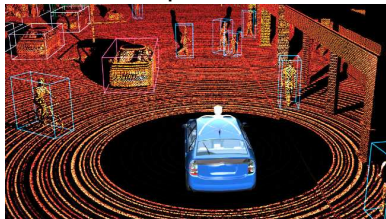


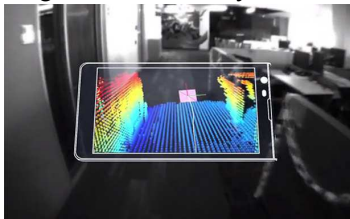
The Need for 3D Deep Learning

Robot Perception



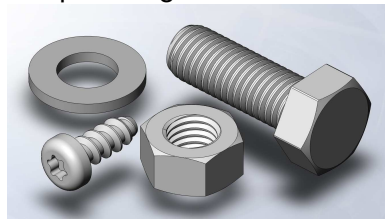
source: Scott J Grunewald

Augmented Reality



source: Google Tango

Shape Design



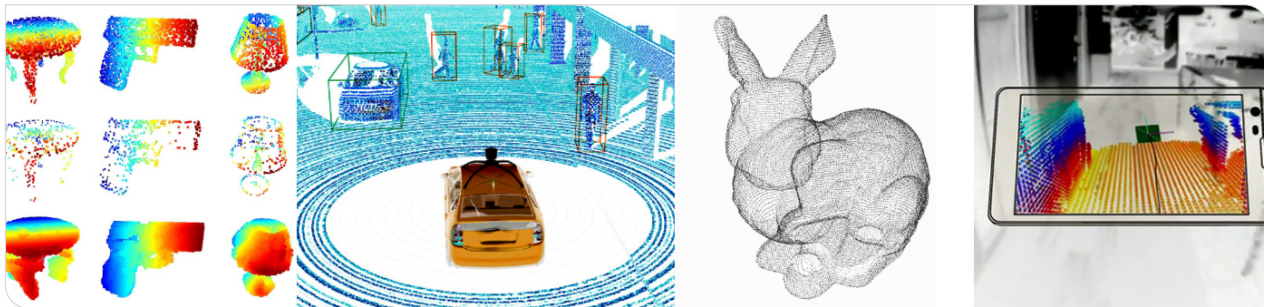
source: solidworks

A number of emerging 3D applications shape the need for 3D deep learning.

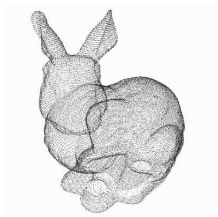
PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation

Felix Karg | 29. Juni 2022

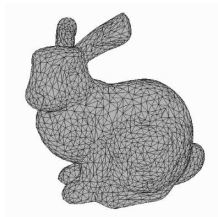
Betreuer: Antonio Zea



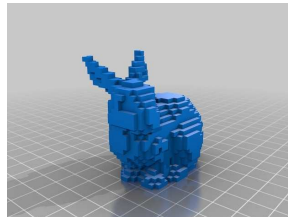
Common Representations of 3D Data



Point Cloud



Mesh



Volumetric



View Rendering

Contrary to 2D, 3D has many different popular representations.

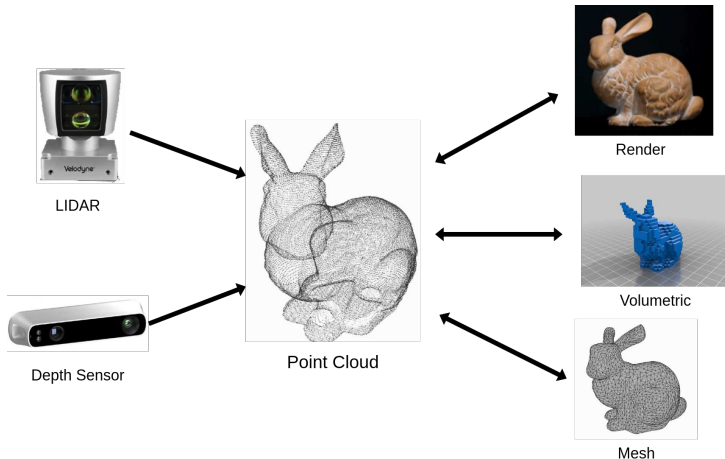
Figures and captions (partially) from CVPR presentation to [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
●○	○○○	○○	○○○○○	○○○	○○	○○○○○	○○○○○	○○	○○

Canonical Representation: Point Cloud

- Point cloud is close to **raw depth sensor data**
- Point cloud is **canonical** (easy conversion from and to other representations)

Individual figures from CVPR presentation to [Qi+17a]



