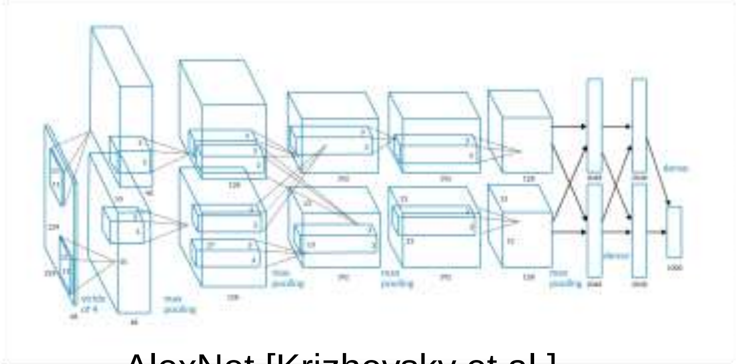
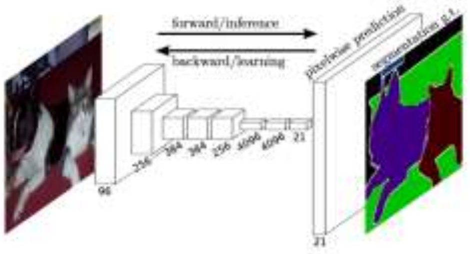
# High Dimensional Convolutional Neural Networks for 3D perception

Chris Choy,
Ph.D. candidate @ Stanford Vision and Learning Lab

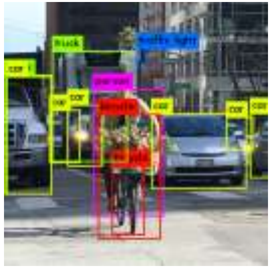
# The Success of Convolutional Networks



AlexNet [Krizhevsky et al.]




FCNN [Long et al.]



R-CNN [Girshick et al.]

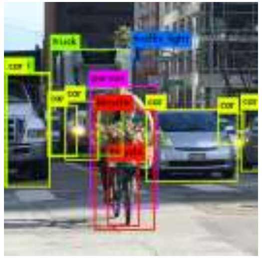
GAN [Goodfellow et al.]

# The Success of Convolutional Networks

Versatility

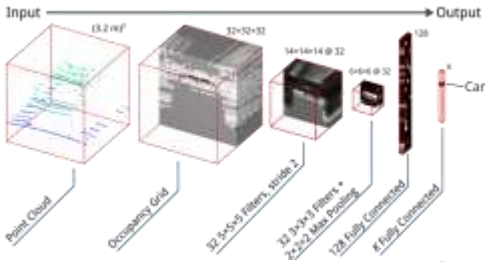
Experience

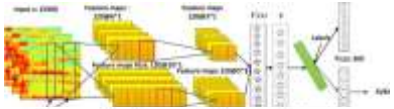
Efficiency

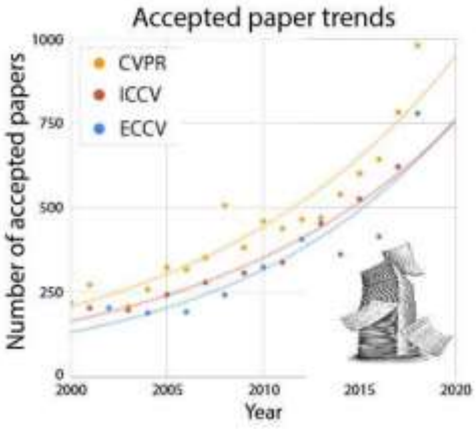


Object Detection

Semantic Segmentation





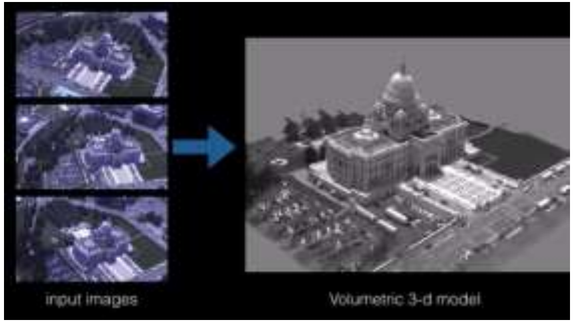

TensorFlow

PyTorch

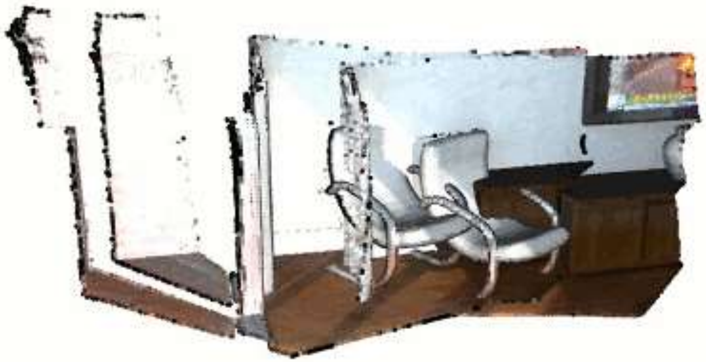
mxnet

cuDNN

# Examples of 3D Vision Tasks




3D Reconstruction



3D Registration

3D Object Pose Estimation


3D Object Tracking

# 3D Vision in Action

Nvidia Research, 2019

Microsoft HoloLens

Amazon AR View

## 3D Reconstruction

Supervised Reconstruction

## 3D Perception

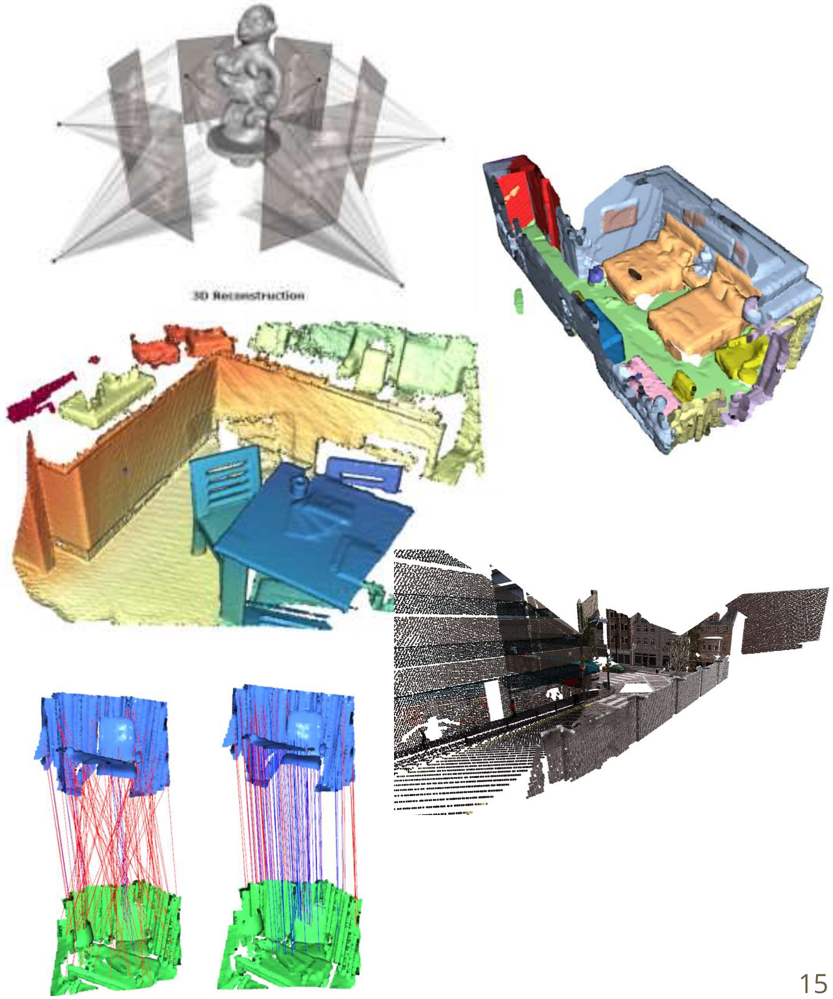
3D Semantic Segmentation

3D Feature Learning

## Perception on a Set of 3D Data

4D Spatio-Temporal Perception

4D and 6D for Registration



## 3D Reconstruction

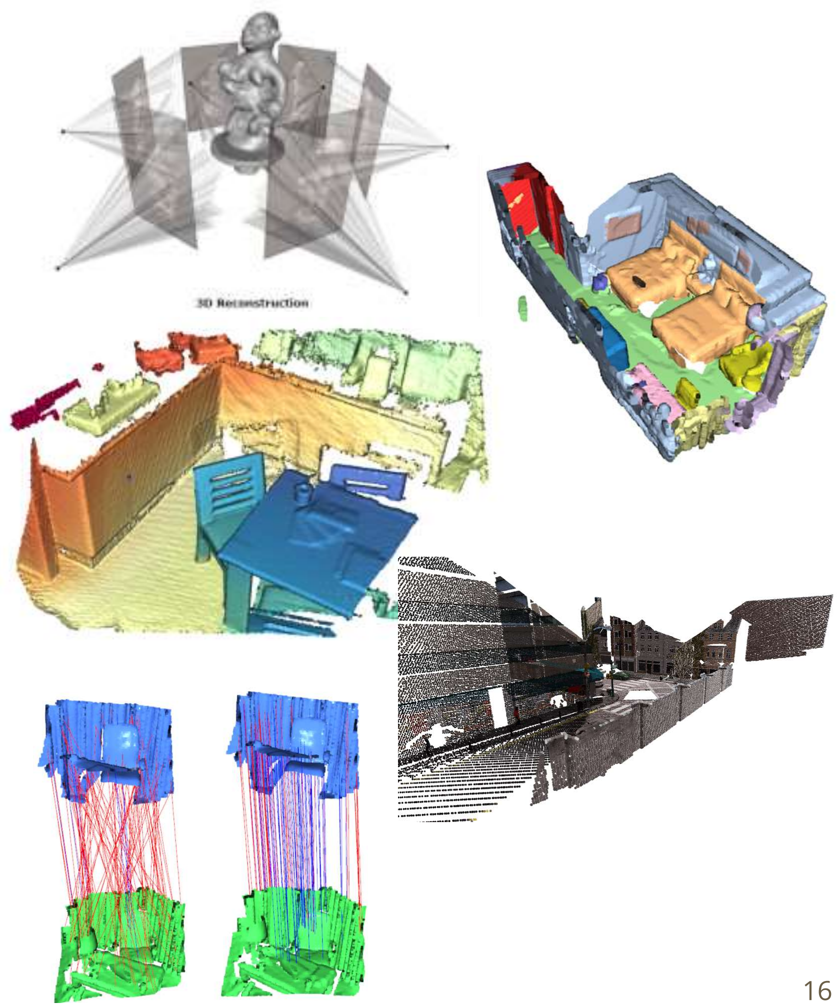
- Supervised Reconstruction

## 3D Perception

- 3D Semantic Segmentation
- 3D Feature Learning

## Perception on a Set of 3D Data

- 4D Spatio-Temporal Perception
- 4D and 6D for Registration



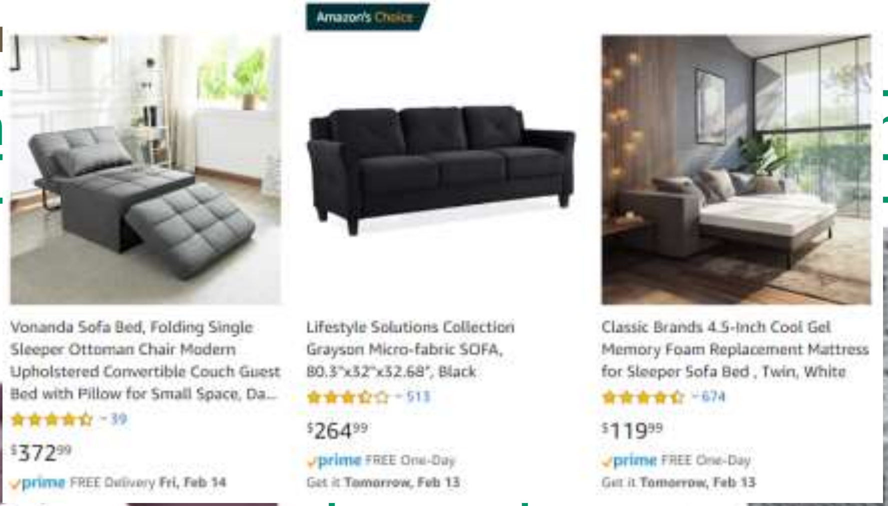
# 3D Reconstruction

- 3D-Recurrent Reconstruction Neural Networks, **Chris**, Danfei, JunYoung, Kevin, Silvio, ECCV’16
- Universal Correspondence Networks, **Chris**, JunYoung, Silvio, Manmohan, NIPS’16
- Weakly supervised 3D Reconstruction with Adversarial Constraint, JunYoung, **Chris**, Manmohan, Animehs, Silvio, 3DV’17
- DeformNet: Free-Form Deformation Network for 3D Shape Reconstruction from a Single Image, Andrey, Jingwei, Animesh, Viraj, JunYoung, **Chris**, Silvio, WACV’18
- Text2Shape: Generating Shapes from Natural Language by Learning Joint Embeddings, Kevin, **Chris**, Manolis, Angel, Thomas, Silvio, ACCV’18
- 4D-Spatio Temporal ConvNets: Minkowski Convolutional Neural Networks, **Chris**, JunYoung, Silvio, CVPR’19

