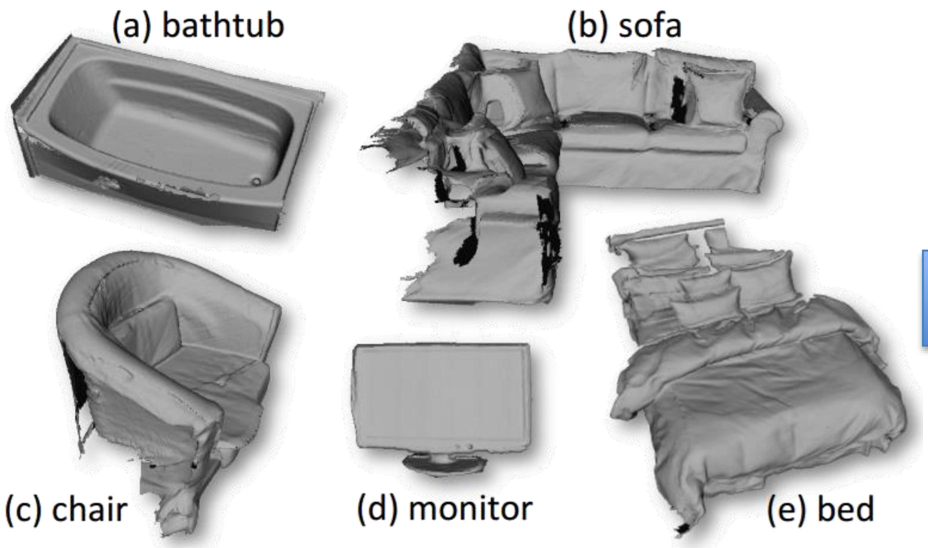
[van der Ooord 16] https://deepmind.com/blog/wavenet-generative-model-raw-audio/

1D ConvNets: WaveNet



[van der Ooord 16] https://deepmind.com/blog/wavenet-generative-model-raw-audio/

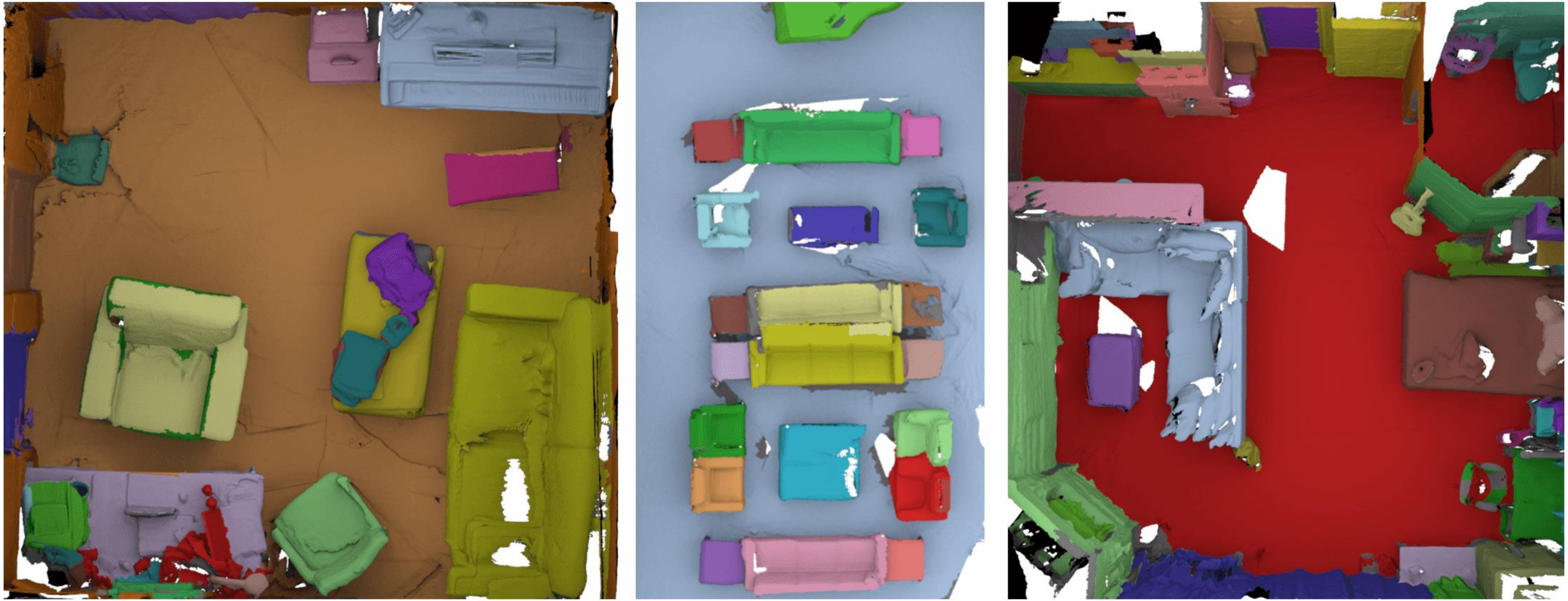
3D Classification



Class from 3D model (e.g., obtained with Kinect Scan)

[Maturana et al. 15] & [Qi et al. 16] 3D vs Multi-view

3D Semantic Segmentation



1500 densely annotated 3D scans; 2.5 mio RGB-D frames

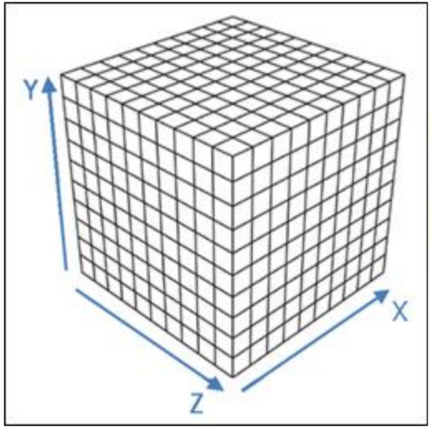
[Dai et al. 17] ScanNet

Volumetric Grids

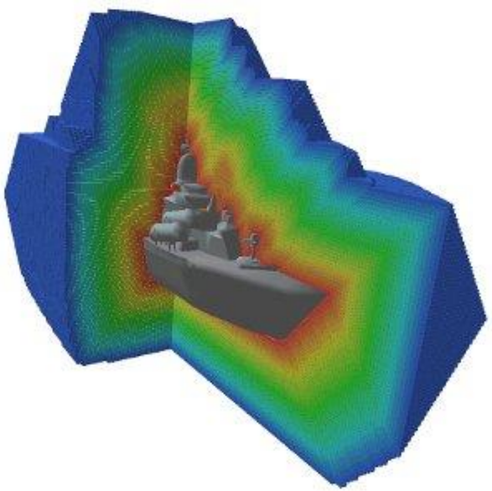
Volumetric Grids

Volumetric Data Structures

- Occupancy grids
- Ternary grids
- Distance Fields
- Signed Distance fields



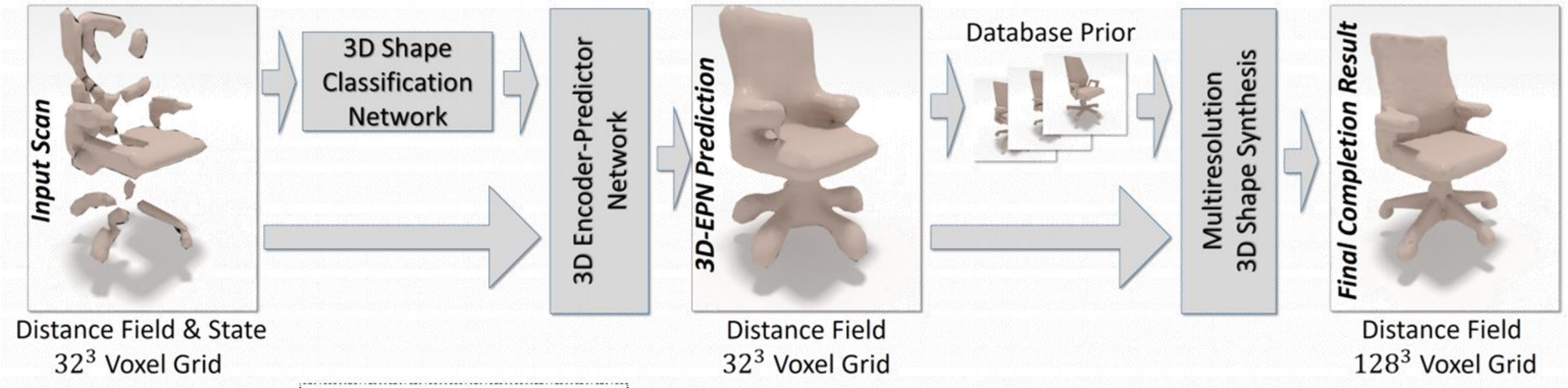
(binary) Voxel Grid

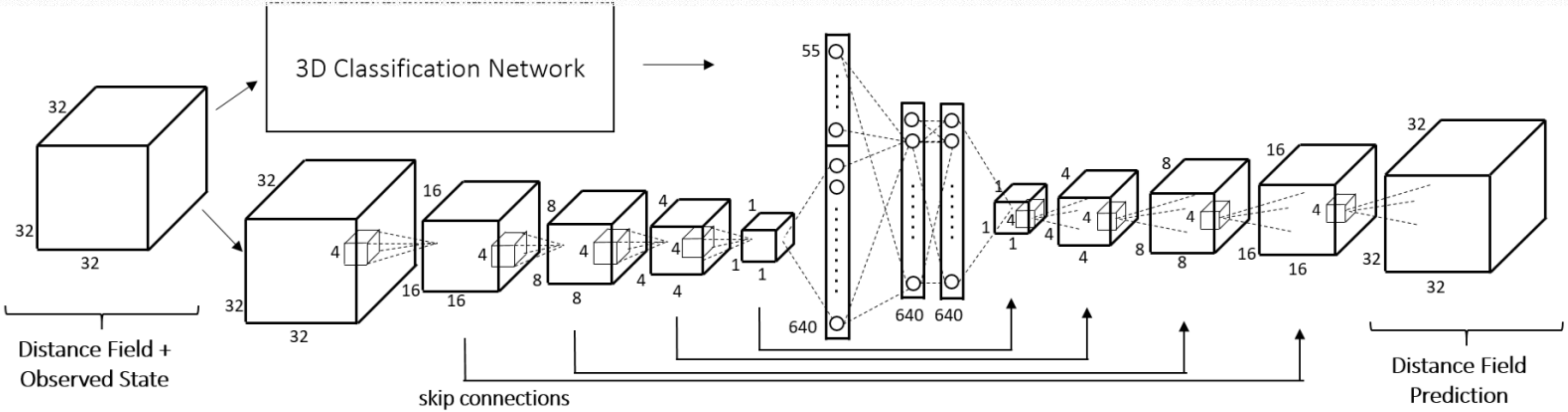


Method	ℓ_1-Err (32^3)	ℓ_1-Err (128^3)
Ours (3D-EPN + synth)	0.382	1.94
Ours (3D-EPN-class + synth)	0.376	1.93
Ours (3D-EPN-unet + synth)	0.310	1.82
Ours (final) 3D-EPN-unet-class + synth	**0.309**	**1.80**

Shape completion error (higher == better)