Representation
●

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○○

Point Cloud Features

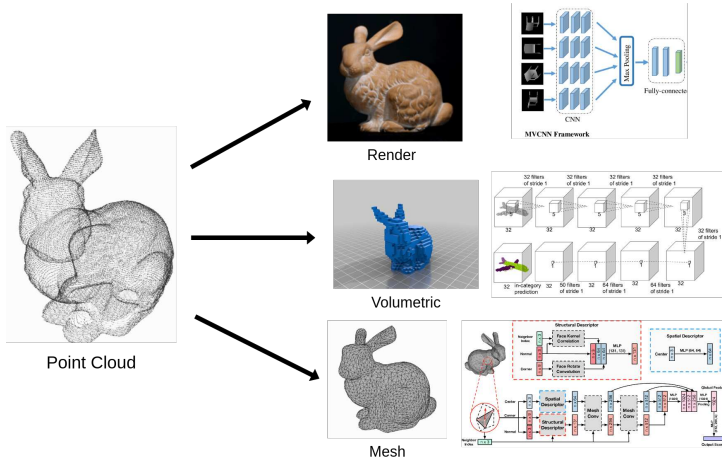
Feature Name	Supports Texture / Color	Local / Global / Regional	Best Use Case
PFH	No	L	2.5D Scans (Pseudo single position range images)
FPH	No	L	
VFH	No	G	Object detection with basic pose estimation
CVFH	No	R	Object detection with basic pose estimation, detection of partial objects
RIFT	Yes	L	Real world 3D-Scans with no mirror effects. RIFT is vulnerable against flipping.
RSD	No	L	
NARF	No	L	2.5D (Range Images)
ESF	No	G	

Overview from <https://github.com/PointCloudLibrary/pcl/wiki/Overview-and-Comparison-of-Features>

Most existing point cloud features are **handcrafted for specific tasks**.

Representation ○○	Related Work ●○○	PointNet ○○	Unordered Input ○○○○○	Geometric Transformation ○○○	Architecture ○○	Results ○○○○○	Visualization ○○○○○	Impact ○○	Conclusion ○○
----------------------	---------------------	----------------	--------------------------	---------------------------------	--------------------	------------------	------------------------	--------------	------------------

Conversion to Other Representations



Figures from:

- Bunnies: CVPR presentation to [Qi+17a]
- MVCNN: [LXL19]
- 3D-CNN: Supplemental to [Qi+17a]
- Mesh-Net: [Fen+19]

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
○○	●●○	○○	○○○○○	○○○	○○	○○○○○	○○○○○	○○	○○

Research Question:

Can we achieve effective **feature learning directly** on point clouds?

Representation
○○

Related Work
○○●

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○○

Introduction to PointNet

- End-to-end learning for unordered point cloud data
- Unified framework for previously separate and specialized tasks
 - Object Classification
 - Object Part Segmentation
 - Semantic Scene parsing

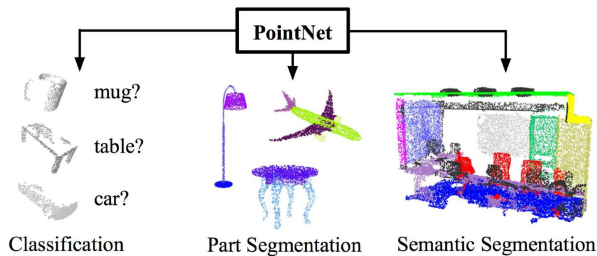
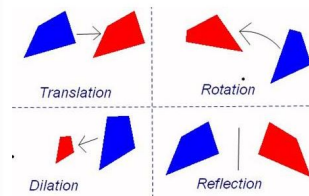
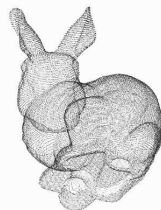


Figure from [Qi+17a].

Challenges

- Unordered point sets as input
 - Model needs to be invariant to $N!$ permutations
- Invariance under geometric transformations
 - Geometric transformations applied to point cloud data should not alter classification results



Point cloud figure from CVPR presentation to [Qi+17a].

Geometric transformation figure from [j2t19].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	o●	ooooo	ooo	oo	oooooo	ooooo	oo	oo

Unordered Point Sets

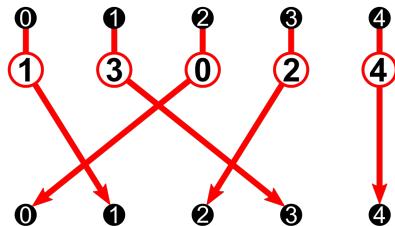
A set of points $p_i := (x_i, y_i, z_i)$

$$\{p_1, p_2, \dots, p_n\}$$

might be represented by any of its vector

permutations $[p_{\pi_1}, p_{\pi_2}, \dots, p_{\pi_n}]$ for any permutation π .

Since point cloud data is orderless, it requires invariance over input permutations when consumed directly.



Example Permutation.

Figure under CC-BY-SA 4.0 from [Wat22]

Solution: Symmetric Functions

Symmetric functions are invariant over argument permutations π :

$$f(x_1, x_2, \dots, x_n) \equiv f(x_{\pi_1}, x_{\pi_2}, \dots, x_{\pi_n})$$

Examples for symmetric functions:

- max
- sum / addition
- mean

Q: How to integrate a symmetric function into a neural network architecture?