# 3D Reconstruction from Few Images

- Single or Multi-view images of an object
- Online retail sto

Input Im




nstruction

# 3D Reconstruction from Few Images

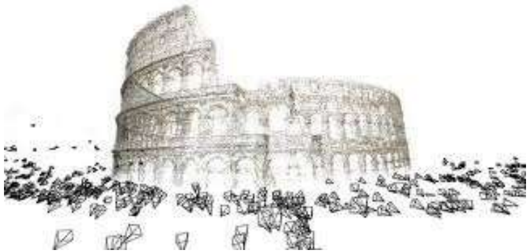
- Wide baseline
- Specular / texture-less region
- Single view

# 3D Reconstruction

Observations (Images)

Algorithms

Structure from Motion



[Longuet-Higgins, Haming et al., Snavely et al., …]

Depth Estimation

[Eigen et al., Saxena et al., …]

MVS

Tomography

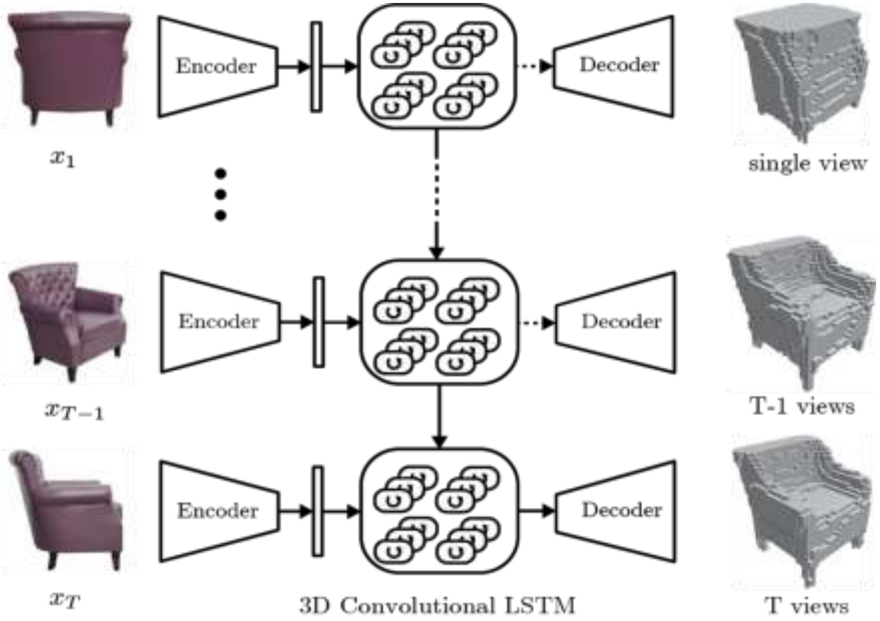
Object-centric Reconstruction

…

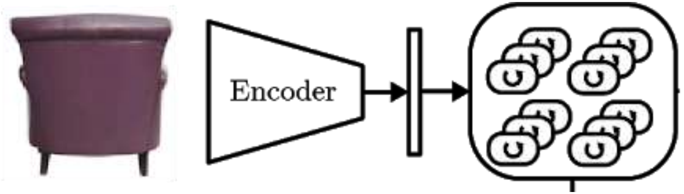
3D Representation

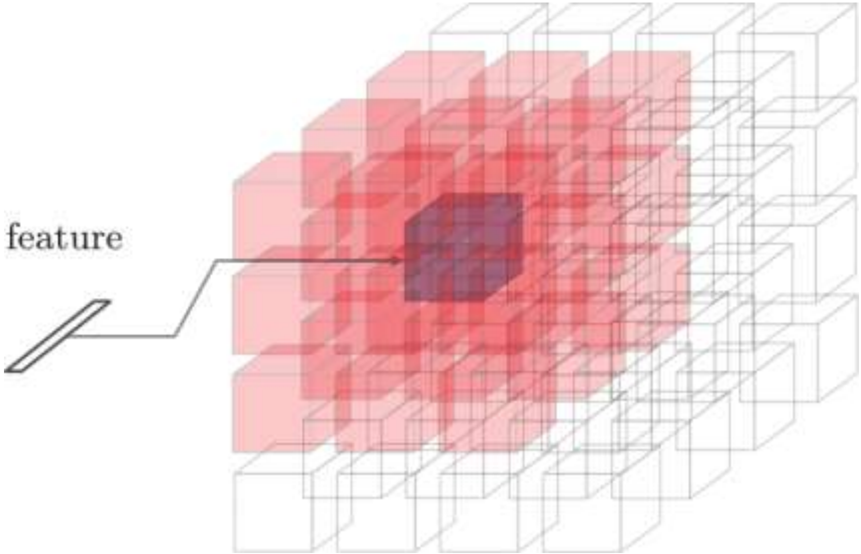
# 3D Recurrent Reconstruction Neural Networks

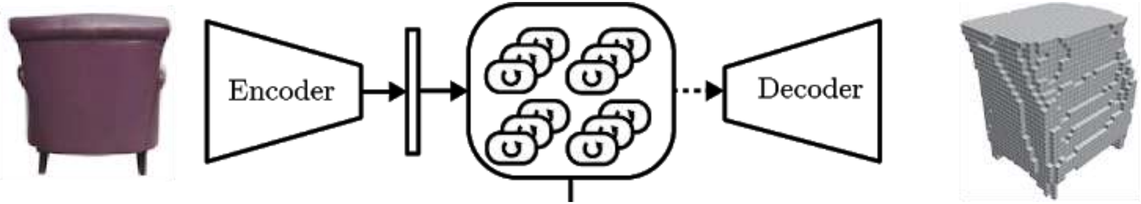
- End-to-end 3D reconstruction
- Unified framework
  - Single-view & Multi-view reconst.
- 3D-Convolutional LSTM
  - Update hidden states