3D Shape Completion on Grids

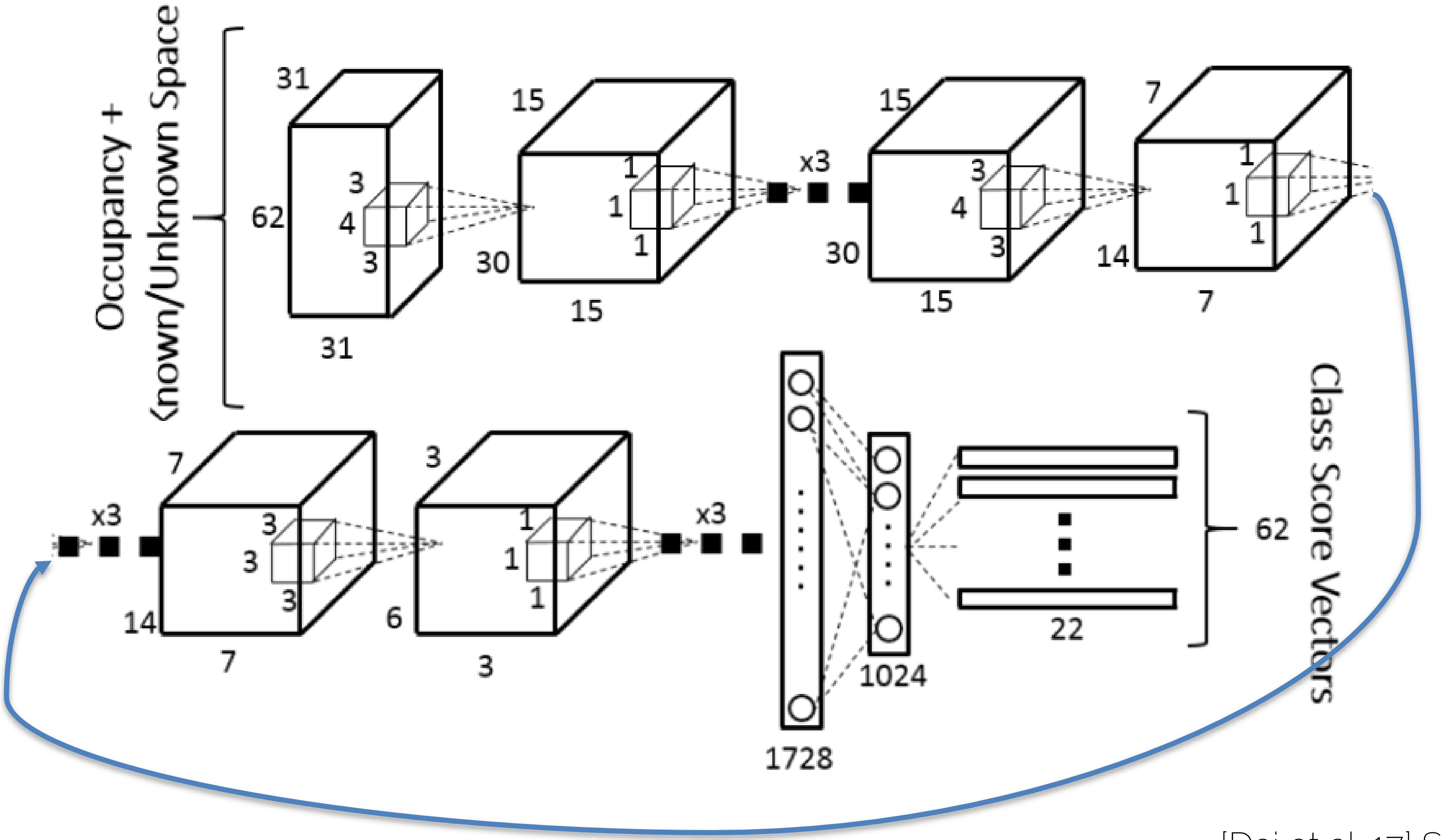




Works with 32 x 32 x 32 voxels...