One Symmetric Function is All You Need

A concatenation of functions $(\gamma \circ g(h, \dots))$ is symmetric if the central function g is symmetric:

$$f(x_1, x_2, \dots, x_n) = \gamma \circ g(h(x_1), \dots, h(x_n))$$

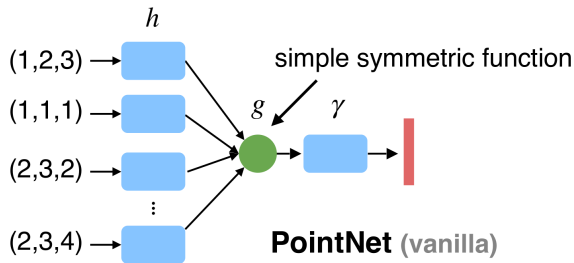


Figure from CVPR presentation to [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○●○○

Geometric Transformation
○○○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○○

Universal Set Function Approximation

PointNet (vanilla) is a universal set function approximator.

Theorem

A Hausdorff continuous symmetric function $f : 2^X \mapsto \mathbb{R}$ can be arbitrarily approximated by PointNet.

$$\left| f(S) - \gamma \left(\underbrace{g_{x_i \in S} \{h(x_i)\}}_{\text{PointNet (vanilla)}} \right) \right| < \varepsilon$$

with $S \subseteq \mathbb{R}^d$

For details see [Qi+17a] and supplemental material.

Basic PointNet Architecture

In practice, both h and γ are **multi-layer perceptrons (MLP)** as generic function approximators. Empirically, **max pooling** provides the best results as symmetric function:

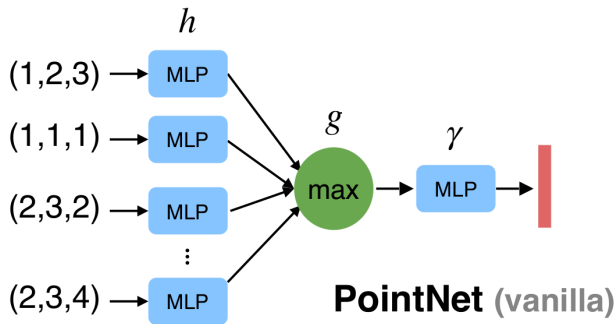


Figure from CVPR presentation to [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	oooo●	ooo	oo	oooooo	ooooo	oo	oo

Geometric Transformations

In particular, point cloud classification should be invariant to:

- Translation
- Rotation
- Scaling (Dilation)

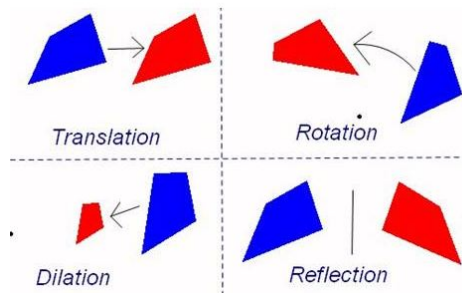
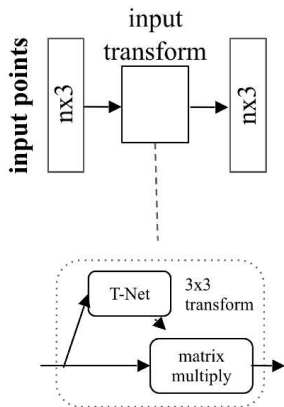


Figure from [i2t19].

Input Alignment by Transformer Network



Solution

Have a transformer network (T-Net) figure out data-dependent transformations.

A T-Net is a PointNet (vanilla) with a matrix as output.

Additionally, regularize matrix close to orthogonal:

$$L_{reg} = ||I - AA^T||_F^2$$

Figure from CVPR presentation to [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
●●○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○○

Effects of T-Net and Regularization

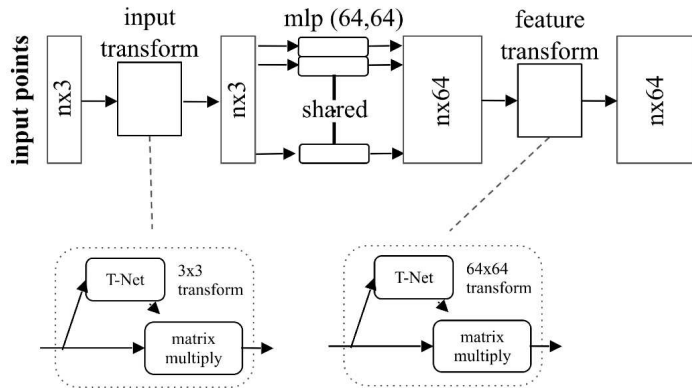


Figure from CVPR presentation to [Qi+17a].

Transform	accuracy
none	87.1
input (3x3)	87.9
feature (64x64)	86.9
feature (64x64) + reg.	87.4
both	89.2

Effects of input feature transforms. Based on overall classification accuracy on the ModelNet40 [Wu+15] test set. Table from [Qi+17a].