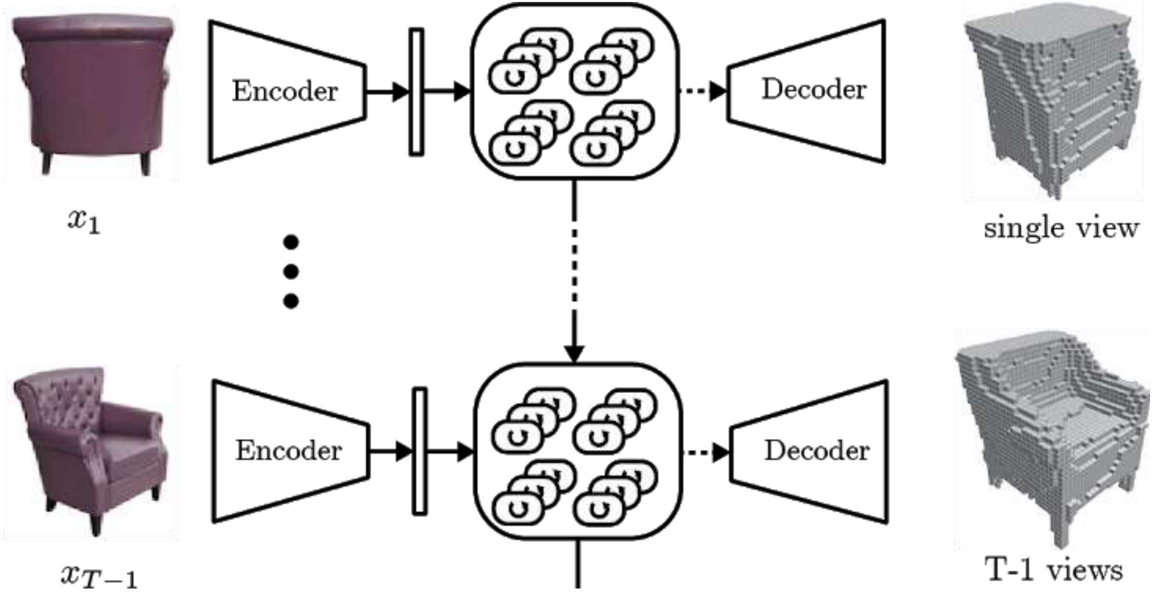


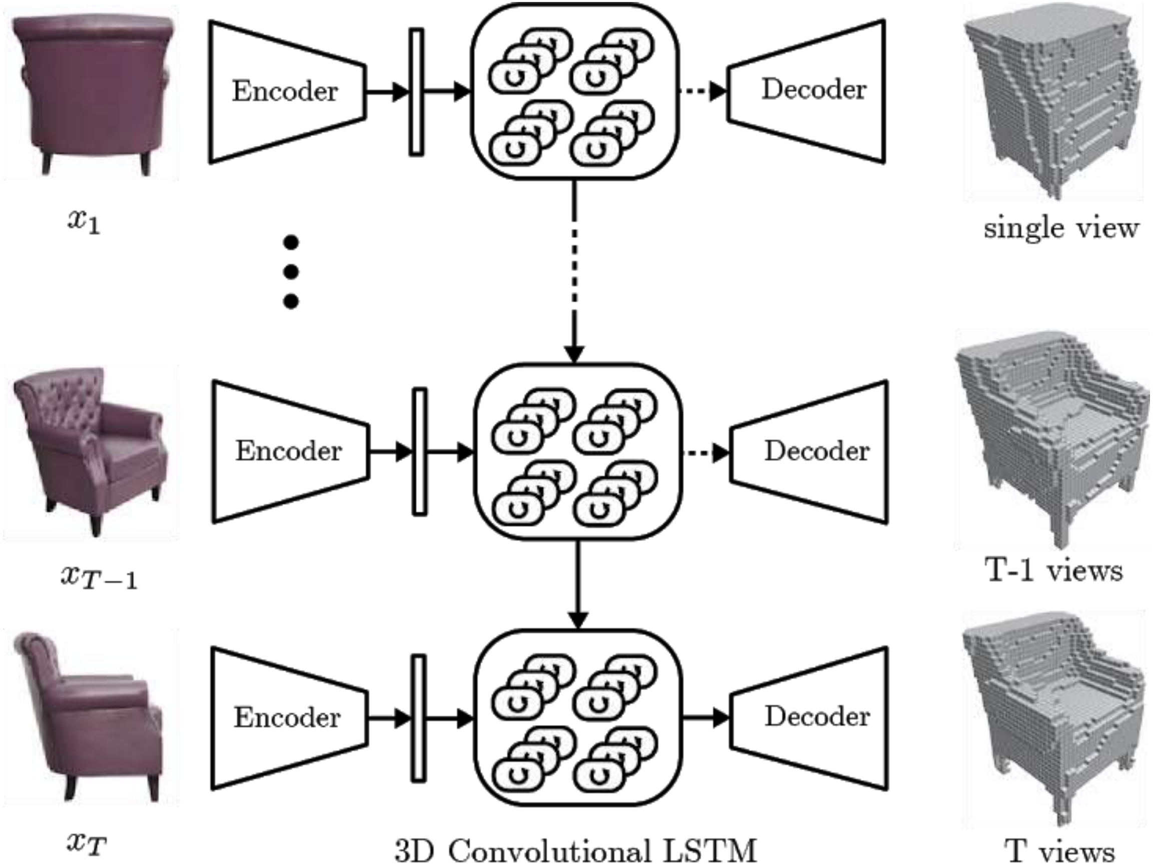

Encoder

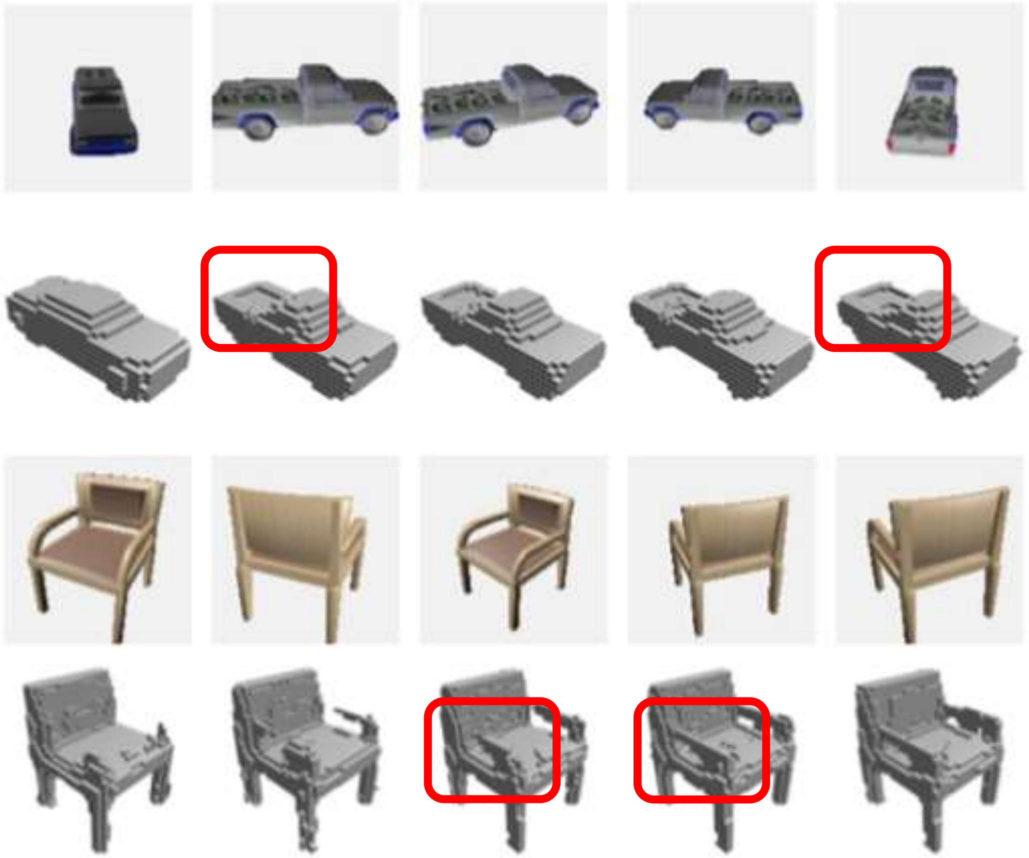

Encoder


feature


Encoder
Decoder


Encoder
Decoder
single view
Encoder
Decoder
T-1 views
$x_1$
$x_{T-1}$


Encoder
Decoder
single view
Encoder
Decoder
T-1 views
Encoder
Decoder
T views
$x_1$
$x_{T-1}$
$x_T$
3D Convolutional LSTM



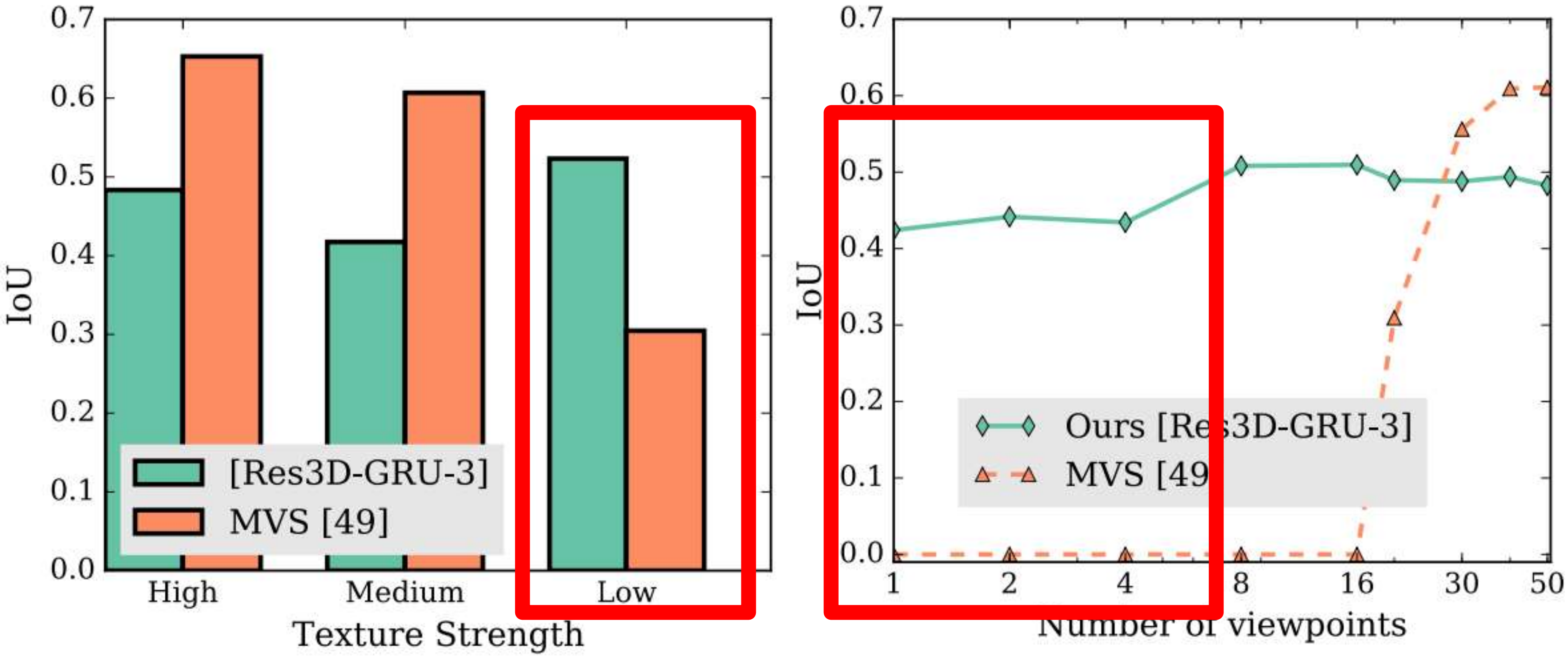
Update / maintain prediction

Increasing confidence on armrests

Number of images

Chris, Danfei, JunYoung, Kevin, Silvio, **3D-Recurrent Reconstruction Neural Networks**, ECCV'16

# Robustness to texture and # views




Chris, Danfei, JunYoung, Kevin, Silvio, **3D-Recurrent Reconstruction Neural Networks**, ECCV'16

## 3D Reconstruction

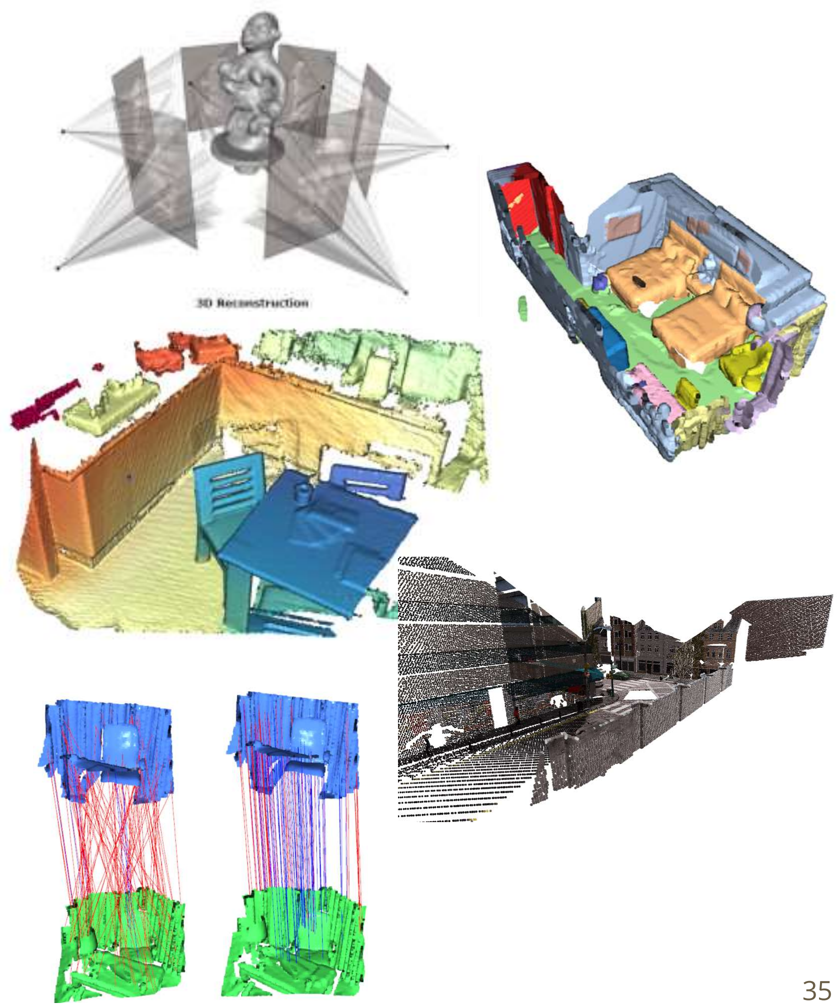
- Supervised Reconstruction

## 3D Perception

- 3D Semantic Segmentation
- 3D Feature Learning

## Perception on a Set of 3D Data

- 4D Spatio-Temporal Perception
- 4D and 6D for Registration



# 3D Perception

- SegCloud: Semantic Segmentation of 3D Point Clouds, Lyne, **Chris**, Iro, JunYoung Silvio, 3DV’17

- 4D-Spatio Temporal ConvNets: Minkowski Convolutional Neural Networks, **Chris**, JunYoung, Silvio, CVPR’19

- Fully Convolutional Geometric Features, **Chris**, Jaesik, Vladlen, ICCV’19

# Sparsity of 3D data

O($N^3$) volume vs. O($N^2$) surface

20cm voxel : 18%

10cm voxel : 9%

5cm voxel : 4.5%

2.5cm voxel : 1.8%

# Sparse Representations and Convolution

## Continuous Representation

Points and PointNet
[Qi et al.]

Occupancy Net
[Mescheder et al.]

Deep SDF
[Park et al.]

Deep Level Sets
[Michalkiewicz et al.]

Continuous Convolution
- PointCNN
- Monte Carlo Conv
- Surface / Tangent Conv

….

## Graph Representation

Graph Net
[Kipf & Wellings]

Conv on Graph
[Defferrard et al.]

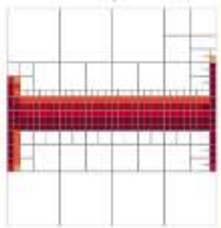
….

## Hybrid Representation

Contiuous + Graph

## Discrete Representation

OctNet and Octree
[Riegler et al.]



Sparse Tensor
[Graham et al., Choy et al.]

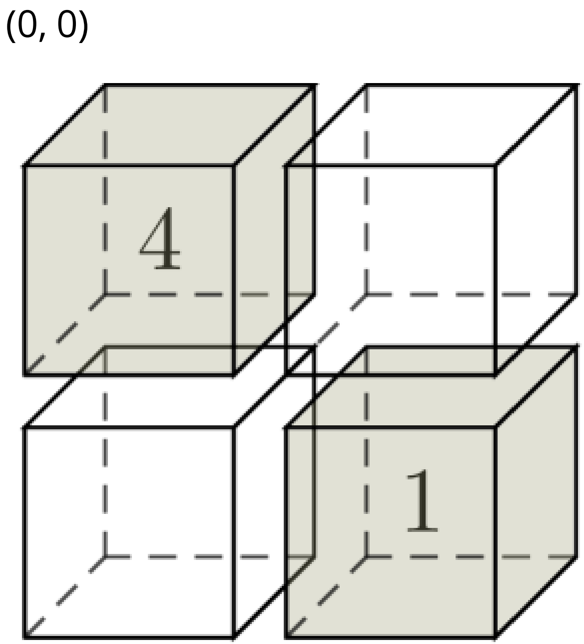
….