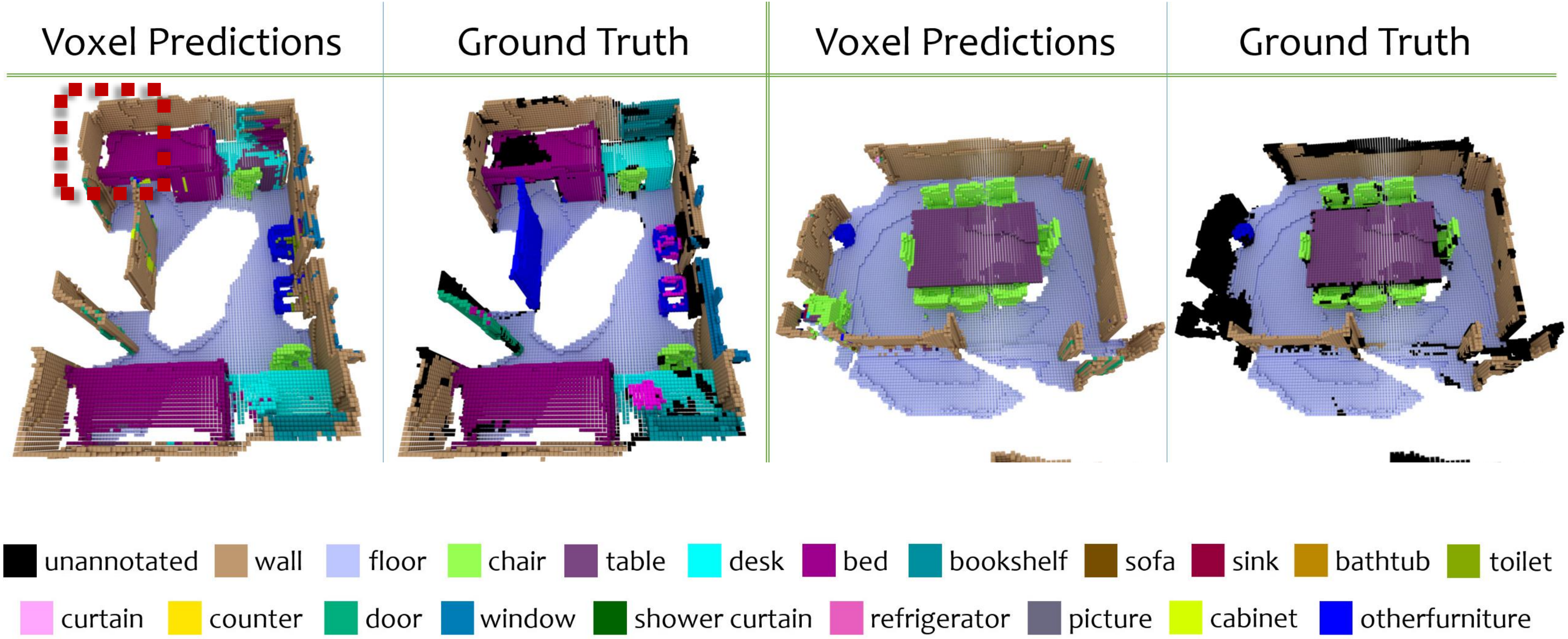
[Dai et al. 17] CNNComplete

ScanNet: Semantic Segmentation in 3D



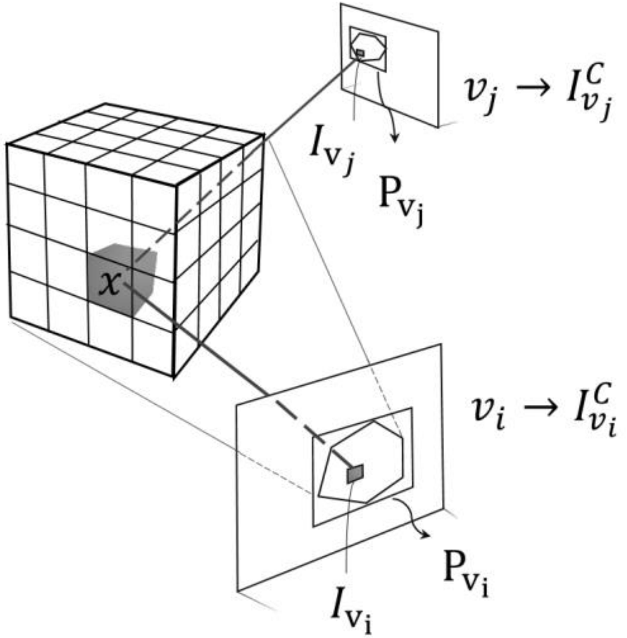
[Dai et al. 17] ScanNet

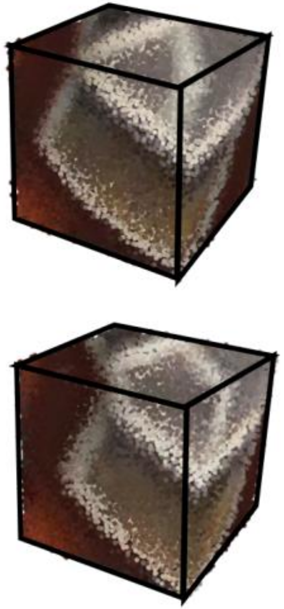
ScanNet: Sliding Window

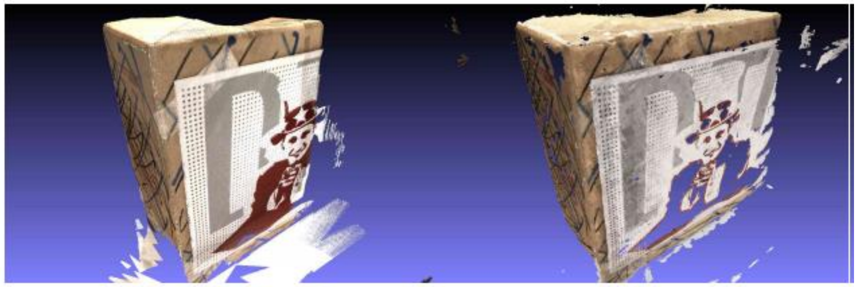


[Dai et al. 17] ScanNet

SurfaceNet: Stereo Reconstruction







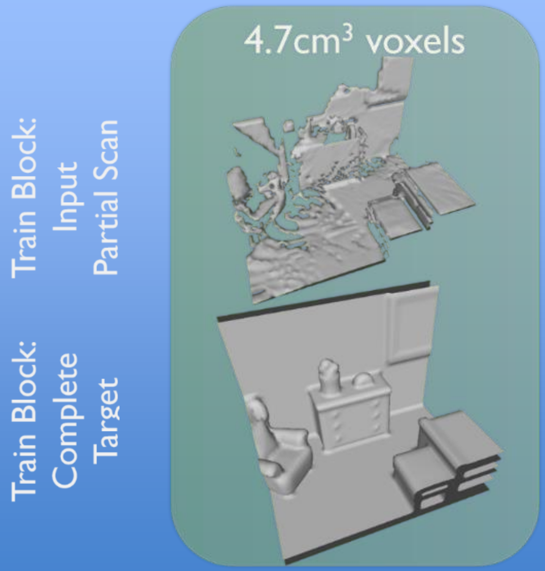
(a) reference model (b) **SurfaceNet**

Run on 32 x 32 x 32 blocks -> takes forever…

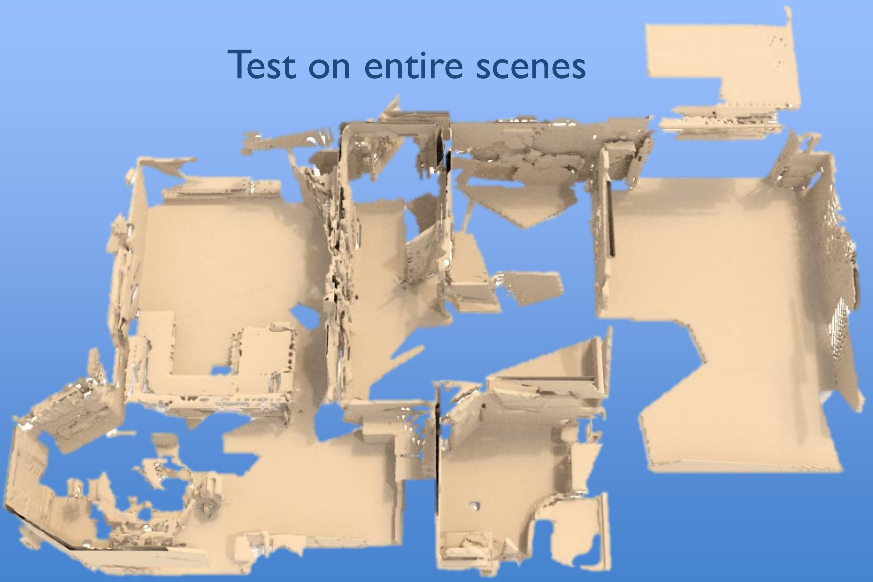
[Ji et al. 17] SurfaceNet

ScanComplete: Fully Convolutional

Train on crops of scenes

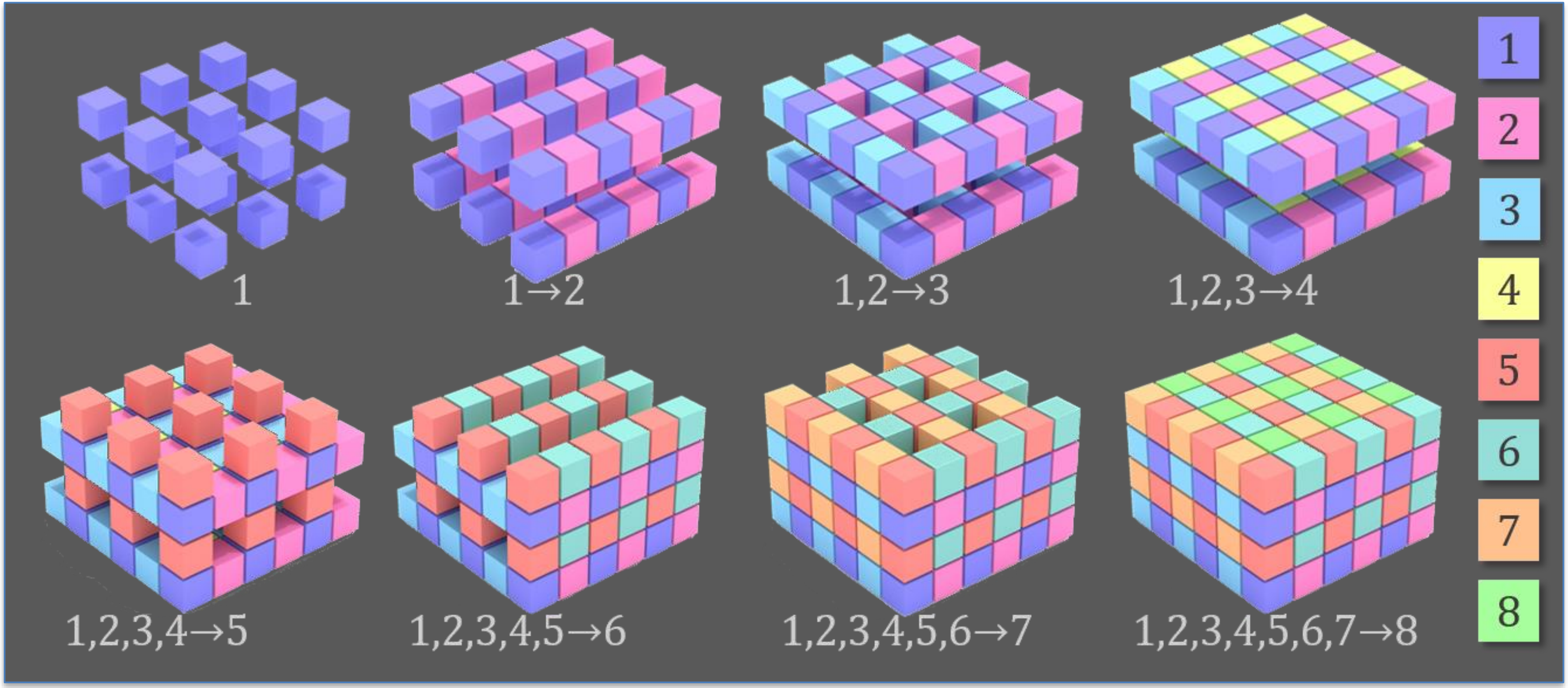


Test on entire scenes



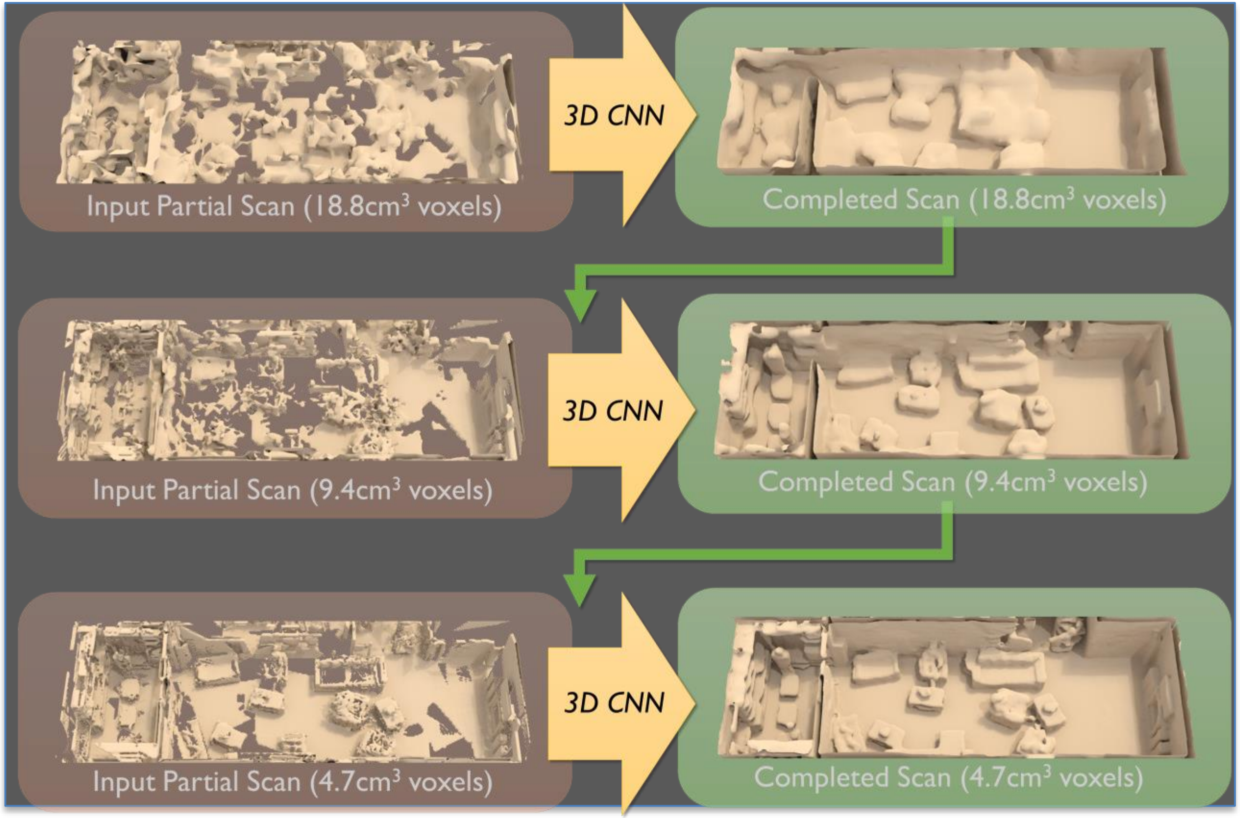
[Dai et al. 18] ScanComplete

Dependent Predictions: Autoregressive Neural Networks



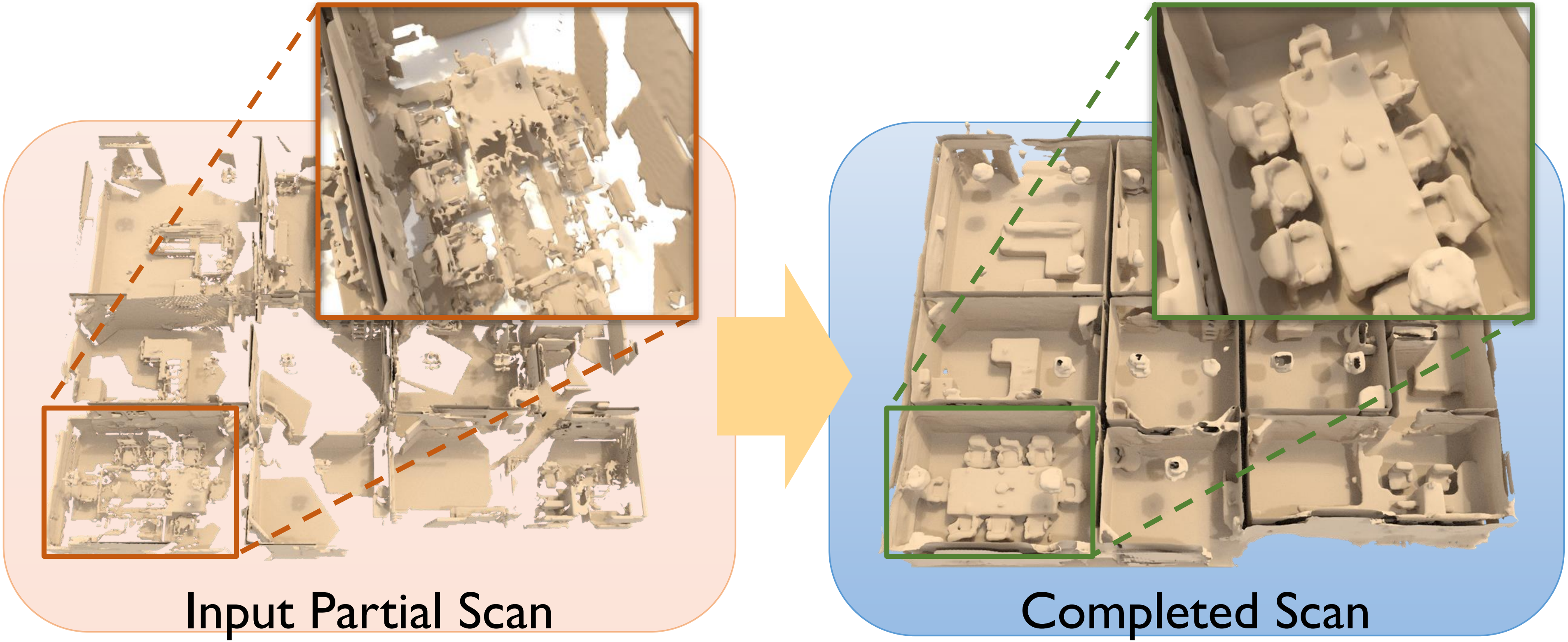
[Dai et al. 18] ScanComplete

Spatial Extent: Coarse-to-Fine Predictions



[Dai et al. 18] ScanComplete

ScanComplete: Fully Convolutional

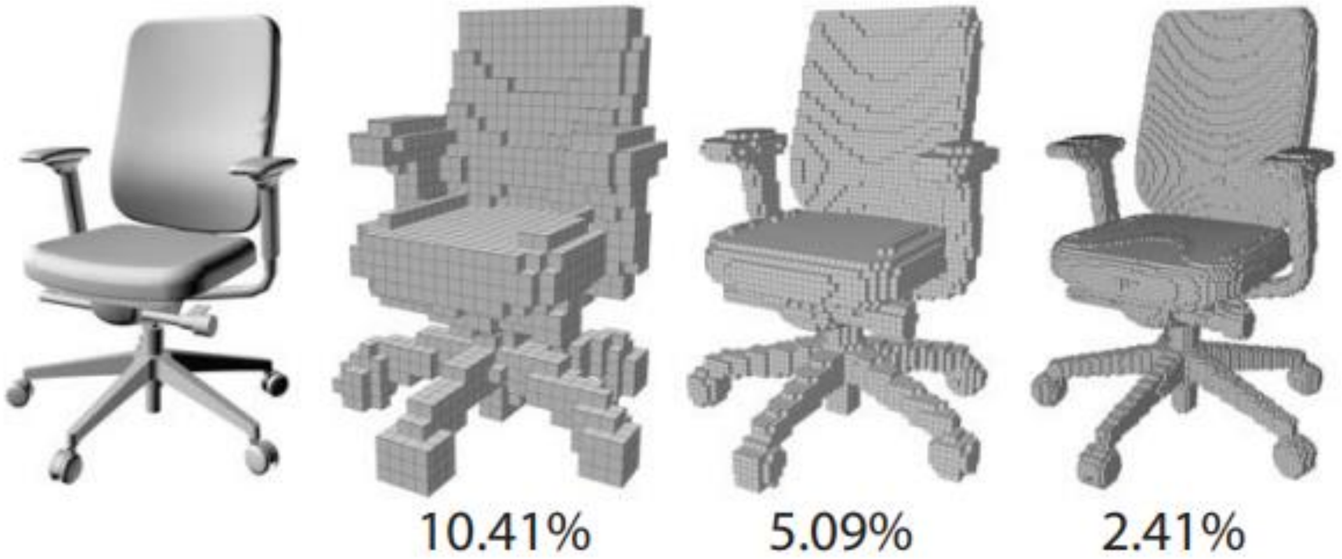


[Dai et al. 18] ScanComplete

Conclusion so far

- Volumetric Grids are easy
  - Encode free space
  - Encode distance fields
  - Need a lot of memory
  - Need a lot of processing time
  - But can be used sliding window or fully-conv.