PointNet Classification Network

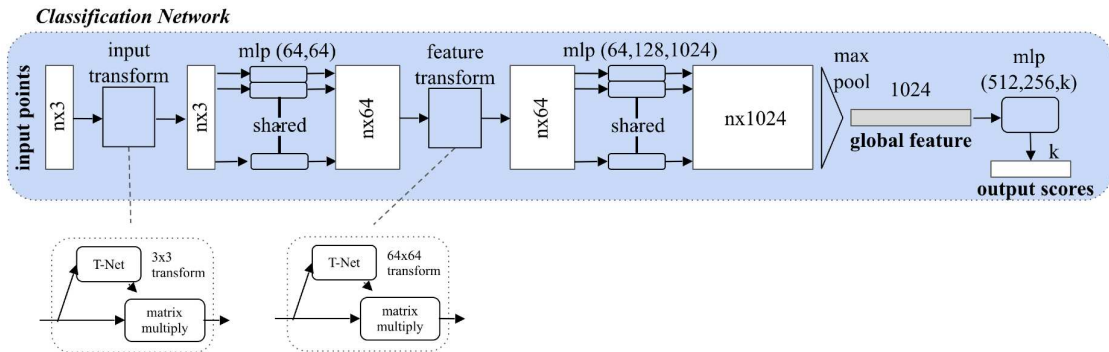


Figure from CVPR presentation to [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	●o	oooooo	ooooo	oo	oo

Extension to PointNet Segmentation Network

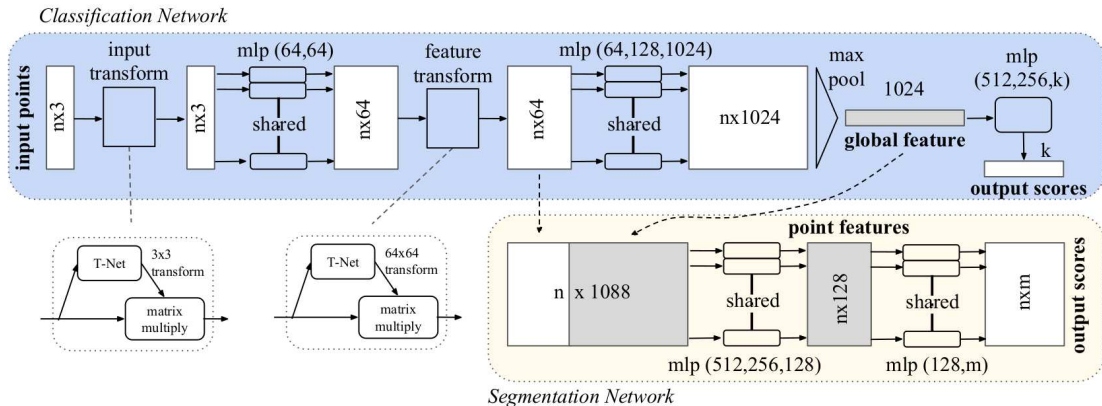


Figure from [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	o●	oooooo	ooooo	oo	oo

Results on Object Classification

	input	#views	accuracy avg. class	accuracy overall
SPH [KFR03]	mesh	-	68.2	-
3DShapeNets [Wu+15]	volume	1	77.3	84.7
VoxNet [MS15]	volume	12	83.0	85.9
Subvolume [Qi+16]	volume	20	86.0	89.2
LFD [Wu+15]	image	10	75.5	-
MVCNN [Su+15]	image	80	90.1	-
Ours baseline	point	-	72.6	77.4
Ours PointNet	point	1	86.2	89.2

Classification results on ModelNet40. PointNet achieves state-of-the-art among deep nets on 3D input. Table from [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

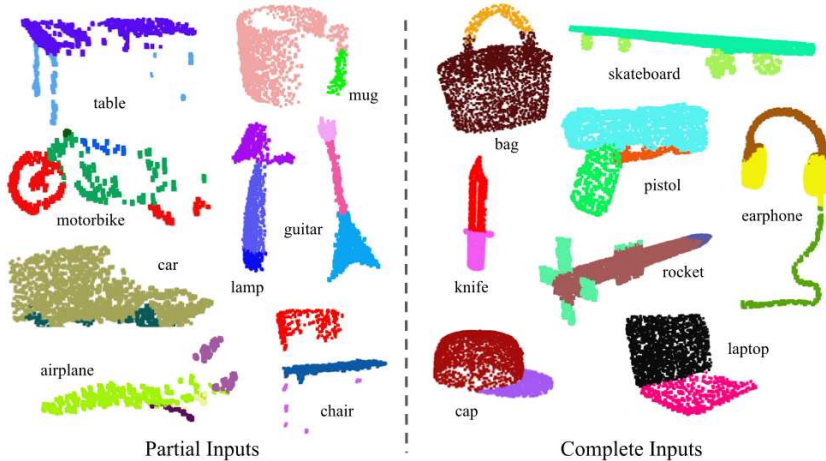
Results
●○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○○