# Sparse Matrix

- Majority of elements are 0

- Efficient representation
    - Non-zero elements only
    - Compressed sparse row (CSR)
    - List of lists
    - COOrdinate list
    - Etc.

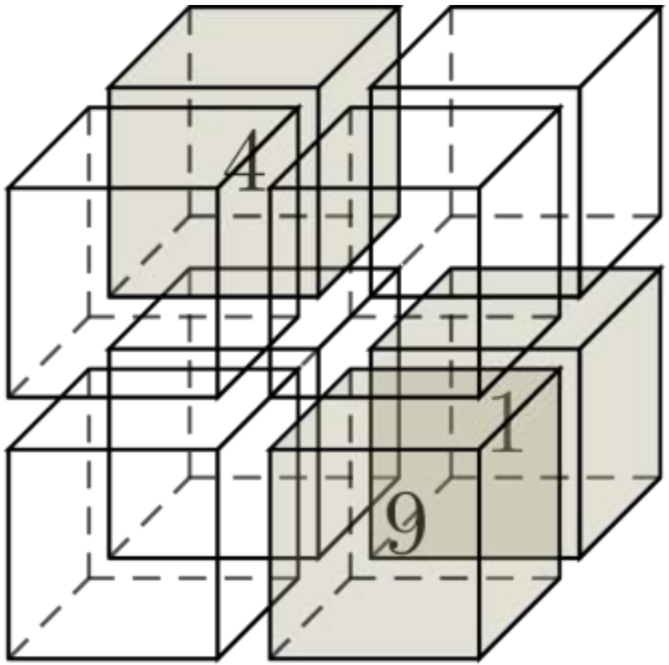
- Example: 2x2 matrix
    - COOrdinate (COO) representation
    - 4 at (0, 0)
    - 1 at (1, 1)



# Sparse Tensor

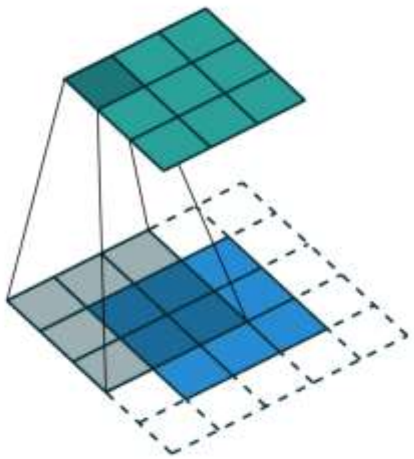
- High-dimensional extension
- COOrdinate representation
  - 4 at (0, 0, 0)
  - 1 at (1, 1, 0)
  - 9 at (1, 1, 1)

(0, 0, 0)

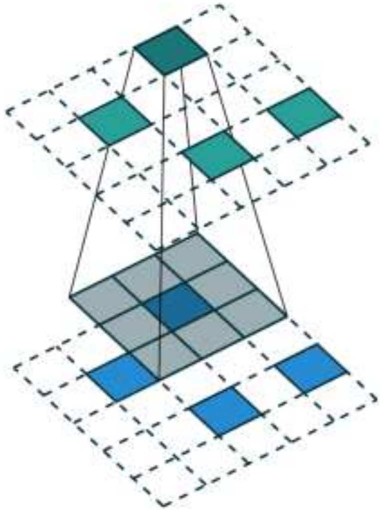


# Convolution on a Sparse Tensor

Convolution



Sparse Convolution



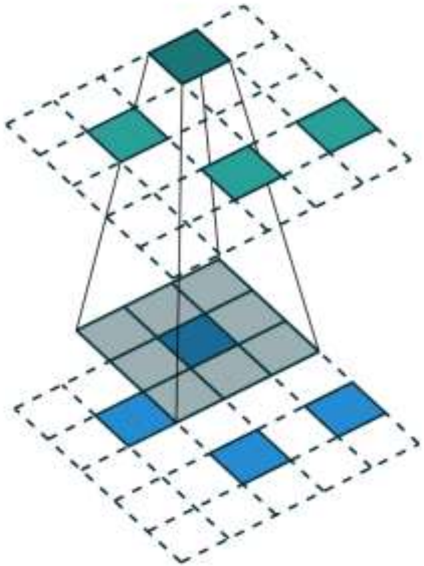
Cannot support arbitrary sparsity

Dense Tensor Kernel

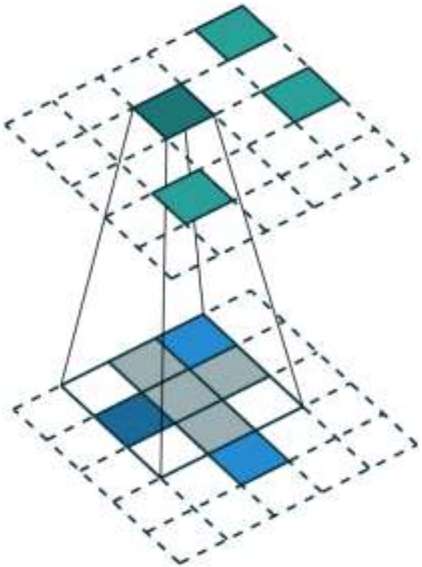
Static Sparsity Pattern

[Graham et al., Submanifold Sparse ConvNet, 2017]
[Graham and Maaten, 3D Sparse ConvNet, 2018]

# Generalized Convolution



[Graham et al.]



[Choy et al.]

Can support arbitrary sparsity

Sparse Tensor Kernel

Dynamic Sparsity Pattern

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR’19

# Generalized Convolution

Can support arbitrary sparsity → Sparsity pattern manipulation
Ex) C = A + B
Ex) Pruning

Sparse Tensor Kernel → High-dimensional ConvNet
Volume of dense convolution kernel: $O(N^D)$

Sparse convolution kernel: $O(D)$

Dynamic Sparsity Pattern → Generative Tasks

# Generalized Convolution: Special Cases

**Sparse Tensor Kernel**

- Dilated Convolution
- Separable Convolution
- Sparse Convolution

**Dynamic Sparsity Pattern**

- Octree Generative Networks

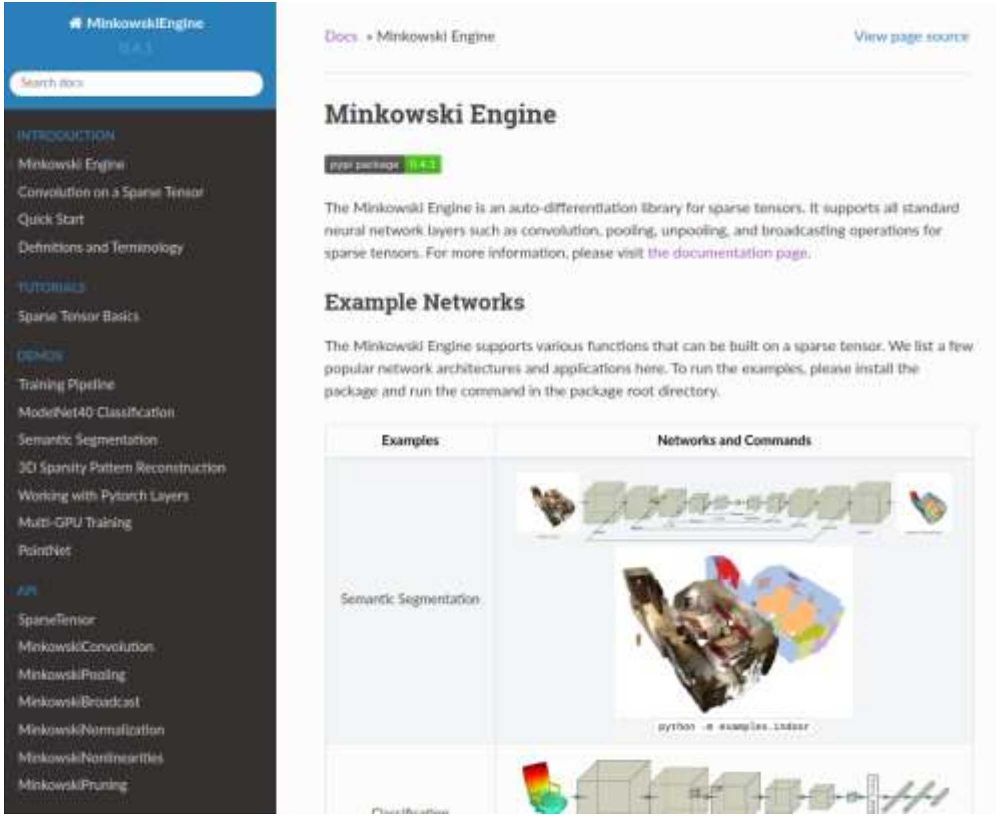
**Arbitrary sparsity**

- Dense Convolution

# Minkowski Engine

A convolutional neural network library for sparse tensors

- Convolution
- [Max/Avg/Global] Pool
- Broadcast
- [Batch/Instance] Normalization
- Tensor arithmetic
- Pruning
- …



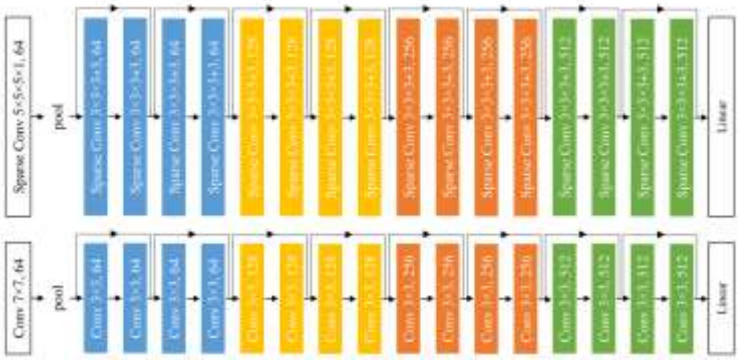
# Minkowski Network

- Very deep convolutional neural networks possible in 3D
  - 42-layer deep neural networks for semantic segmentation
  - 101 layers for classification

- Reuse network architectures from years of research in 2D

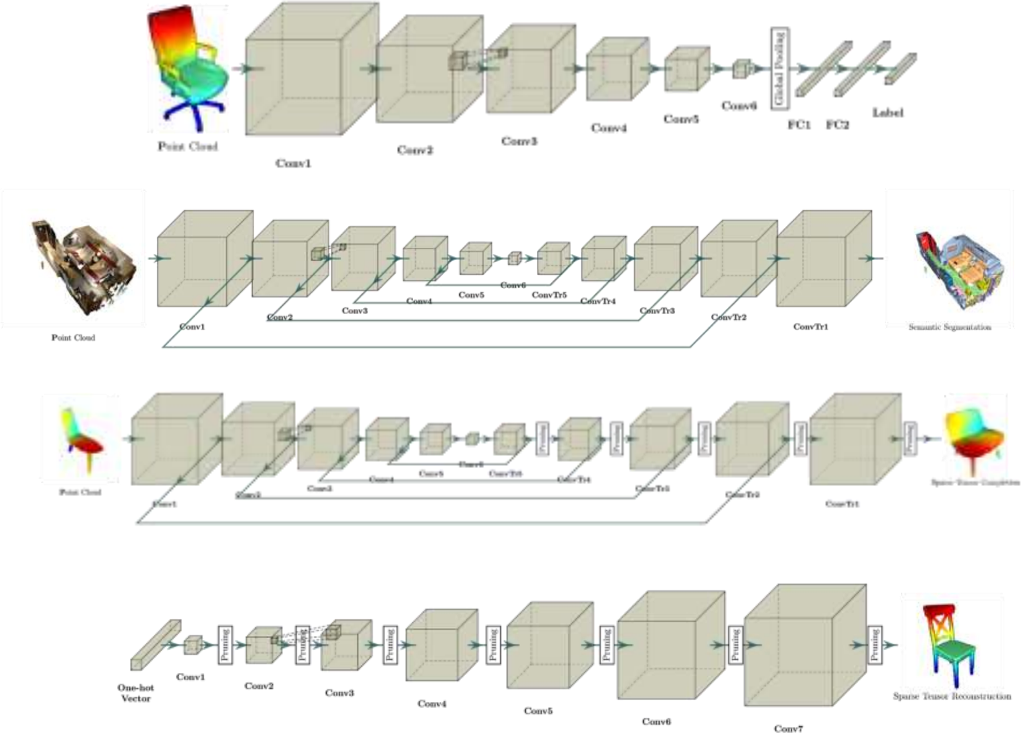
4D MinkNet18

ResNet18



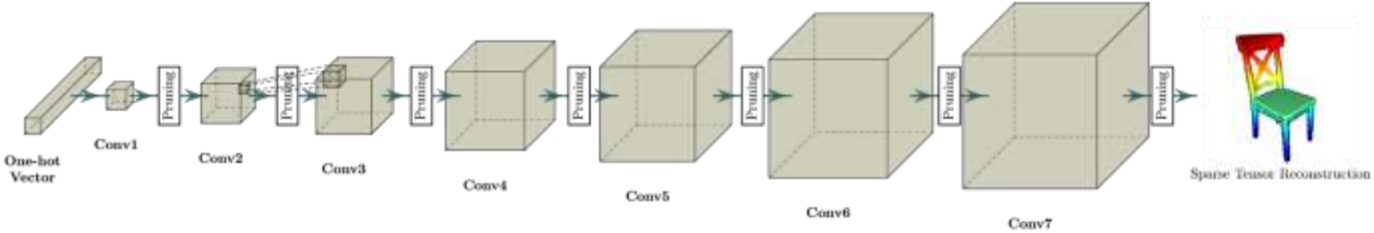
Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