Conclusion so far



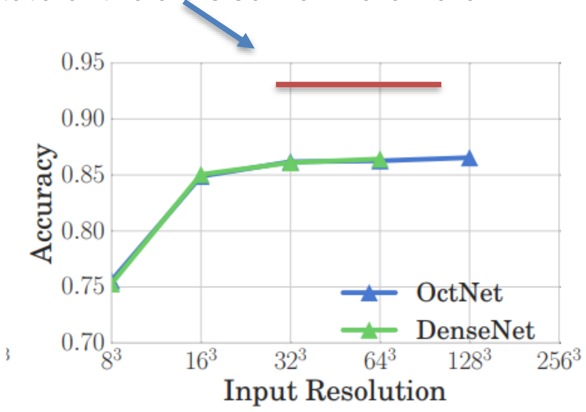
Surface occupancy gets smaller with higher resolutions

Volumetric Hierarchies

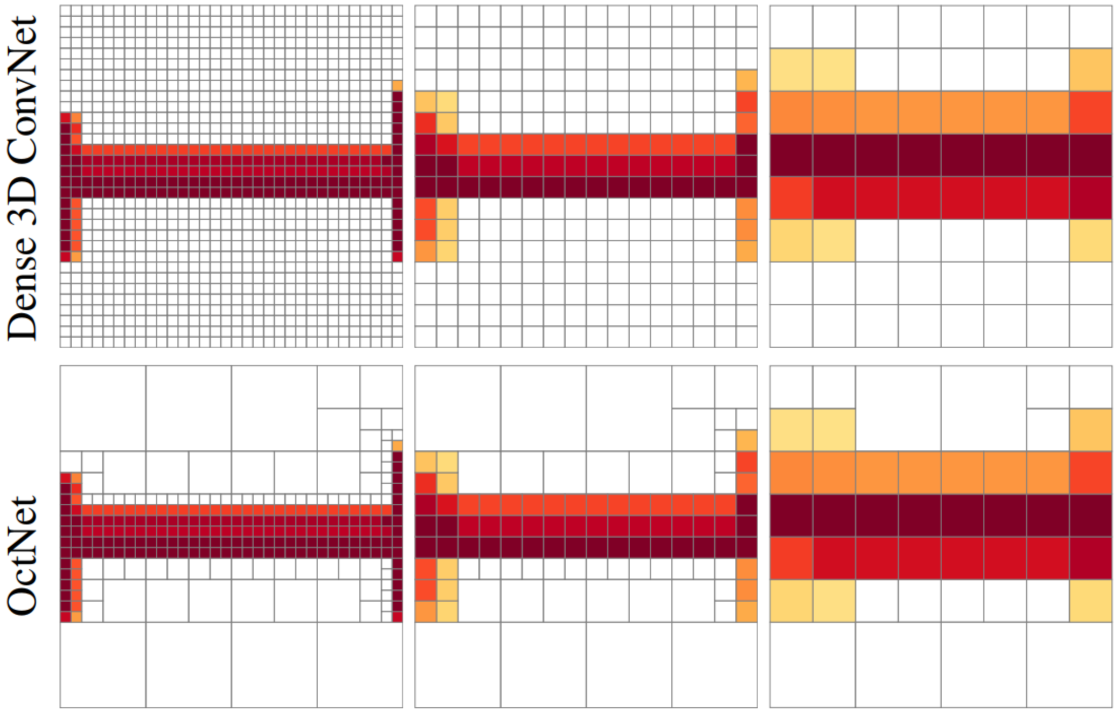
Discriminative Tasks

Structure is known in advance!

State of the art is somewhere here…



(b) Accuracy



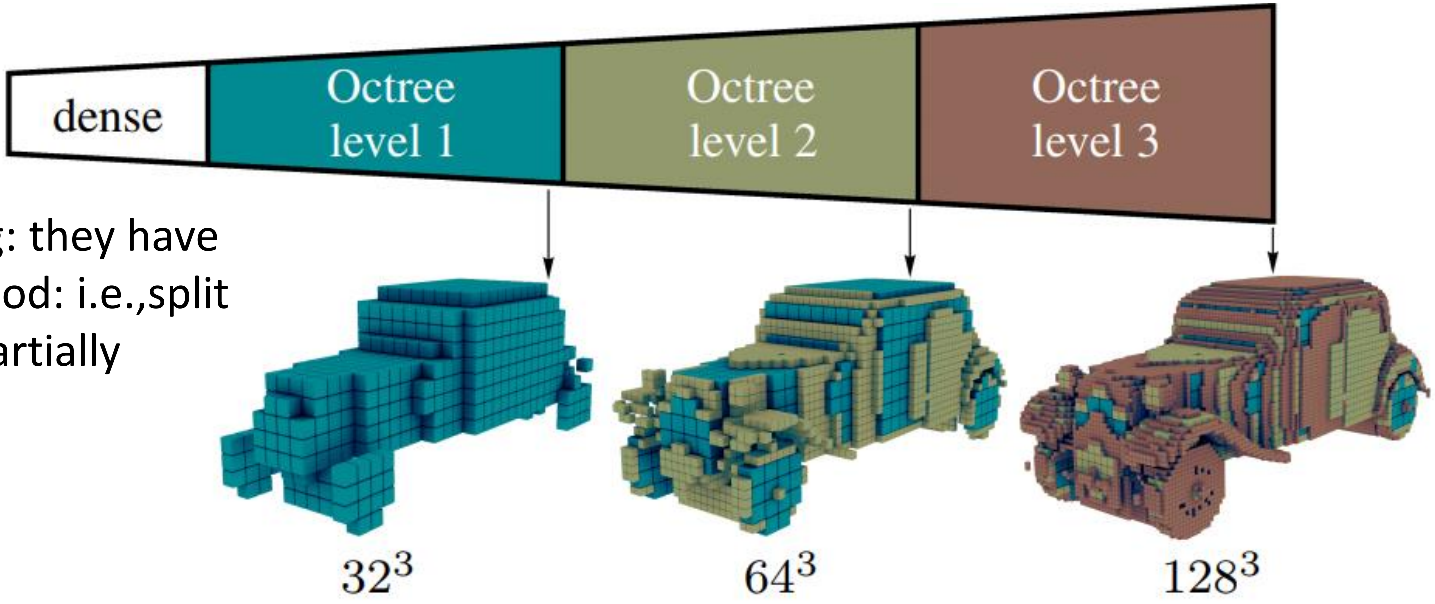
(a) Layer 1: 32^3 (b) Layer 2: 16^3 (c) Layer 3: 8^3

[OctNet: Learning Deep 3D Representations at High Resolutions]() (CVPR 2017)

[O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis]() (SIG17)

Generative Tasks

Need to infer structure!



Pretty interesting: they have end-to-end method: i.e.,split voxels that are partially occupied

[Octree Generating Networks: Efficient Convolutional Architectures for High-resolution Outputs]()
[OctNetFusion: Learning Depth Fusion from Data]() (that one not end to end)

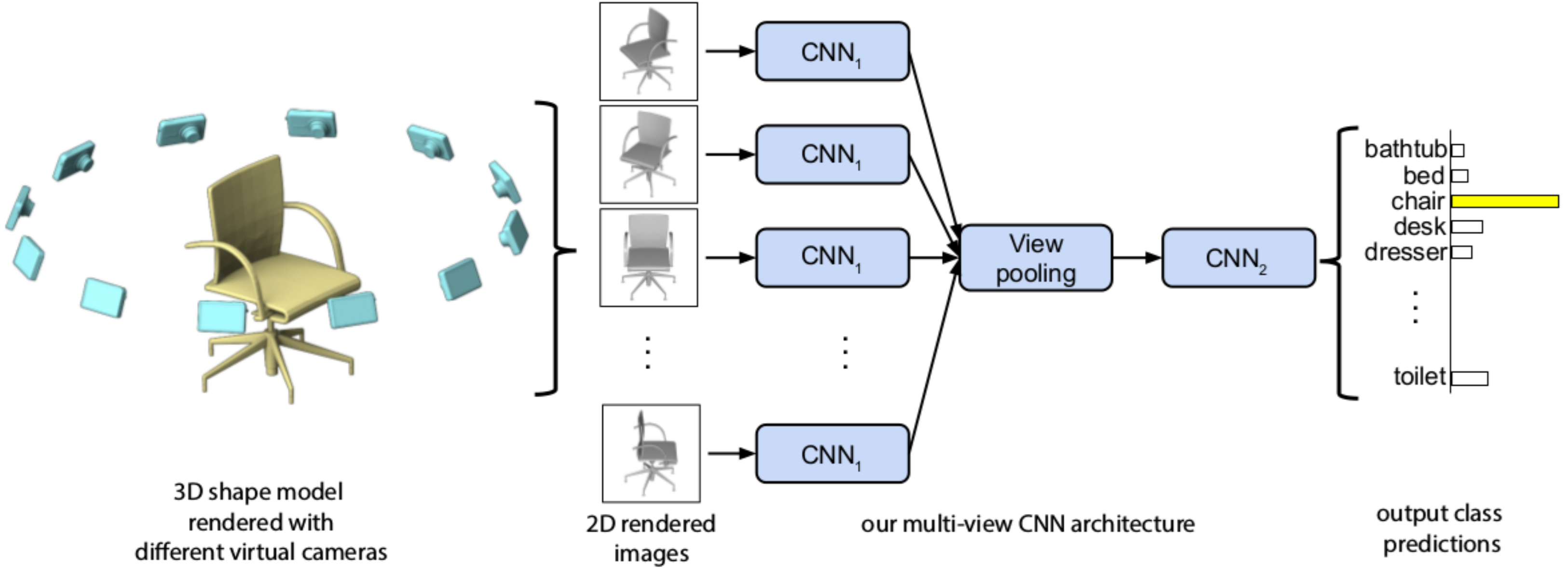
Conclusion so far

- Hierarchies
  - are great for reducing memory and runtime
  - Comes at a performance hit
  - Easier for discriminative tasks when structure is known

Multi-view

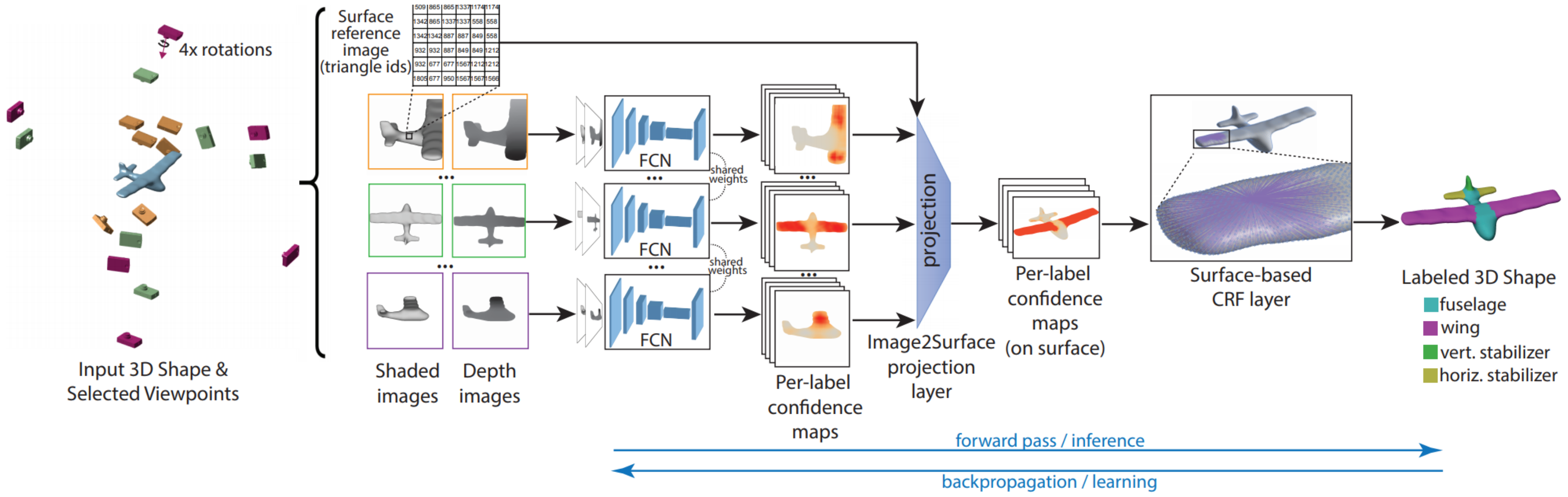
Multiple Views: Classification

- RGB images from fixed views around object:
  - view pooling for classification (only RGB; no spatial corr.)