Visualization of Object Part Segmentation



Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	●oooo	ooooo	oo	oo

Results on Object Part Segmentation

	mean	aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skate board	table
# shapes		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
Wu [24]	-	63.2	-	-	-	73.5	-	-	-	74.4	-	-	-	-	-	-	74.8
Yi [26]	81.4	81.0	78.4	77.7	75.7	87.6	61.9	92.0	85.4	82.5	95.7	70.6	91.9	85.9	53.1	69.8	75.3
3DCNN	79.4	75.1	72.8	73.3	70.0	87.2	63.5	88.4	79.6	74.4	93.9	58.7	91.8	76.4	51.2	65.3	77.1
Ours	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6

Segmentation results on ShapeNet part dataset. The metric used is mIoU(%) on points. Figure/Table from [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

Results
○○●○○○

Visualization
○○○○○

Impact
○○

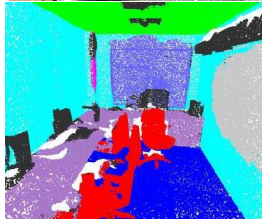
Conclusion
○○

Semantic Scene Parsing

■ Input



■ Output



Figures from [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

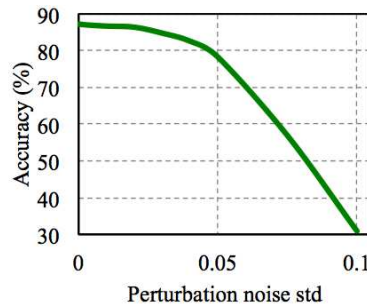
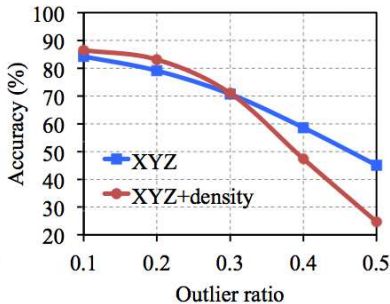
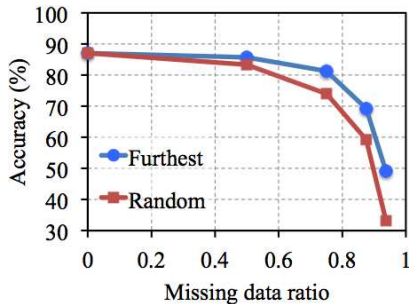
Results
○○○●○○

Visualization
○○○○○

Impact
○○

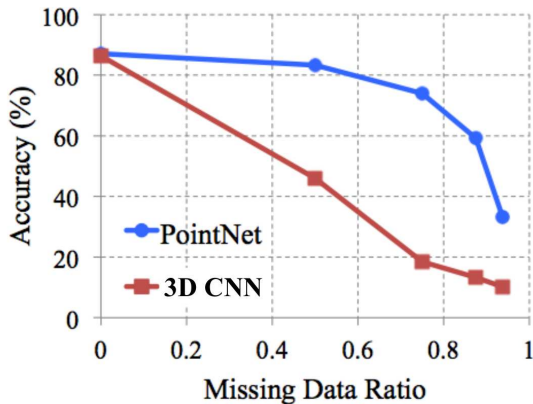
Conclusion
○○

Robustness to Data Corruption



Robustness tests. Accuracy measured on ModelNet40. Figure from [Qi+17a].

Robustness in comparison



Q: Why is PointNet so robust to missing data?

Robustness in comparison with 3D CNN. Figure from CVPR presentation to [Qi+17a].

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

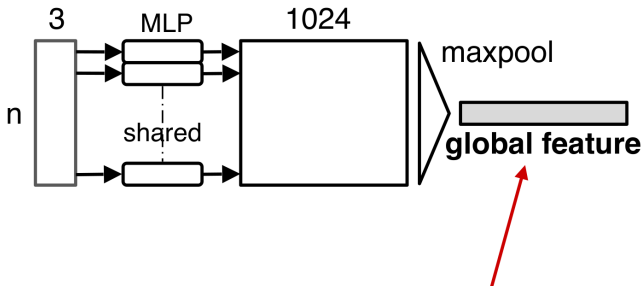
Results
○○○○○●

Visualization
○○○○○

Impact
○○

Conclusion
○○

Visualizing Global Point Cloud Features



Which points contribute to the global feature vector? (**critical points**)
Which additional points won't affect the global feature vector? (**upper bound**)

Figure from CVPR presentation to [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	●oooo	oo	oo

Visualizing Global Point Cloud Features

- Original Shape
- Critical Point Set
- Upper Bound Set

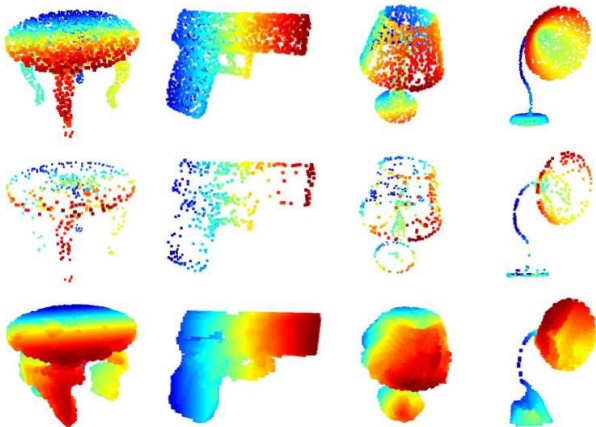


Figure from [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	o●ooo	oo	oo

Visualizing Global Point Cloud Features (OOS)