# Minkowski Engine for other applications




Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

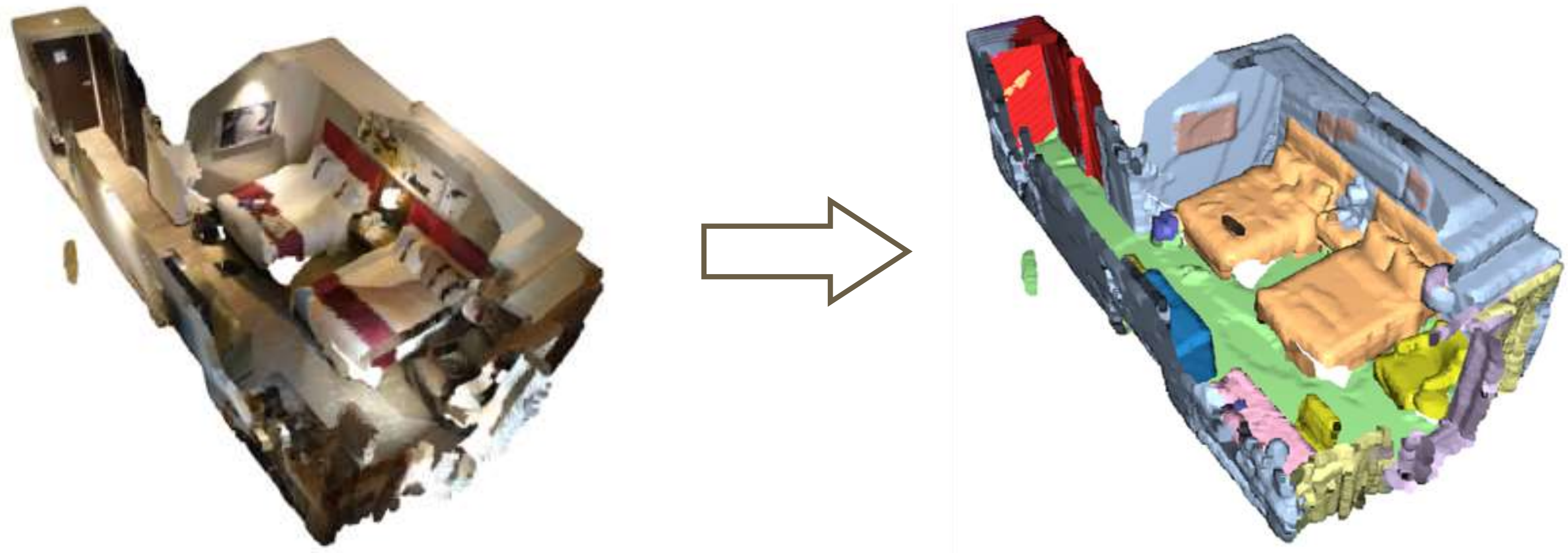
# Sparsity Pattern Reconstruction



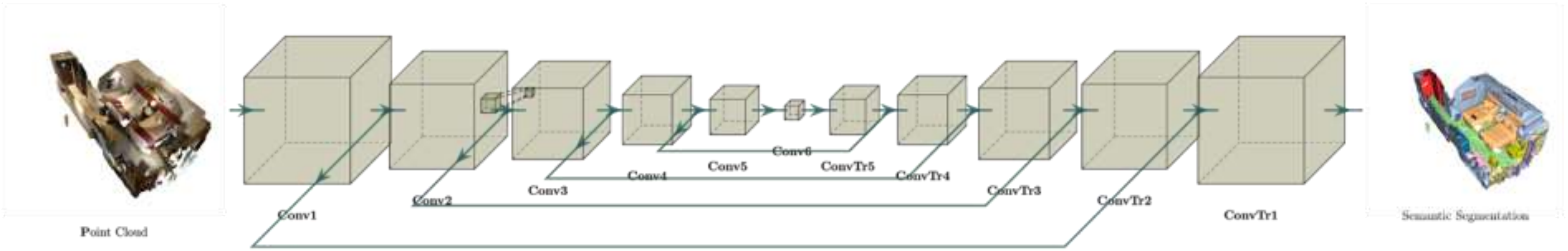

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

# 3D Perception: Semantic Segmentation

- Partition 3D scans or data into semantic parts
- Label each voxel or 3D point as one of semantic labels



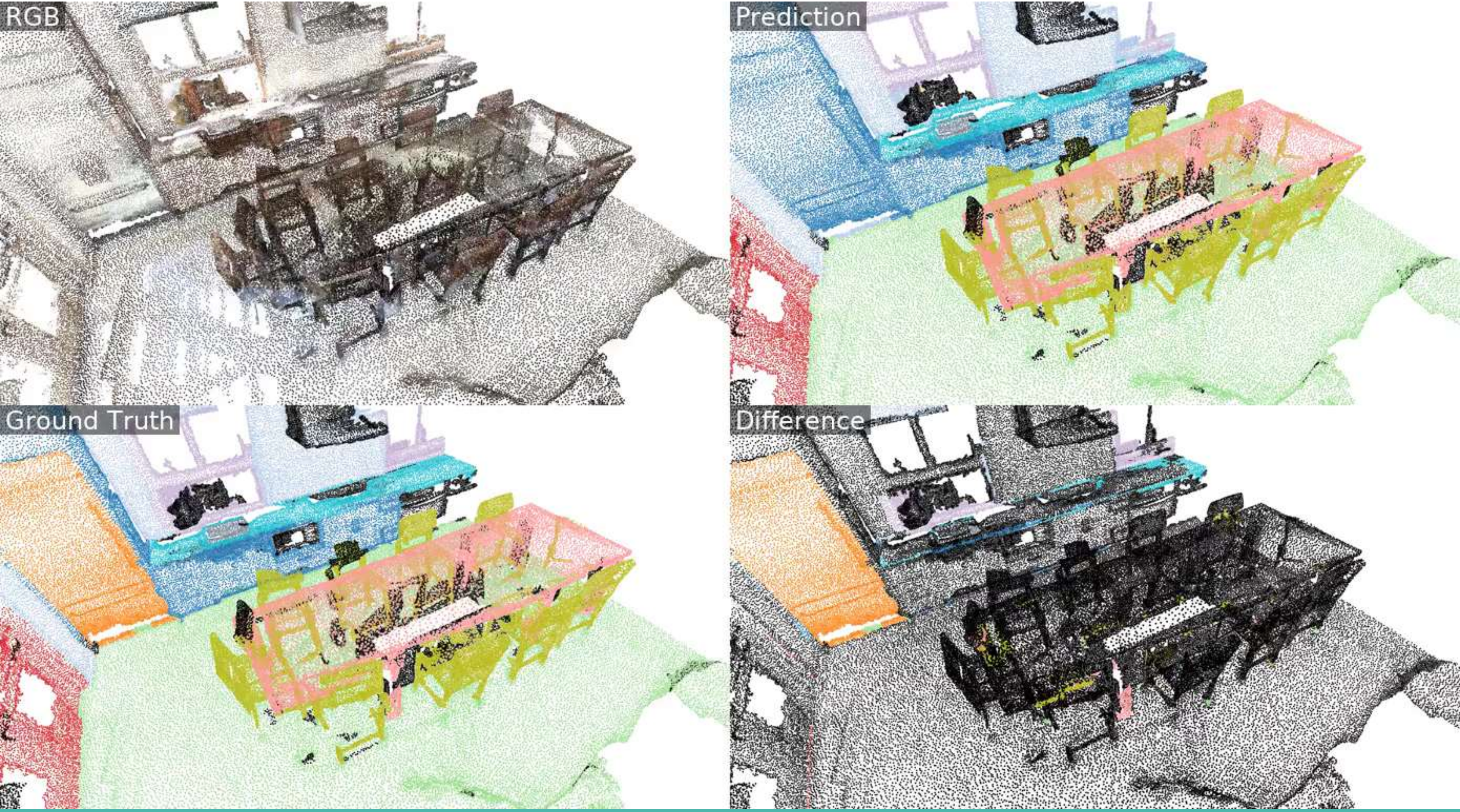
# 3D Semantic Segmentation on Sparse Tensors



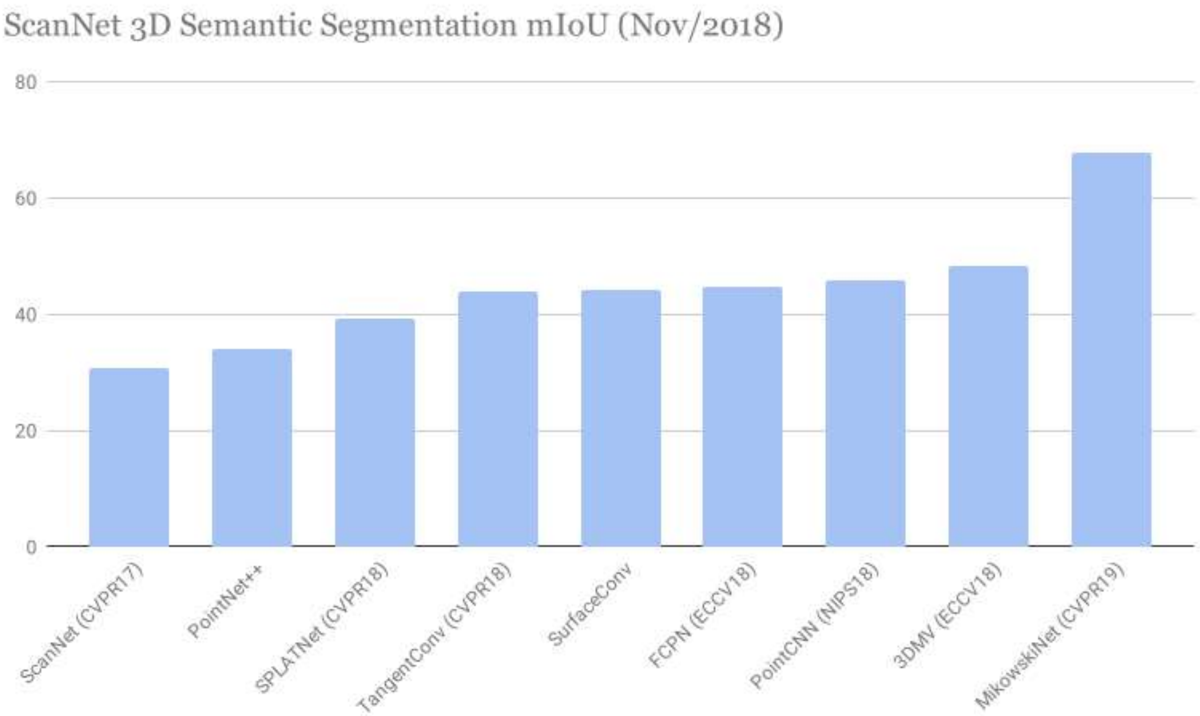

- Sparse tensors for all input/output feature maps
- U-shaped network
  - Hierarchical map
  - Increases receptive field size exponentially

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19


RGB
Prediction
Ground Truth
Difference


RGB
Prediction
Ground Truth
Difference

# Results: ScanNet




Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

# Results: Stanford 3D

| Method | mIOU | mAcc |
|---|---|---|
| PointNet [22] | 41.09 | 48.98 |
| SparseUNet [9] | 41.72 | 64.62 |
| SegCloud [30] | 48.92 | 57.35 |
| TangentConv [29] | 52.8 | 60.7 |
| 3D RNN [32] | 53.4 | 71.3 |
| PointCNN [15] | 57.26 | 63.86 |
| SuperpointGraph [14] | 58.04 | 66.5 |
| MinkowskiNet20 | 62.60 | 69.62 |
| MinkowskiNet32 | **65.35** | **71.71** |

Per class IoU in the supplementary material.