[Multi-view Convolutional Neural Networks for 3D Shape Recognition]()

Multiple Views: Segmentation



[3D Shape Segmentation with Projective Convolutional Networks]

This one is interesting in a sense that it does 3D shape segmentation (only on CAD models)

But it uses multi-view and has a spatial correlation on top of the mesh surface

Fun thing…

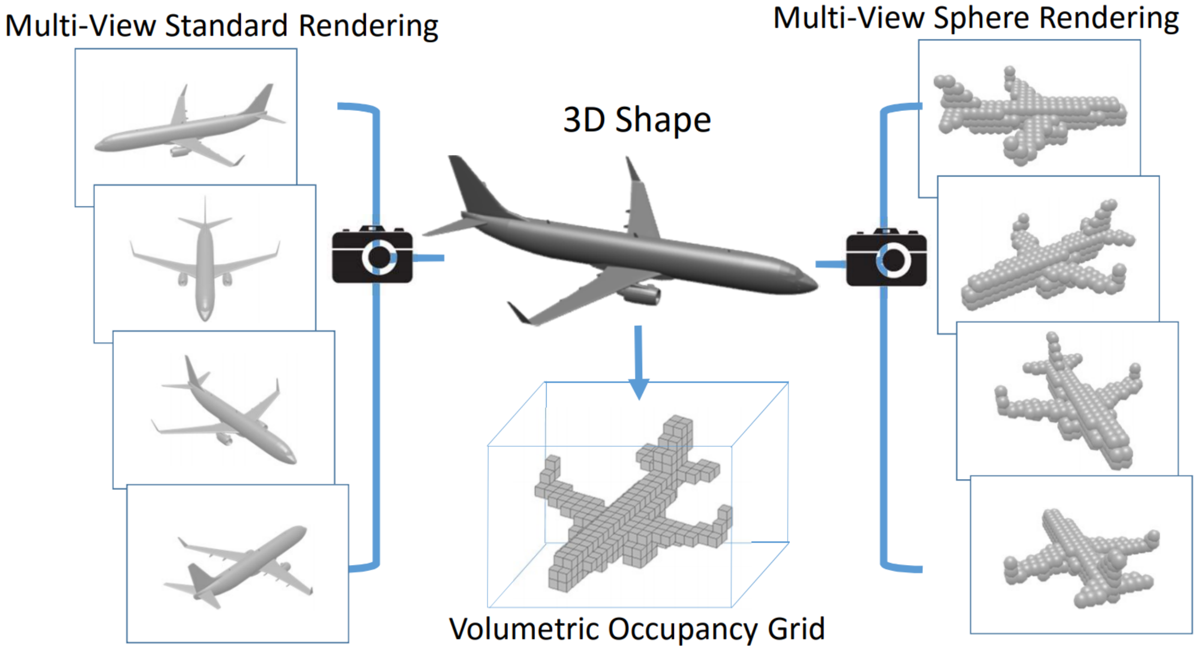


Figure 1. 3D shape representations.

Method	#Views	Accuracy (class)	Accuracy (instance)
SPH (reported by [33])	-	68.2	-
LFD (reported by [33])	-	75.5	-
FV (reported by [32])	12	84.8	-
Su-MVCNN [32]	80	90.1	-
PyramidHoG-LFD	20	87.2	90.5
Ours-MVCNN	20	89.7	92.0
Ours-MVCNN-MultiRes	20	**91.4**	**93.8**

Table 3. Comparison of multi-view based methods. Numbers reported are classification accuracy (class average and instance average) on ModelNet40.

Volumetric and Multi-View CNNs for Object Classification on 3D Data

Hybrid: Volumetric + Multi-view

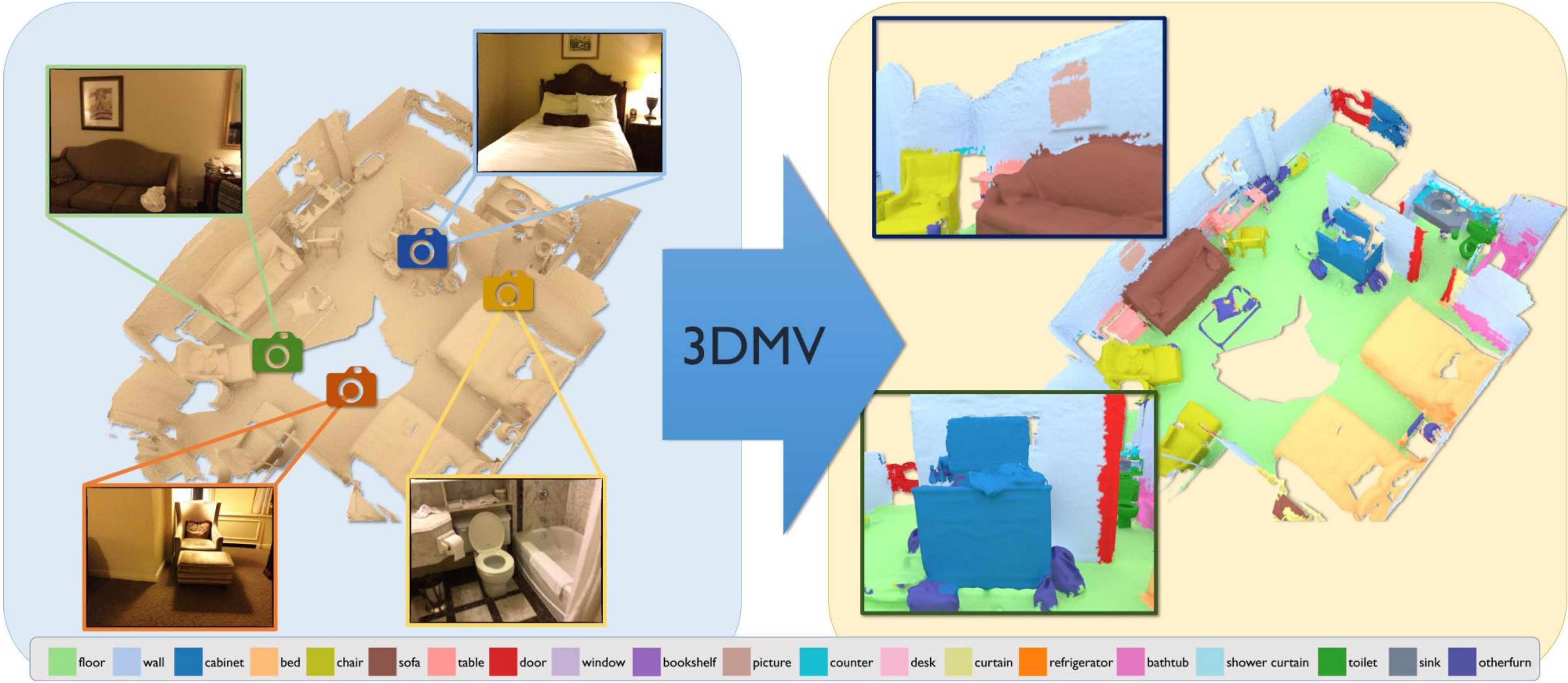
2D + 3D Semantic Segmentation

	avg class accuracy
geometry only	54.4
geometry + voxel colors	55.9

Resolution Mismatch!

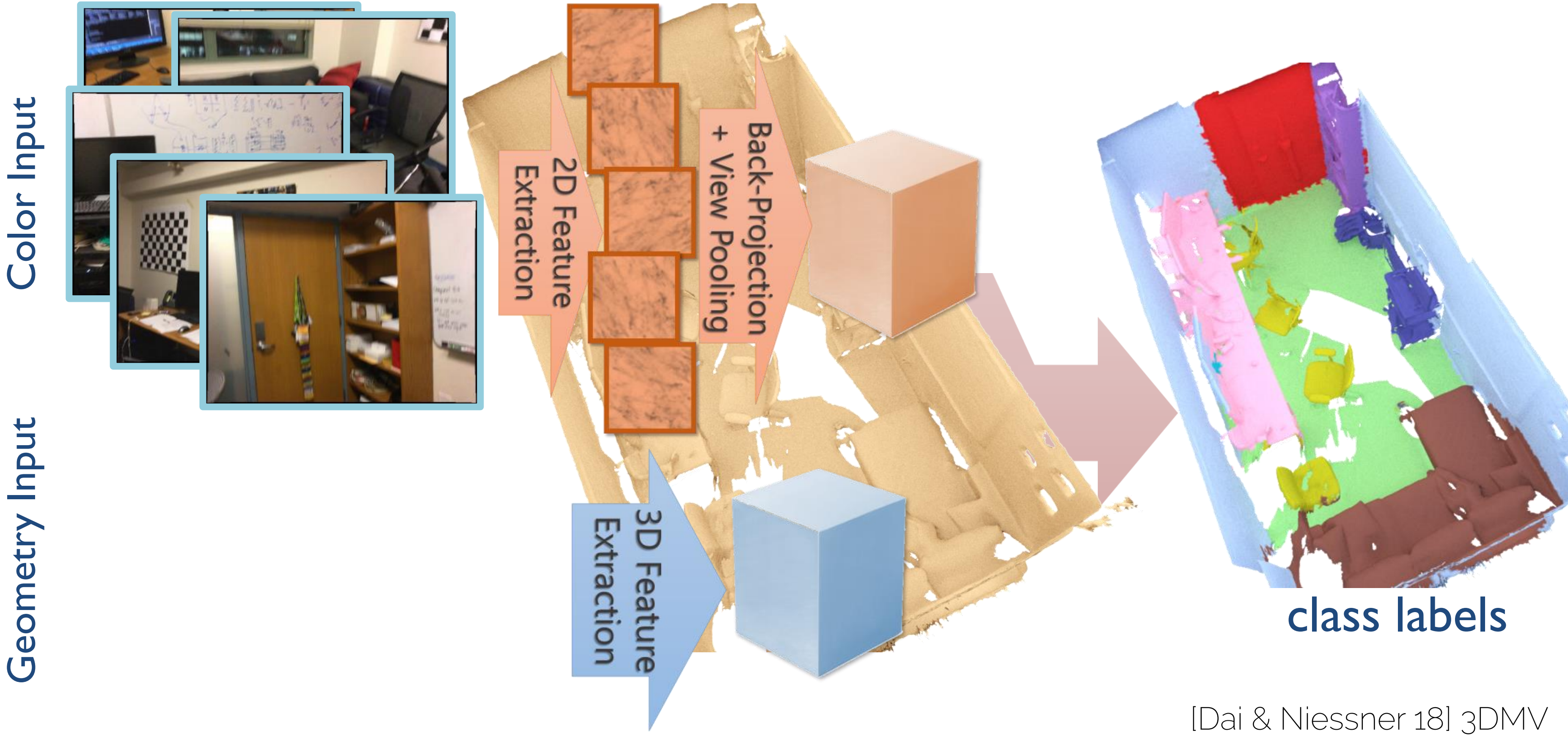
[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view