- Original Shape
- Critical Point Set
- Upper Bound Set

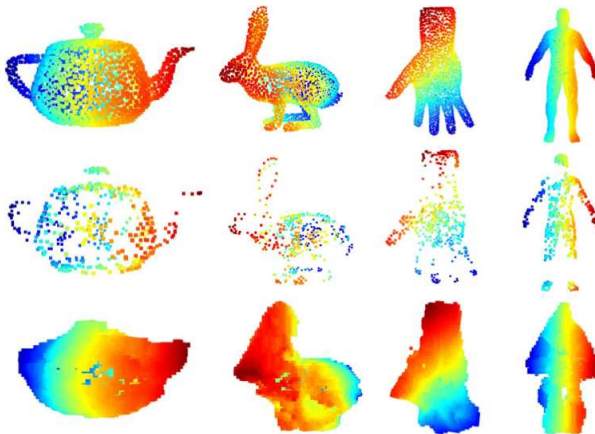
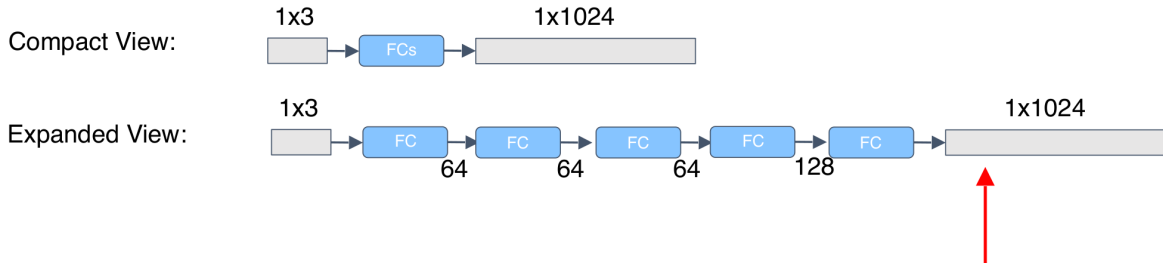


Figure from [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	oo●oo	oo	oo

Approach to Features Visualization



Which input point will activate neuron X?

Find the top-K points in a dense volumetric grid that activates neuron X.

Figure from CVPR presentation to [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	ooo●oo	oo	oo

Selective Visualization of Activation Features

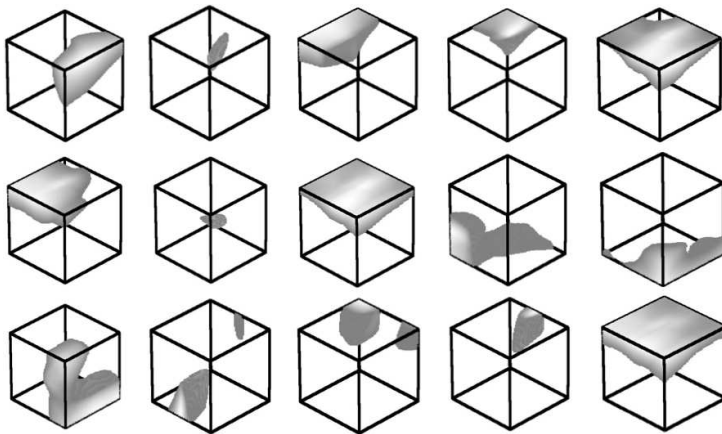


Figure from [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	oooo●	oo	oo

Derivative Works of PointNet

Pointnet: Deep learning on point sets for 3d classification and segmentation

[PDF] thecvf.com

[CR Qi](#), [H Su](#), [K Mo](#), [LJ Guibas](#) - Proceedings of the IEEE ..., 2017 - [openaccess.thecvf.com](#)

... Our network, named **PointNet**, provides a unified architecture for applications ranging from object classification, part segmentation, to scene semantic parsing. Though simple, **PointNet** ...

☆ Save 📄 Cite Cited by 7847 Related articles All 18 versions 🔗

Core architecture ideas were adapted in:

- A sift-like network module [JWL18]
- Similarity group proposal network [Wan+18]
- Point cloud upsampling [Yu+18]
- Application to Neuroanatomy [GW18]
- Frustum pointnets [Qi+18]
- Pointcnn [Li+18]
- many more ...

Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

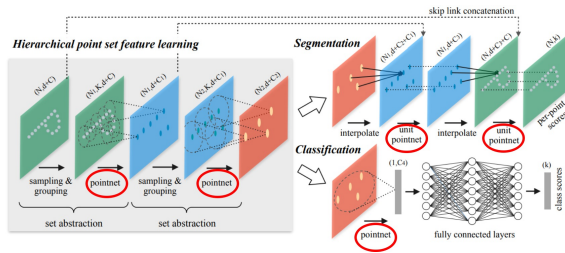
Impact
●○

Conclusion
○○

Derivative Works of PointNet II

PointNet has been used as a module in:

- PointNet++ [Qi+17b]
- SyncSpecCNN [Yi+17]
- VoxelNet [ZT18]
- ...