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

## 3D Reconstruction

Supervised Reconstruction

## 3D Perception

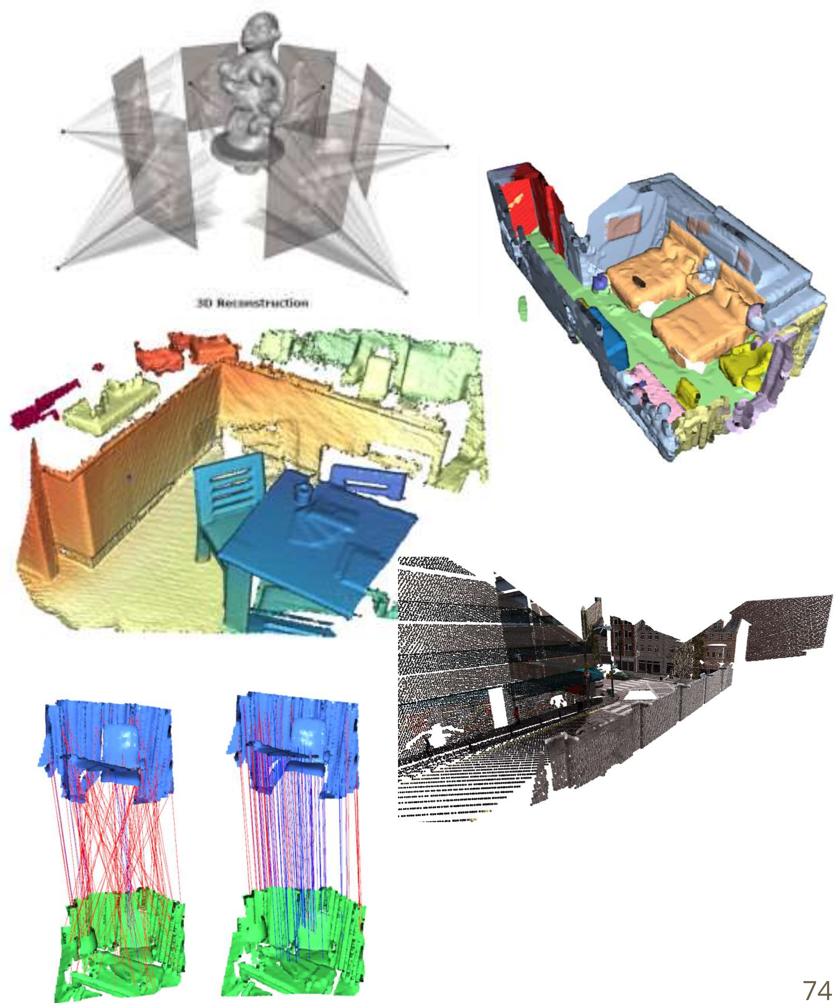
3D Semantic Segmentation

3D Feature Learning

## Perception on a Set of 3D Data

4D Spatio-Temporal Perception

4D and 6D for Registration



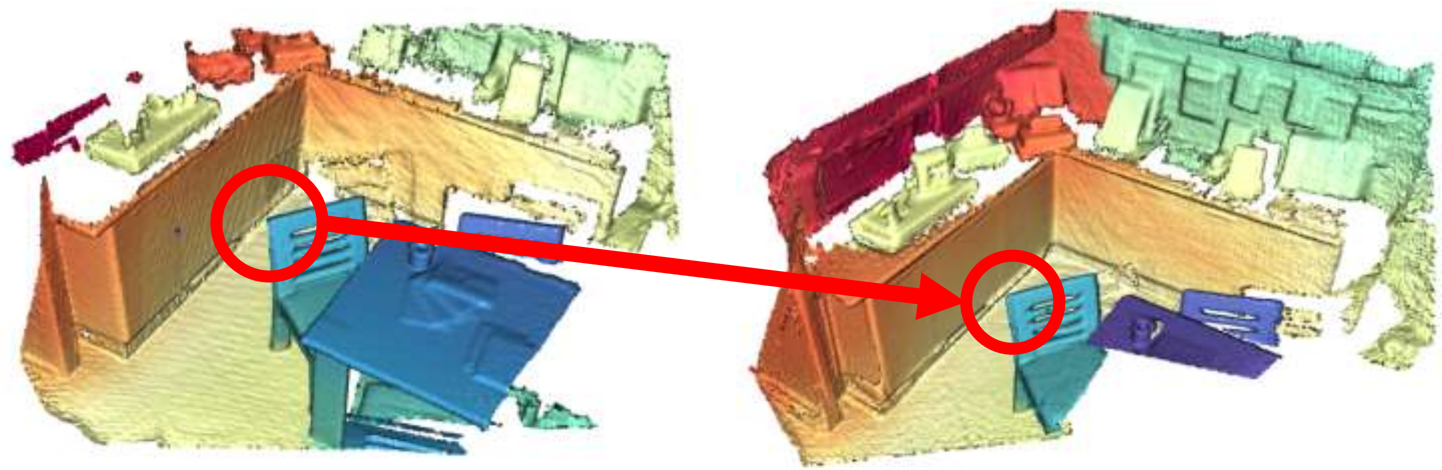
# 3D Feature Learning

- Universal Correspondence Network, **Chris**, JunYoung, Silvio, Manmohan, NIPS’16

- Fully Convolutional Geometric Features, **Chris**, Jaesik, Vladlen, ICCV’19

# 3D Geometric Feature

- A vector representation of the local / global 3D geometry
  - Correspondence, registration, tracking, scene flow, ...



# Prior works in 3D Geometric Features

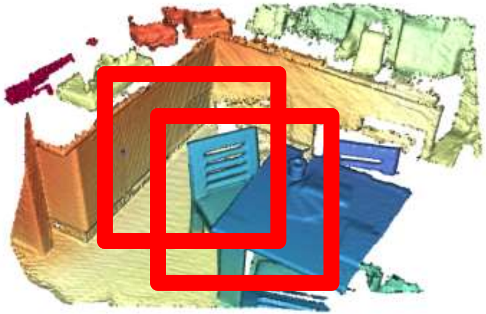
**Hand-designed Features**

Spin Image, USC, SHOT, PFH, FPFH

**Learned Features**

3DMatch, CGF, PointNet, PPF, FoldNet, PPFFold, CapsuleNet, DirectReg, SmoothNet

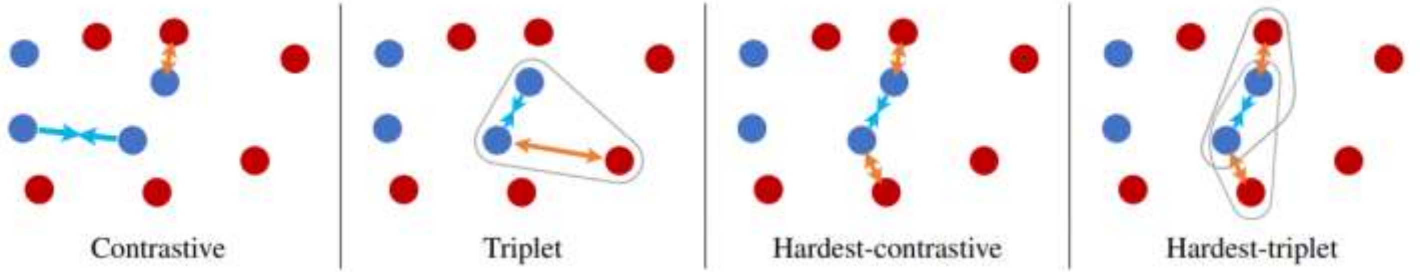
- Extract a small 3D patch
  - Limits context, receptive field
  - Features extracted separately
- Preprocessing
  - Normal, Signed Distance Function, curvatures



Choy et al., **Fully Convolutional Geometric Features**, ICCV'19

# Fully Convolutional Metric Learning

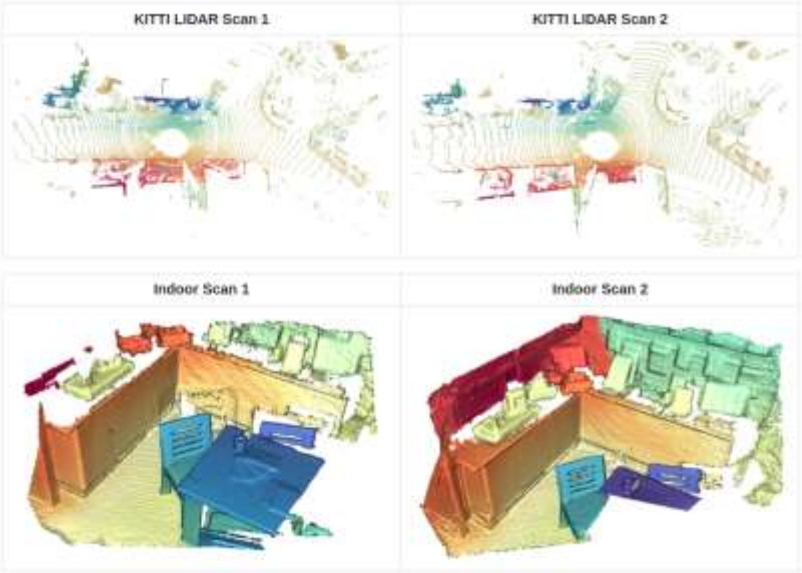
- No preprocessing, no patch extraction
  - no receptive field limit by crop size
  - Efficient reuse of shared computation
- Hardest Negative Mining

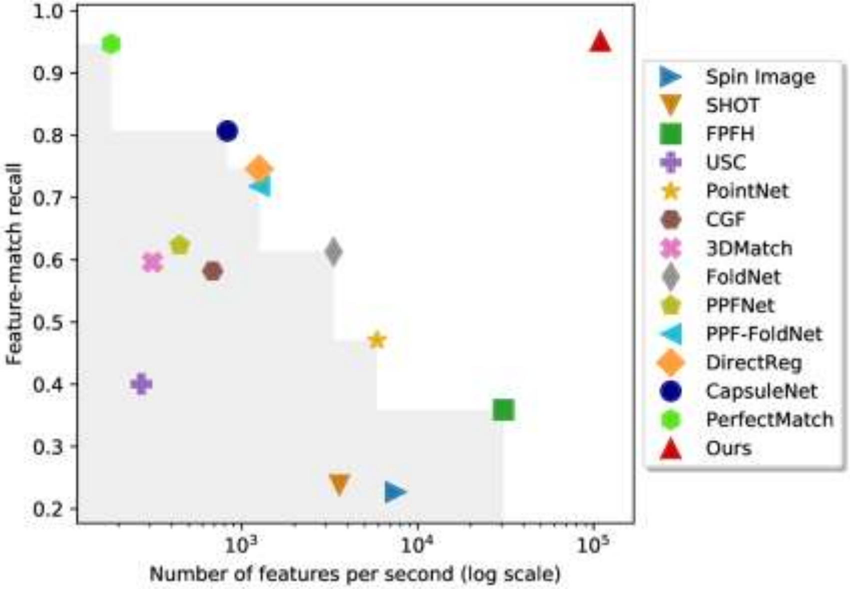




Choy et al., **Universal Correspondence Network**, NIPS'16
Choy et al., **Fully Convolutional Geometric Features**, ICCV'19