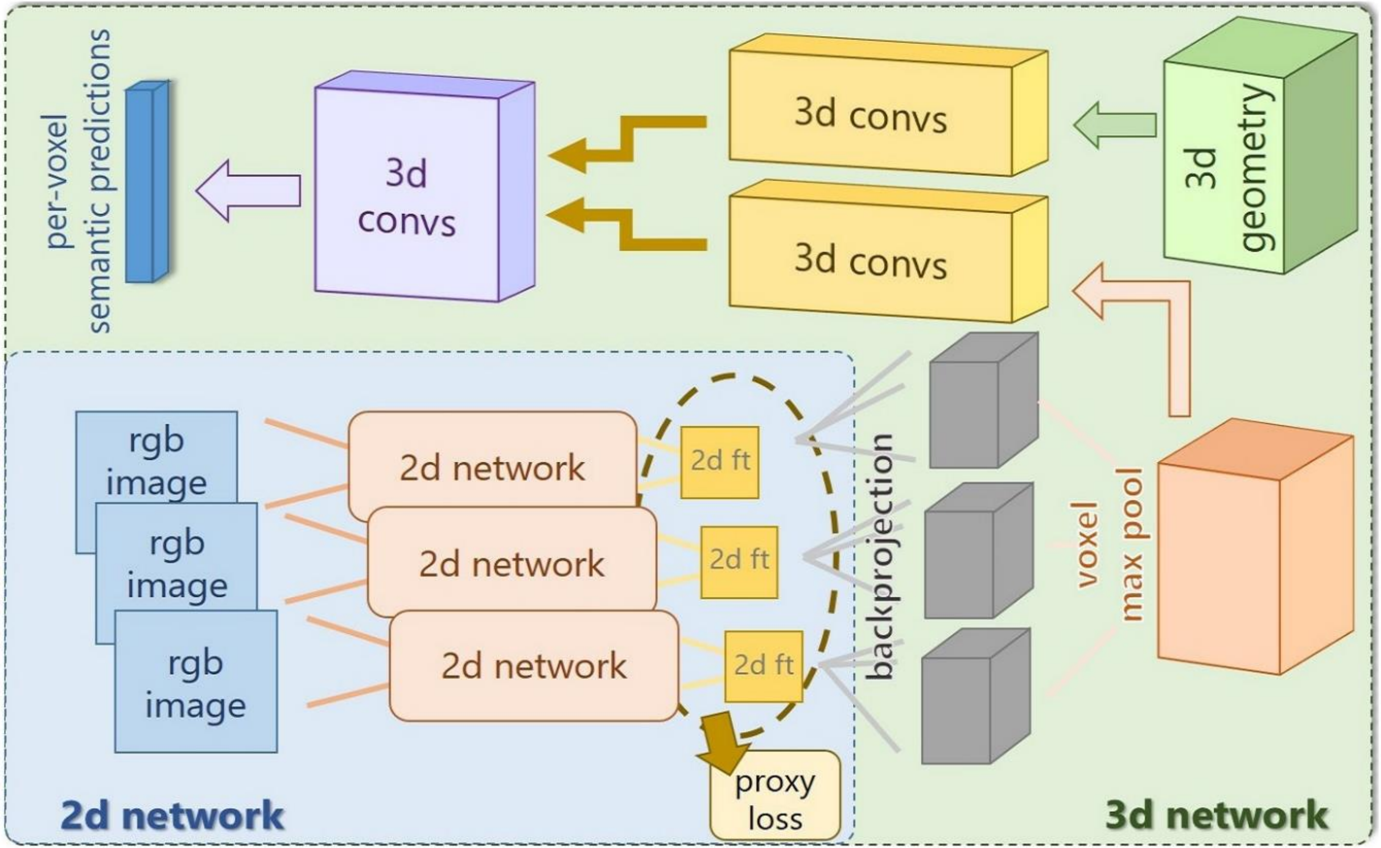
[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view



[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view



[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view

	avg class accuracy
color only	58.2
geometry only	54.4

[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view

	avg class accuracy
color only	58.2
geometry only	54.4
color+geometry	**75.0**

[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view

	avg class accuracy
geometry only	54.4
color+geometry (1 views)	70.1
color+geometry (3 views)	73.0
color+geometry (5 views)	**75.0**

[Dai & Niessner 18] 3DMV

3D Volumetric + Multi-view

	wall	floor	cab	…	bath	other	avg
2d only (1 view)	37.1	39.1	26.7		36.3	20.4	27.1
2d only (3 views)	58.6	62.5	40.8		61.5	34.3	44.2
Ours (no geo input)	76.2	92.9	59.3		80.8	9.3	58.2
Ours (3d geo only)	60.4	95.0	54.4		87.0	20.6	54.4
Ours (3d geo+voxel color)	58.8	94.7	55.5		85.4	20.5	55.9
Ours (1 view, fixed 2d)	77.3	96.8	**70.0**	…	87.0	58.5	69.1
Ours (1 view)	70.7	96.8	61.4		81.6	51.7	70.1
Ours (3 view, fixed 2d)	**81.1**	96.4	58.0		92.5	**60.7**	72.8
Ours (3 view)	75.2	**97.1**	66.4		89.9	57.2	73.0
Ours (5 view, fixed 2d)	77.3	95.7	68.9		93.5	59.6	74.5
Ours (5 view)	73.9	95.6	69.9		**94.7**	58.5	**75.0**

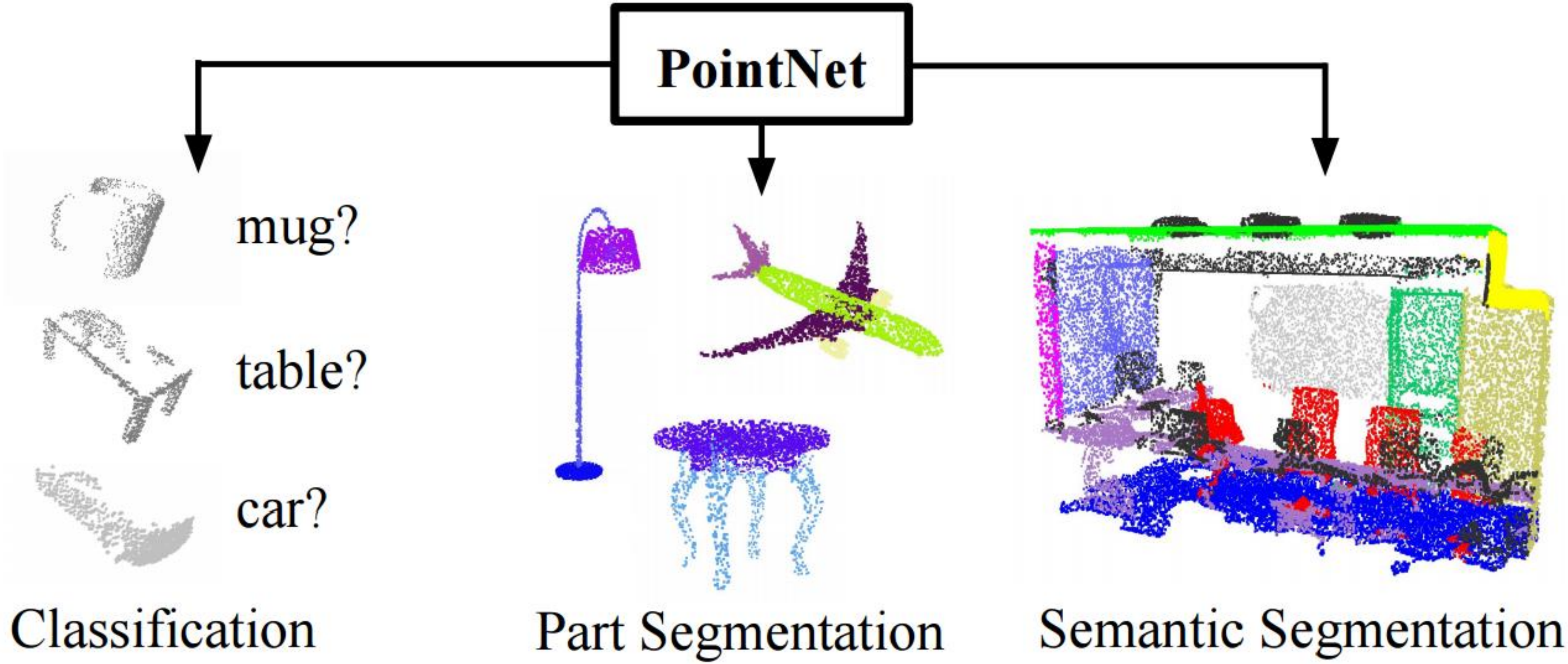
[Dai & Niessner 18] 3DMV

Conclusion so far

- Hybrid:
  - Nice way to combine color and geometry
  - Great performance (best so far for segmentation)
  - End-to-end helps less than we hoped for
  - Could be faster…

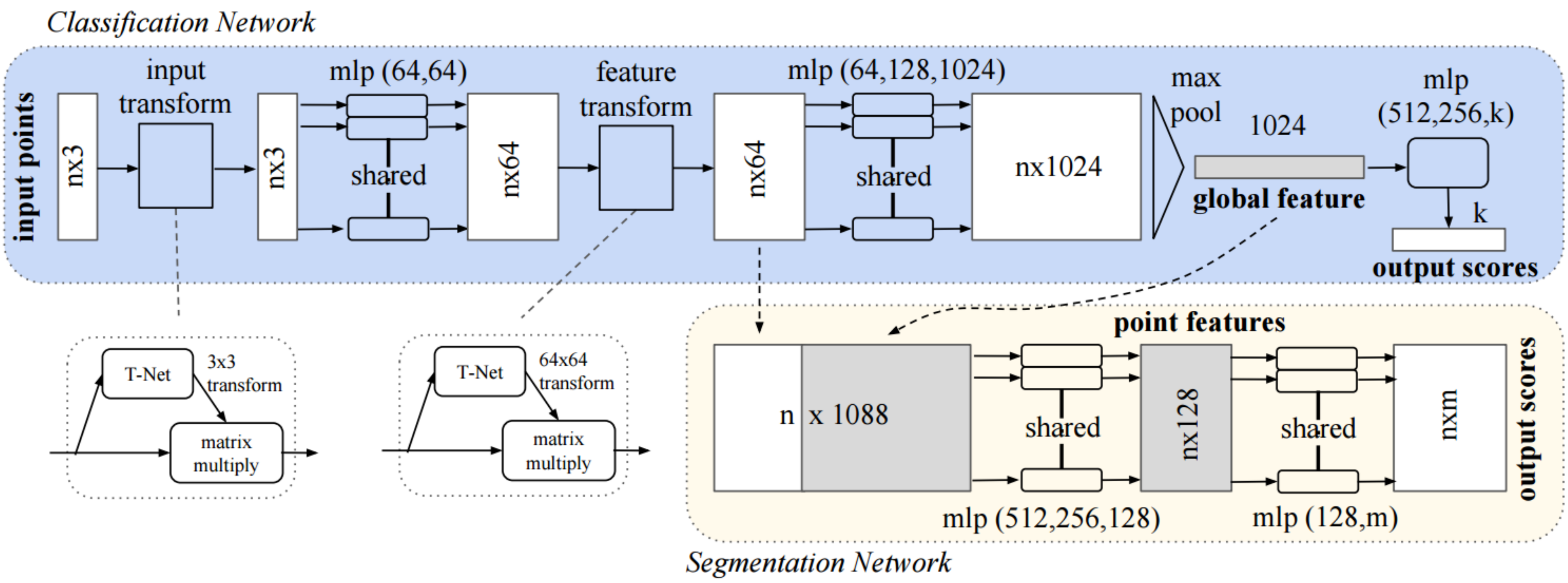
Point Clouds

DeepLearning on Point Clouds: PointNet



[Qi et al. 17] PointNet

DeepLearning on Point Clouds: PointNet



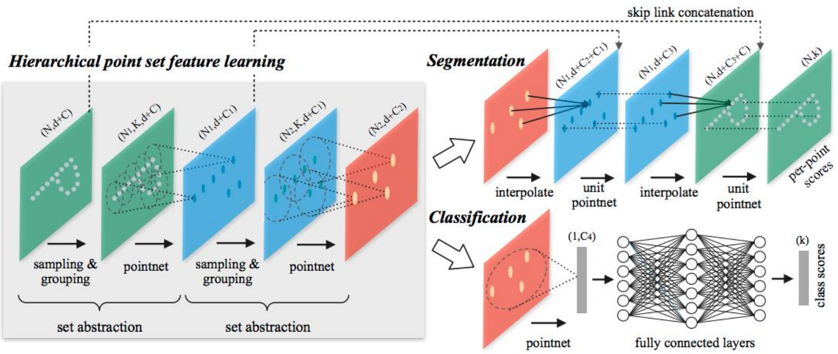
[Qi et al. 17] PointNet

PointNet++

Main idea

- Learn hierarchical representation of point cloud
- Apply multiple (simplified) PointNets at different locations and scales
- Each Scale: Furthest-Point Sampling -> Query Ball Grouping -> PointNet
- Multi-scale or Multi-resolution grouping for sampling density robustness

Evaluations: Classification, Part-Segmentation, Scene-Segmentation



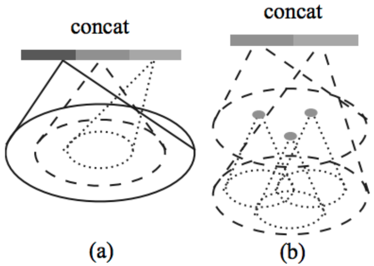


Figure 3: (a) Multi-scale grouping (MSG); (b) Multi-resolution grouping (MRG).