Architecture of PointNet++ with highlighted PointNet layers.
Figure adapted from PointNet++ [Qi+17b]

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
○○	○○○	○○	○○○○○	○○○	○○	○○○○○	○○○○○	●●	○○

Conclusion

- PointNet is a novel deep neural network directly consuming point cloud data
- Enabling a unified approach to various 3D recognition tasks
- Task performance is on par or better than state of the art
- PointNet saw usage as a module in other architectures
- Core ideas (symmetry, T-Nets, ...) have been adapted too

Paper, code, presentation and slides are available at <https://stanford.edu/~rqi/pointnet>

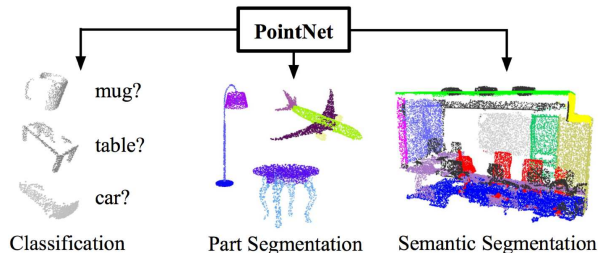
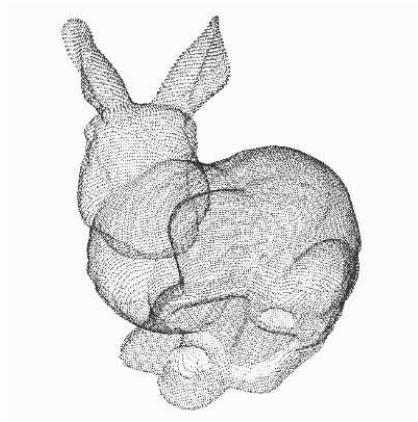


Figure from [Qi+17a].

Representation	Related Work	PointNet	Unordered Input	Geometric Transformation	Architecture	Results	Visualization	Impact	Conclusion
oo	ooo	oo	ooooo	ooo	oo	oooooo	ooooo	oo	●o

What are your Questions?



Representation
○○

Related Work
○○○

PointNet
○○

Unordered Input
○○○○○

Geometric Transformation
○○○

Architecture
○○

Results
○○○○○

Visualization
○○○○○

Impact
○○

Conclusion
○●

Sources I

- [1] aldiptoli. “pointnet”. In: <https://github.com/aldiptoli/pointnet> (Aug. 2021). [Online; accessed 4. Jun. 2022].
- [2] charlesq34. “pointnet”. In: <https://github.com/charlesq34/pointnet> (Sept. 2019). [Online; accessed 4. Jun. 2022].
- [3] Yutong Feng et al. “Meshnet: Mesh neural network for 3d shape representation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 8279–8286.
- [4] Benjamín Gutiérrez-Becker and Christian Wachinger. “Deep multi-structural shape analysis: application to neuroanatomy”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 523–531.
- [5] Shikun Huang et al. “A CLAIM Approach to Understanding the PointNet”. In: *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*. 2019, pp. 97–103.

Sources II

- [6] i2tutorials. *What Is Geometric Transformation?* [Online; accessed 27. Jun. 2022]. Oct. 2019. URL: <https://www.i2tutorials.com/what-is-geometric-transformation>.
- [7] Mingyang Jiang, Yiran Wu, and C PointSIFT Lu. “A sift-like network module for 3d point cloud semantic segmentation”. In: *Comput. Vis. Pattern Recognit.* 2018.
- [8] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. “Rotation invariant spherical harmonic representation of 3 d shape descriptors”. In: *Symposium on geometry processing*. Vol. 6. 2003, pp. 156–164.
- [9] Yangyan Li et al. “Pointcnn: Convolution on x-transformed points”. In: *Advances in neural information processing systems* 31 (2018).
- [10] Zhaoqun Li, Cheng Xu, and Biao Leng. “Angular triplet-center loss for multi-view 3d shape retrieval”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 8682–8689.

Sources III

- [11] Ming Liang et al. “Multi-task multi-sensor fusion for 3d object detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7345–7353.
- [12] Daniel Maturana and Sebastian Scherer. “Voxnet: A 3d convolutional neural network for real-time object recognition”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 922–928.
- [13] Damian Mrowca et al. “Flexible neural representation for physics prediction”. In: *Advances in neural information processing systems* 31 (2018).
- [14] Charles R Qi et al. “Frustum pointnets for 3d object detection from rgb-d data”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 918–927.
- [15] Charles R Qi et al. “Pointnet: Deep learning on point sets for 3d classification and segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.