# Fully Convolutional Geometric Features




Choy et al., **Fully Convolutional Geometric Features**, ICCV'19

## 3D Reconstruction

Supervised Reconstruction

## 3D Perception

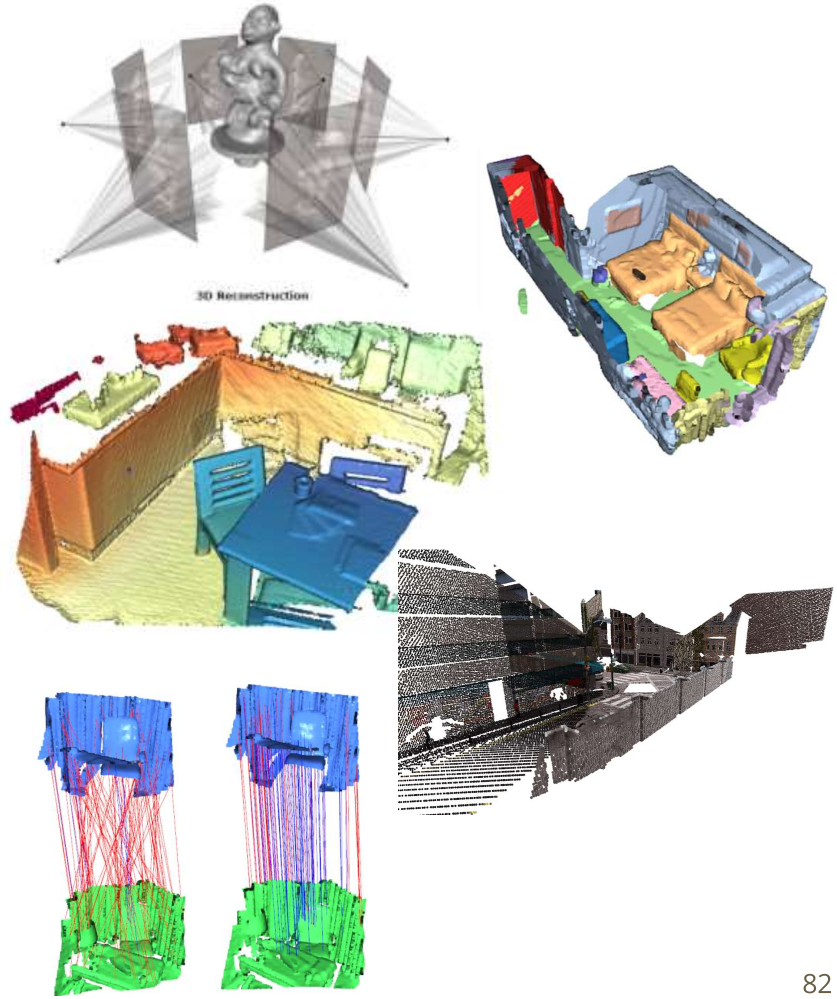
3D Semantic Segmentation

3D Feature Learning

## Perception on a Set of 3D Data

4D Spatio-Temporal Perception

4D and 6D for Registration



# 4D Spatio-temporal data (3D Video)

# 3D to 4D Spatio-temporal perception

**Advantages of 4D data**

- Temporal consistency
- Novel viewpoint
- Dynamics / Action

**Challenges of 4D data**

- Weak 3D perception
- Complexity

  Memory: $O(TN^3)$

  Computation: $O(K^4\ TN^3)$

- 4D Markov Random Fields for Medical Imaging [McInerney & Terzopoulos, 1995]
- 4D Cardiac Image Segmentation [Lorenzo-Valdés et al., 2014]

# High Dimensional Spaces and Generalized Convolution

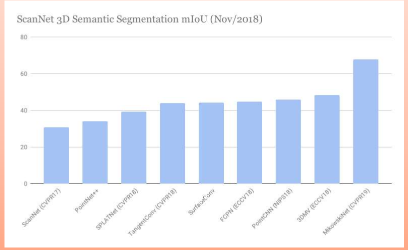
## Challenges

- Weak 3D perception
- Complexity

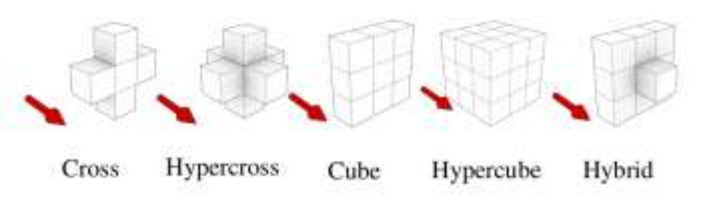
  Memory: $O(TN^3)$

  Computation: $O(K^4 \, TN^3)$

## Minkowski ConvNet



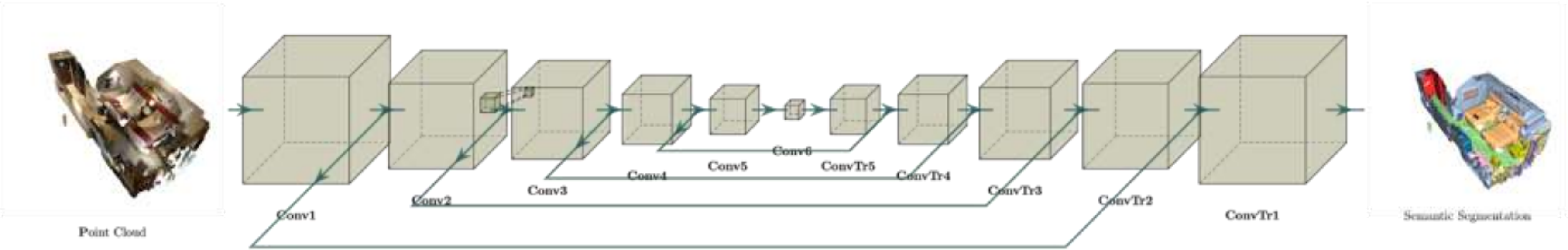

## Sparse Tensor Generalized Convolution



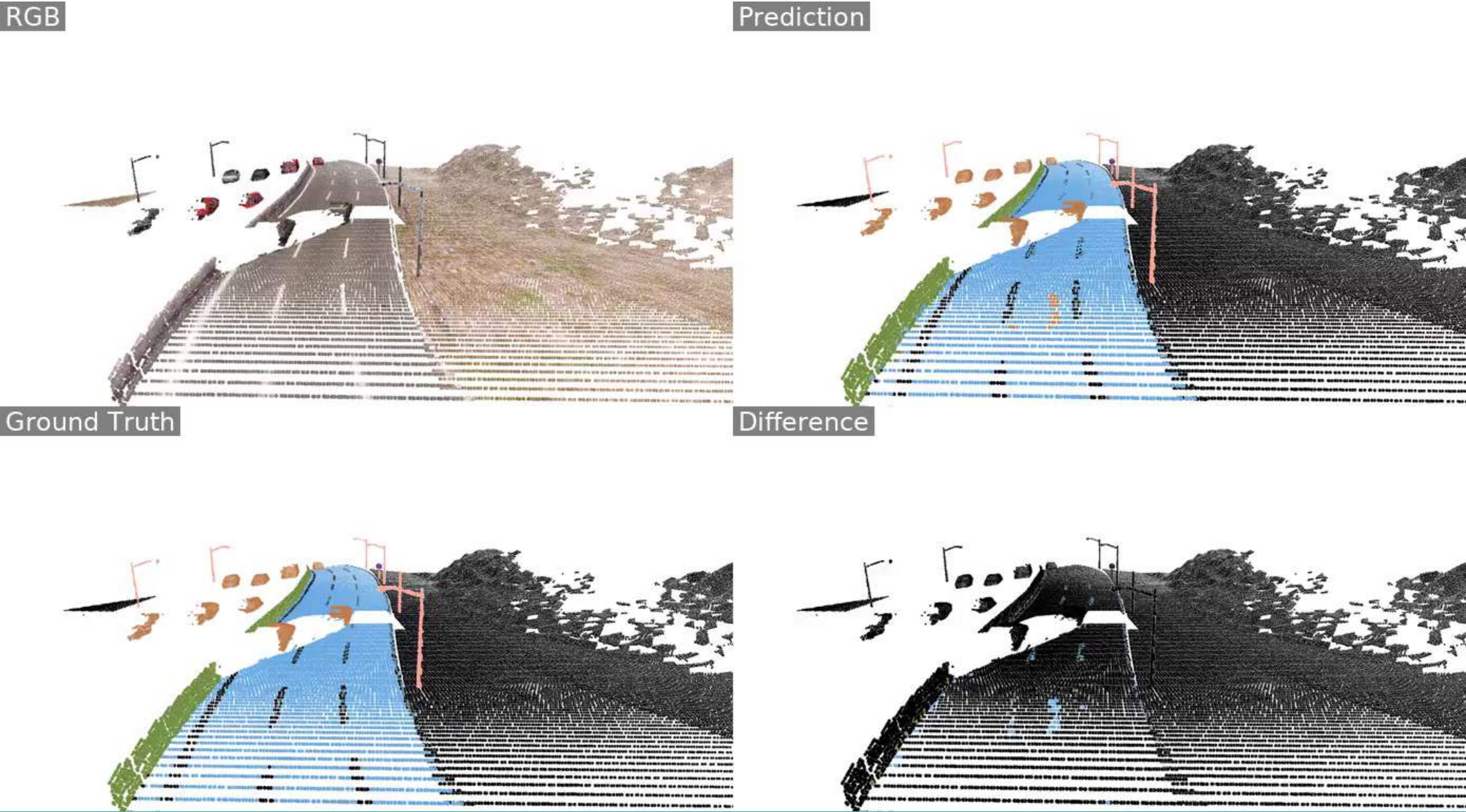

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

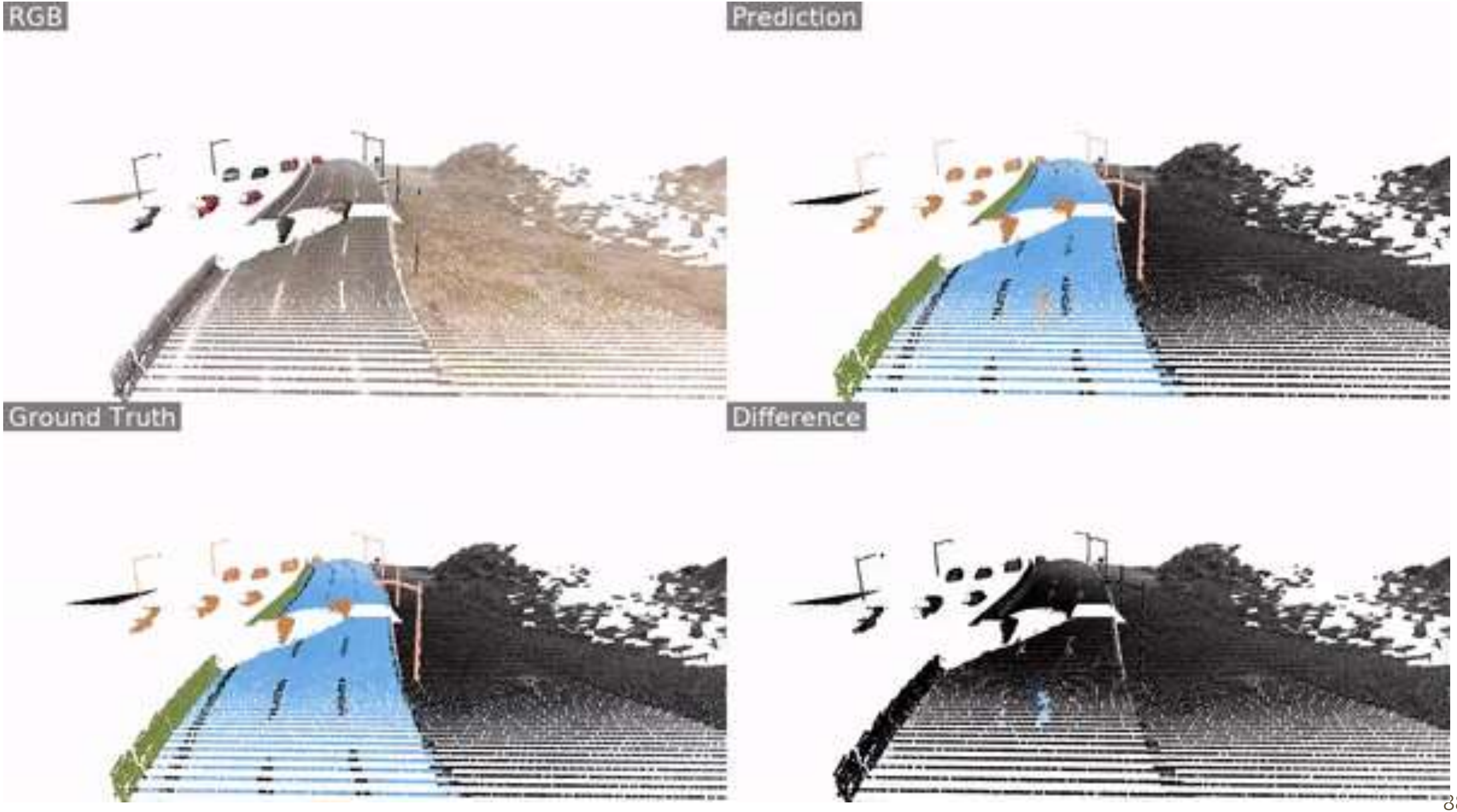
# 4D Spatio-Temporal Semantic Segmentation