[Qi et al. 17] PointNet++

Point Convolutions

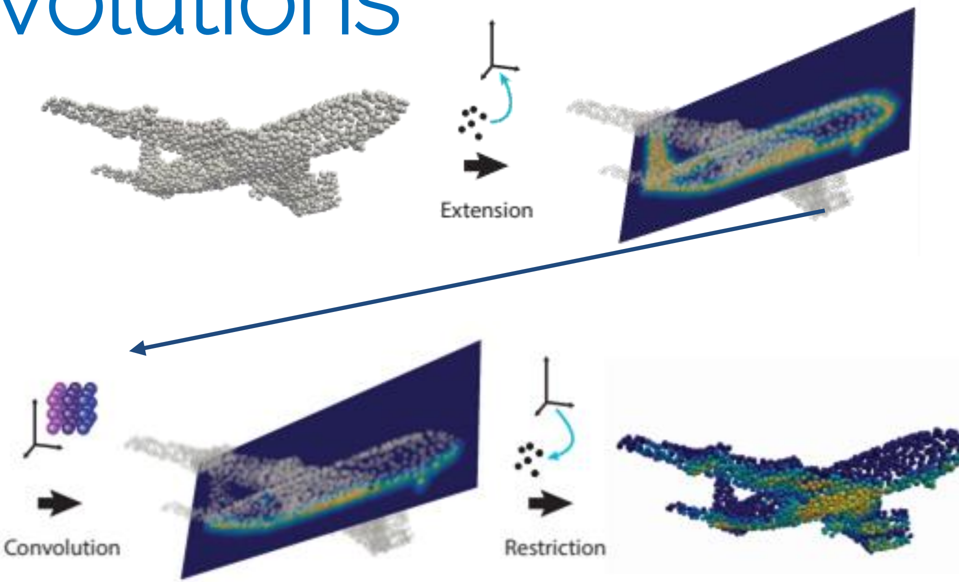
Main idea

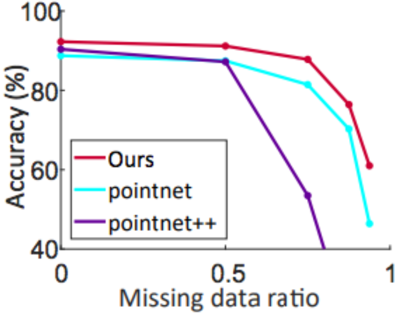
- Transform points to continuous R3 representation (RBFs)
- Convolve in R3
- Restrict results to points

Uses Gaussian RBF representation.

Boils down to computing fixed weights for convolution.

Don't use real data as far as I know!





[Point Convolutional NN by Extension Operators]()
Matan Atzmon, Haggai Maron, Yaron Lipman (SIGGRAPH 2018)

Conclusions so far

- PointNet variants:
  - Train super fast (also testing)
  - Can cover large spaces in one shot
  - Cannot represent free space
  - Performance (mostly) worse than pure volumetric
  - Still lots of ongoing research!

Point Sets (global)

Unordered point set

PointNet (CVPR 2017)

Hierarchy of point sets

PointNet++: Deep Hierarchical Feature Learning on Point Sets in a ...(NIPS 2017)

Generalized Convolutional Neural Networks for Point Cloud Data (ICMLA17)

Kd-tree

Escape from Cells: Deep Kd-Networks (ICCV 2017)

PointCNN

PointCNN (seems arxiv only)

Point Sets (local)

RBF

- Point Convolutional NN by Extension Operators (SIGGRAPH 2018)
- Tangent Convolutions for Dense Prediction in 3D (CVPR 2018)

Nearest point neighborhoods

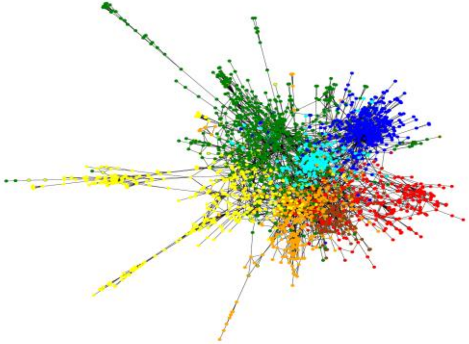
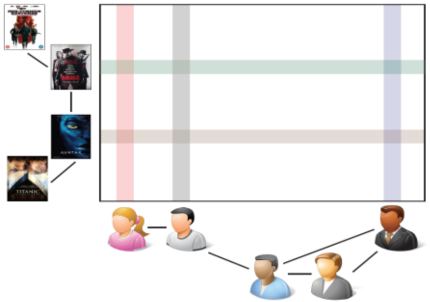
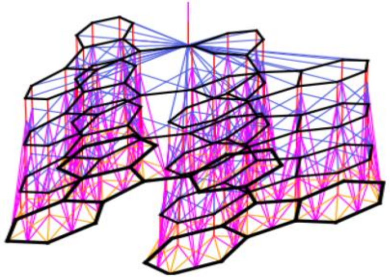
- Dynamic edge-conditioned filters in convolutional neural networks on graphs (CVPR17)
- 3D Graph Neural Networks for RGBD Semantic Segmentation (ICCV17)
- PPFNet: Global context aware local features for robust 3d point matching (CVPR18)
- FeaStNet: Feature-Steered Graph Convolutions for 3D Shape Analysis (CVPR18)

Very interesting combination where convolutions are essentially over line segments in 3D, and where both locations and are being optimized
https://arxiv.org/abs/1605.06240
Idea is great, performance could be a bit better (probably hard to optimize)

Mesh-based

Convs on Meshes and Graphs

- Lots of work by Michael Bronstein et al.

https://www.imperial.ac.uk/people/m.bronstein/publications.html

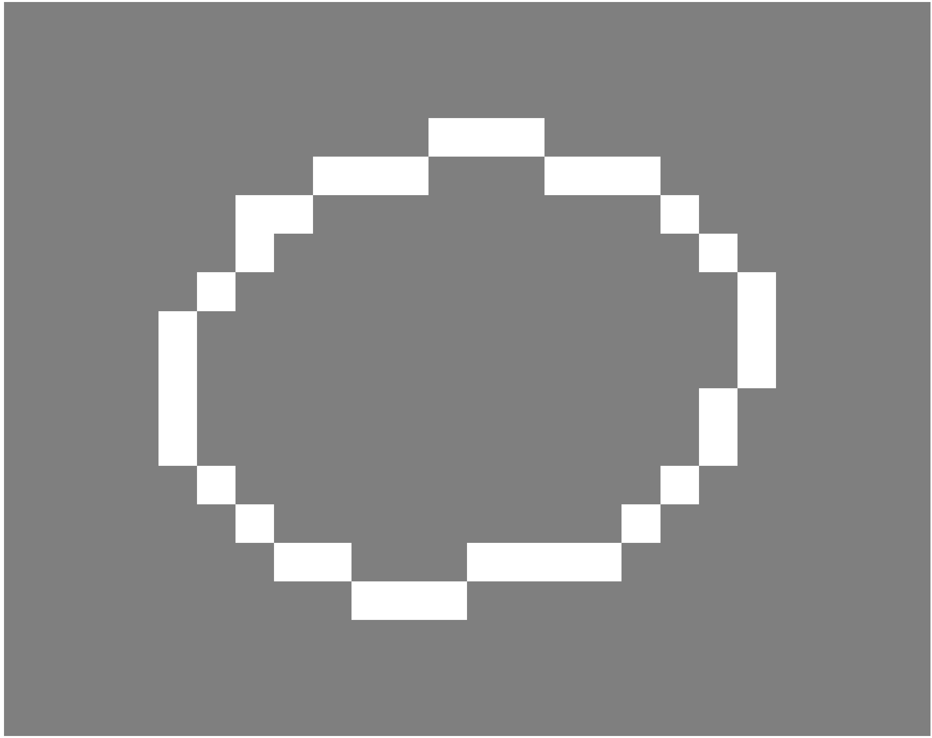
Conclusion so far

- Meshes / Surfaces:
  - Needs some differential geometry approximation
  - Convolutions in DG space
  - I haven't seen results on real-world data
    - Probably prone to noise and incomplete scans

Sparse Convolutions

Sparse Convolutional Networks

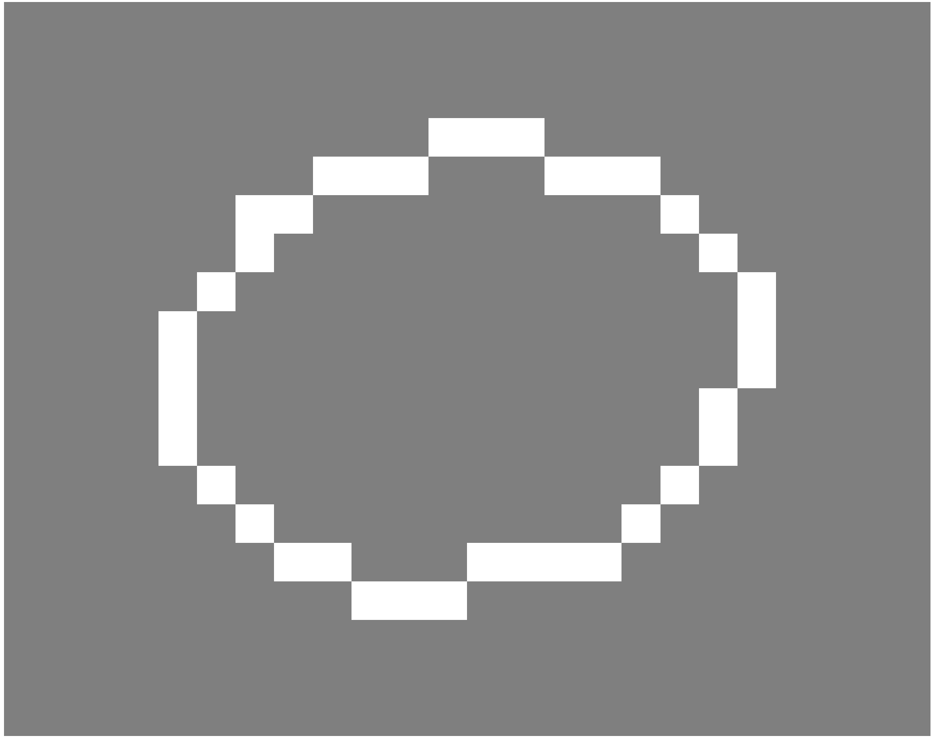
Regular, dense 3x3 Convolution
-> set of actives (non-zeros) grows rapidly
-> need a lot of memory
-> takes a long time for feature prop.



[Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307](https://arxiv.org/abs/1706.01307) **[Graham et al. 18]**
3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

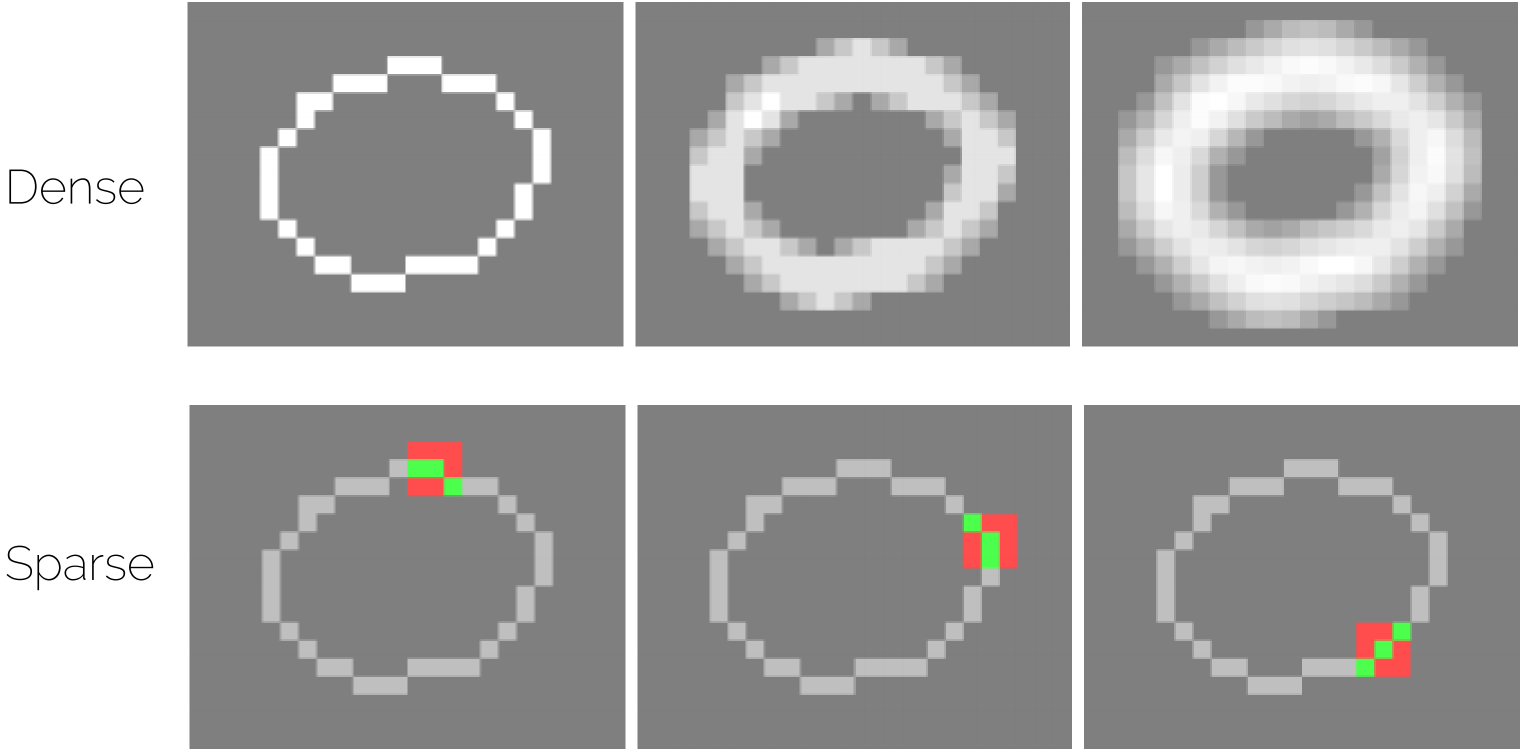
Sparse Convolutional Networks

Regular, dense 3x3 Convolution
-> set of actives (non-zeros) grows rapidly
-> need a lot of memory
-> takes a long time for feature prop.



[Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307](https://arxiv.org/abs/1706.01307) **[Graham et al. 18]**
3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

Sparse Convolutional Networks

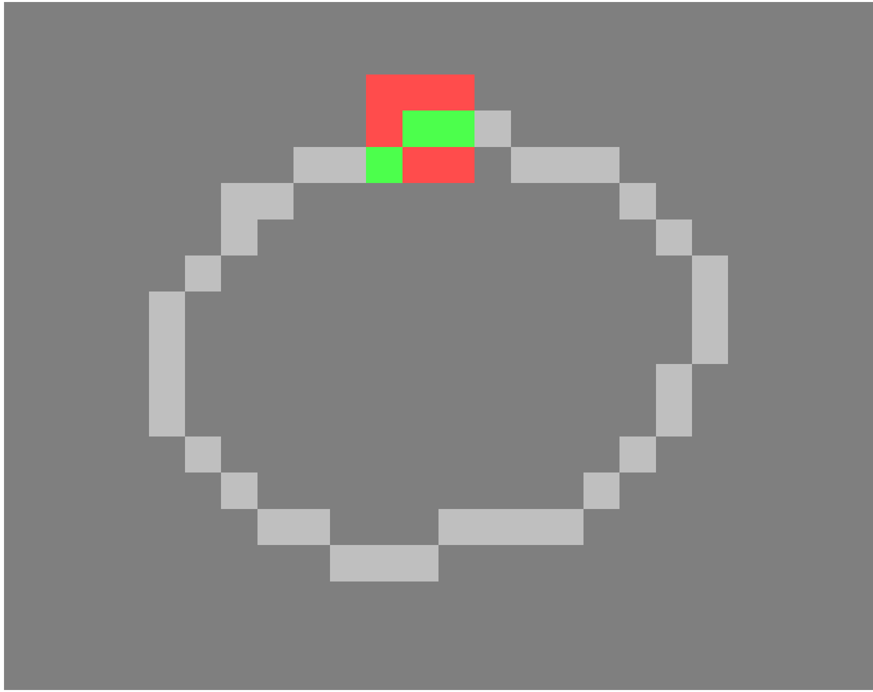


[Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307](https://arxiv.org/abs/1706.01307) **[Graham et al. 18]**

3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

Sparse Convolutional Networks

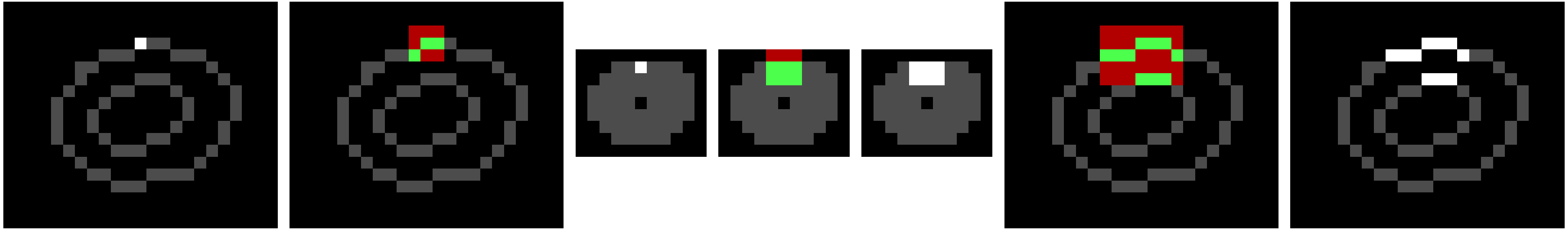
Submanifold Sparse Conv:
-> set of active sites is unchanged
-> active sites look at active neighbors (green)
-> non-active sites (red) have no overhead



[Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307](https://arxiv.org/abs/1706.01307) **[Graham et al. 18]**
3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

Sparse Convolutional Networks

Submanifold Sparse Conv:
-> disconnected components do not communicate at first
-> although they will merge due to effect of stride, pooling, convs, etc.



from left: (i) an active point is highlighted; a convolution with stride 2 sees the green active sites (ii) and produces output (iii), 'children' of hightlighted active point from (i) are highlighted; a submanifold sparse convolution sees the green active sites (iv) and produces output (v); a deconvolution operation sees the green active sites (vi) and produces output (vii).

[Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307](https://arxiv.org/abs/1706.01307) [Graham et al. 18]
3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

Sparse Convolutional Networks

Dimension	Name in 'torch.nn'	Use cases
1	Conv1d	Text, audio
2	Conv2d	Lines in 2D space, e.g. handwriting
3	Conv3d	Lines and surfaces in 3D space or (2+1)D space-time
4	-	Lines, etc, in (3+1)D space-time

https://github.com/facebookresearch/SparseConvNet

Submanifold Sparse Convolutional Networks, https://arxiv.org/abs/1706.01307 **[Graham et al. 18]**
3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

Conclusions so far

- Spares (volumetric) Convs:
  - Implemented with spatial hash function
  - Features only around "surface"
  - Require significantly less memory
  - Allow for much higher resolutions
  - It's slower, but much higher accuracy

3D Scene Understanding

3D Semantic label benchmark

This table lists the benchmark results for the 3D semantic label scenario.

Evaluation on Hidden Test Set on ScanNet

Method	Info	avg iou	bathtub	bed	bookshelf	cabinet	chair	counter	curtain	desk	door	floor	otherfurniture	picture	refrigerator	show curt
SparseConvNet		0.726 1	0.629 6	0.801 1	0.858 1	0.713 1	0.884 1	0.505 1	0.799 1	0.636 2	0.628 1	0.956 1	0.602 1	0.299 1	0.712 1	0.85
MinkowskiNet34		0.679 2	0.811 1	0.734 3	0.739 3	0.641 2	0.804 2	0.413 4	0.759 3	0.696 1	0.545 2	0.938 5	0.518 2	0.141 10	0.623 2	0.75
joint point-based		0.621 3	0.645 4	0.746 2	0.612 6	0.571 4	0.795 3	0.386 5	0.798 2	0.485 4	0.539 3	0.943 4	0.445 3	0.287 2	0.520 4	0.41
TextureNet		0.566 4	0.672 2	0.664 5	0.671 4	0.494 5	0.719 5	0.445 2	0.678 4	0.411 8	0.396 7	0.935 6	0.356 7	0.225 3	0.412 7	0.53
DVVNet		0.562 5	0.648 3	0.700 4	0.770 2	0.586 3	0.687 8	0.333 7	0.650 5	0.514 3	0.475 4	0.906 12	0.359 6	0.223 4	0.340 9	0.44
PointConv		0.556 6	0.636 5	0.640 7	0.574 8	0.472 7	0.739 4	0.430 3	0.433 8	0.418 7	0.445 6	0.944 2	0.372 5	0.185 7	0.464 5	0.57
3DMV, FTSDF		0.501 7	0.558 8	0.608 9	0.424 13	0.478 6	0.690 7	0.246 11	0.586 6	0.468 5	0.450 5	0.911 10	0.394 4	0.160 8	0.438 6	0.21
3DMV		0.484 8	0.484 11	0.538 11	0.643 5	0.424 8	0.606 13	0.310 8	0.574 7	0.433 6	0.378 8	0.796 14	0.301 8	0.214 5	0.537 3	0.20
Angela Dai, Matthias Niessner: 3DMV: Joint 3D-Multi-View Prediction for 3D Semantic Scene Segmentation. ECCV'18																
PointCNN with RGB	P	0.479 9	0.510 9	0.583 10	0.417 14	0.414 9	0.708 6	0.241 13	0.367 11	0.405 10	0.323 9	0.944 2	0.300 9	0.132 11	0.226 13	0.41
Yangyan Li, Rui Bu, Mingchao Sun, Baoquan Chen: PointCNN. NIPS 2018																
SurfaceConvPF		0.442 10	0.505 10	0.622 8	0.380 15	0.342 12	0.654 10	0.227 14	0.397 10	0.367 11	0.276 11	0.924 8	0.240 11	0.198 6	0.359 8	0.26
Hao Pan, Shilin Liu, Yang Liu, Xin Tong: Convolutional Neural Networks on 3D Surfaces Using Parallel Frames.																
Tangent Convolutions	P	0.438 11	0.437 13	0.646 6	0.474 10	0.369 10	0.645 11	0.353 6	0.258 13	0.282 14	0.279 10	0.918 9	0.298 10	0.147 9	0.283 10	0.29
Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, Qian-Yi Zhou: Tangent convolutions for dense prediction in 3d. CVPR 2018																
SPLAT Net	C	0.393 12	0.472 12	0.511 12	0.606 7	0.311 13	0.656 9	0.245 12	0.405 9	0.328 13	0.197 14	0.927 7	0.227 13	0.000 16	0.001 16	0.24
Hang Su, Varun Jampani, Deqing Sun, Subhransu Maji, Evangelos Kalogerakis, Ming-Hsuan Yang, Jan Kautz: SPLATNet: Sparse Lattice Networks for Point Cloud Processing. CVPR 2018																
ScanNet+FTSDF		0.383 13	0.297 15	0.491 13	0.432 12	0.358 11	0.612 12	0.274 9	0.116 15	0.411 8	0.265 12	0.904 13	0.229 12	0.079 14	0.250 11	0.18
PointNet++	P	0.339 14	0.584 7	0.478 14	0.458 11	0.256 15	0.360 16	0.250 10	0.247 14	0.278 15	0.261 13	0.677 16	0.183 14	0.117 12	0.212 14	0.14
Charles R. Qi, Li Yi, Hao Su, Leonidas J. Guibas: pointnet++: deep hierarchical feature learning on point sets in a metric space.																
SSC-UNet	P	0.308 15	0.353 14	0.290 16	0.278 16	0.166 16	0.553 14	0.169 16	0.286 12	0.147 16	0.148 16	0.908 11	0.182 15	0.064 15	0.023 15	0.01
ScanNet	P	0.306 16	0.203 16	0.366 15	0.501 9	0.311 13	0.524 15	0.211 15	0.002 16	0.342 12	0.189 15	0.786 15	0.145 16	0.102 13	0.245 12	0.15
Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, Matthias Nießner: ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. CVPR'17																

⋮

http://www.scan-net.org/

Next Lectures

- This is the last lecture slot!
- Keep working on the projects!
- Research opportunities ☺ ☺ ☺

See you ☺