Sources IV

- [16] Charles R Qi et al. “Volumetric and multi-view cnns for object classification on 3d data”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5648–5656.
- [17] Charles Ruizhongtai Qi et al. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”. In: *Advances in neural information processing systems 30* (2017).
- [18] Hang Su et al. “Multi-view convolutional neural networks for 3d shape recognition”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 945–953.
- [19] SuperDataScience Team. *Convolutional Neural Networks (CNN): Step 4 - Full Connection*. [Online; accessed 29. Jun. 2022]. Aug. 2018. URL: <https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-4-full-connection>.
- [20] Alexandre H Thiery et al. “Medical Application of Geometric Deep Learning for the Diagnosis of Glaucoma”. In: *arXiv preprint arXiv:2204.07004* (2022).

References	Multi-Layer Perceptron ○	Related ○	Complexity ○	Permutation Invariance ○
------------	-----------------------------	--------------	-----------------	-----------------------------

Sources V

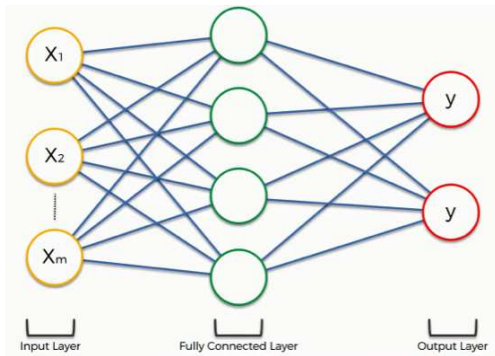
- [21] Larissa T Triess et al. “A survey on deep domain adaptation for lidar perception”. In: *2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops)*. IEEE. 2021, pp. 350–357.
- [22] Weiyue Wang et al. “Sgpn: Similarity group proposal network for 3d point cloud instance segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2569–2578.
- [23] Watchduck. *Permutations*. [Online; accessed 25. Jun. 2022]. June 2022. URL: https://en.wikiversity.org/wiki/File:5-el_perm%3B_active%3B_only_P_as_arrow_diagram.svg.
- [24] Zhirong Wu et al. “3d shapenets: A deep representation for volumetric shapes”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1912–1920.
- [25] Xu Yan. “Pointnet/Pointnet++ Pytorch”. In: https://github.com/yanx27/Pointnet_Pointnet2_pytorch (June 2019). [Online; accessed 4. Jun. 2022].

Sources VI

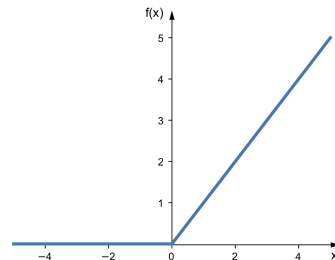
- [26] Li Yi et al. “Syncspecnn: Synchronized spectral cnn for 3d shape segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2282–2290.
- [27] Lequan Yu et al. “Pu-net: Point cloud upsampling network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2790–2799.
- [28] yunxiaoshi. “pointnet-pytorch”. In: <https://github.com/yunxiaoshi/pointnet-pytorch> (June 2021). [Online; accessed 4. Jun. 2022].
- [29] Binbin Zhang et al. “Explaining the PointNet: What Has Been Learned Inside the PointNet?” In: *CVPR Workshops*. 2019, pp. 71–74.
- [30] Weichen Zhang et al. “Collaborative and adversarial network for unsupervised domain adaptation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 3801–3809.
- [31] Yin Zhou and Oncel Tuzel. “Voxelnet: End-to-end learning for point cloud based 3d object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4490–4499.

References	Multi-Layer Perceptron ○	Related ○	Complexity ○	Permutation Invariance ○
------------	-----------------------------	--------------	-----------------	-----------------------------

Multi-Layer Perceptron



Multi-Layer Perceptron with one fully connected layer. Alternative names include 'dense', 'fully connected' and 'mlp' layer. Figure from [Sup18].



Common activation function:
ReLU, short for Rectified Linear Unit.

References

Multi-Layer Perceptron



Related



Complexity



Permutation Invariance



Based on PointNet

A number of works build on PointNet [Qi+17a]:

- Implementations and tools for visualization: [cha19; ald21; yun21; Yan19]
- Further attempts at explaining what PointNet learned: [Zha+19; Hua+19]
- Application of PointNet to different domains and problems: [Thi+22; GW18; Tri+21; Lia+19; Zha+18; Mro+18]

Speed and Model Size

	#params	FLOPs/sample
PointNet (vanilla)	0.8M	148M
PointNet	3.5M	440M
Subvolume [Qi+16]	16.6M	3633M
MVCNN [Su+15]	60.0M	62057M

Time and space complexity of different deep learning architectures for 3D data classification. PointNet (vanilla) is the classification PointNet without input and feature T-Net transformation networks. FLOP is floating-point operations. The “M” stands for a million units. Both Subvolume and MVCNN used input data pooling from multiple rotations or views, without which they have much inferior performance. Table from [Qi+17a].

Permutation Invariance: Sorting

Unfortunately, there is no canonical order in high dim space.

	Accuracy
Unordered Input	12%
Lexsorted Input	40%
LSTM	75%
PointNet (vanilla)	87%

Validation on the ModelNet40 dataset. Table from CVPR presentation to [Qi+17a].