- Spatially aligned 3D video
  - Static objects have the same 3D coordinates
  - GPS, SLAM
- Synthetic dataset: Synthia
- Network:
  - U-shaped Net for semantic segmentation, in 4D



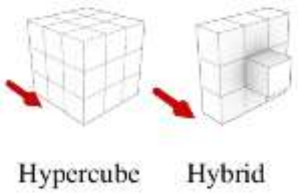

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR’19


RGB
Prediction
Ground Truth
Difference


RGB
Prediction
Ground Truth
Difference

# Results: 4D Synthia Dataset

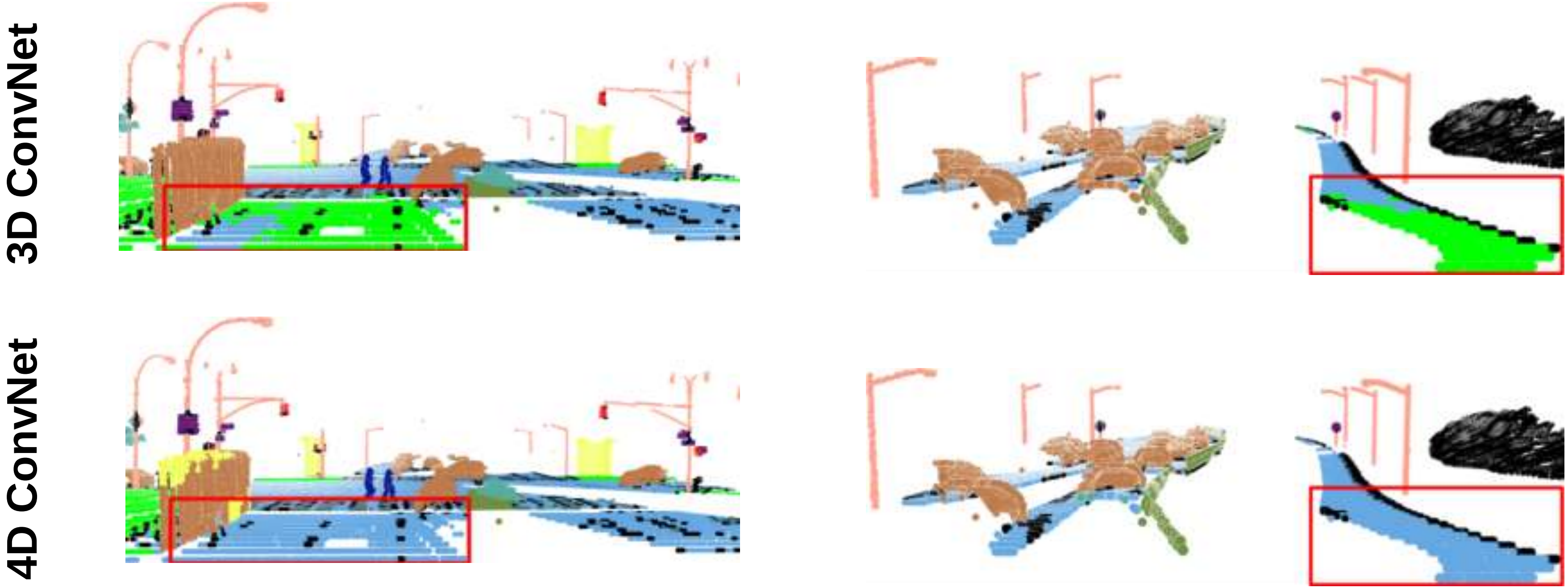
| IoU | Building | Road | Sidewalk | Fence | Vegetation | Pole | Car | Traffic Sign | Pedestrian | Lanemarking | Traffic Light | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3D MinkNet42 | 87.954 | 97.511 | 78.346 | 84.307 | 96.225 | 94.785 | 87.370 | 42.705 | **66.666** | 52.665 | 55.353 | 76.717 |
| 4D Tesseract MinkNet42 | **89.957** | 96.917 | 81.755 | 82.841 | 96.556 | 96.042 | 91.196 | 52.149 | 51.824 | **70.388** | **57.960** | 78.871 |
| 4D MinkNet42 | 88.890 | **97.720** | **85.206** | **84.855** | **97.325** | **96.147** | **92.209** | **61.794** | 61.647 | 55.673 | 56.735 | **79.836** |




Full 4D convolution
More effective for small objects

Faster & Better
Regularized

Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19




Choy et al., **4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks**, CVPR'19

## 3D Reconstruction

Supervised Reconstruction

## 3D Perception

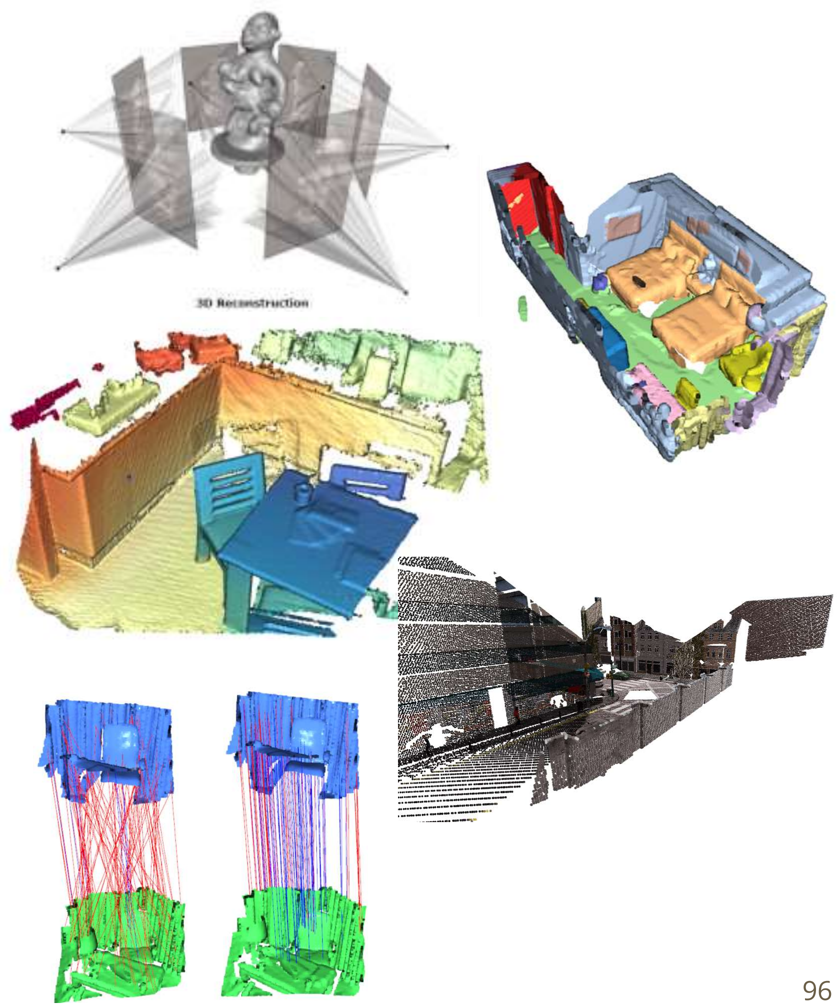
3D Semantic Segmentation

3D Feature Learning

## Perception on a Set of 3D Data

4D Spatio-Temporal Perception

4D and 6D for Registration



# 3D Reconstruction

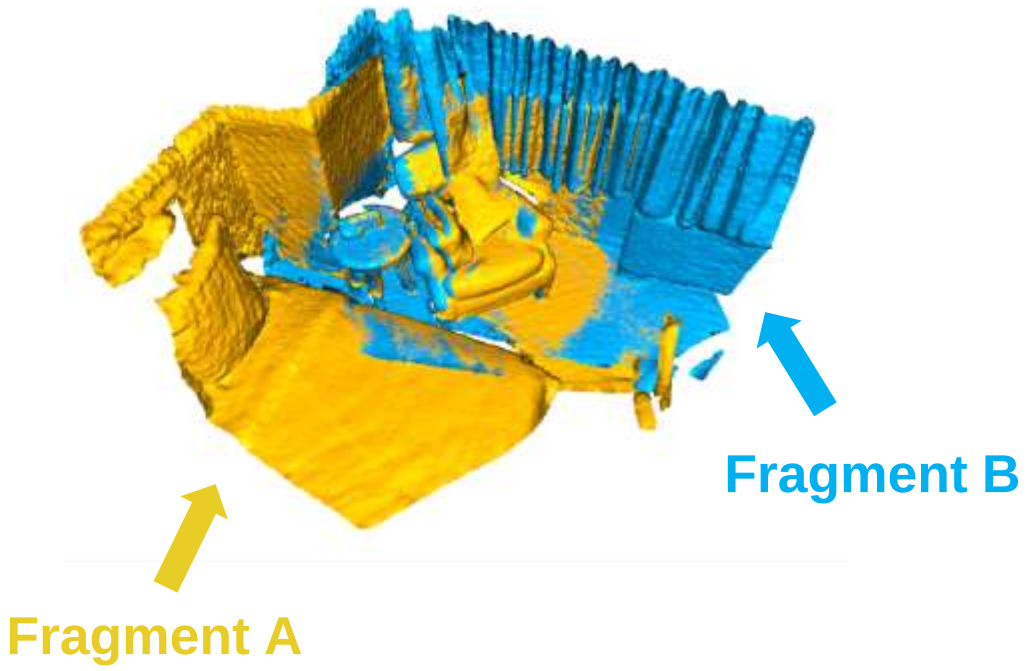
3D Scans

Preprocessing

Fragments

3D Pairwise Registration

Global Consistency



# 3D Pairwise Registration

3D Fragments

Feature Extraction

Correspondence

Global Registration

FCGF, Choy et al. 2019
SmoothNet, Gojcic et al. 2019
CapsuleNet, Zhao et al., 2019
PPF, PPF-Fold, Deng et al., 2019, 2018
FoldingNet, Yang et al., 2017
…

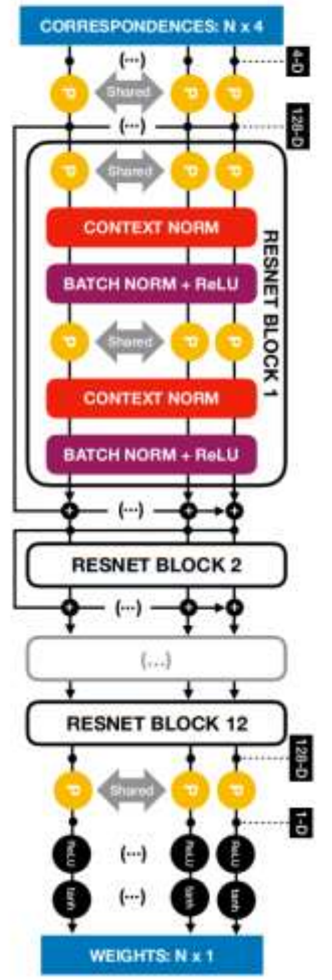
OANet, Zhao et al., 2019
LFGC, Yi et al., 2018

# 3D Pairwise Registration

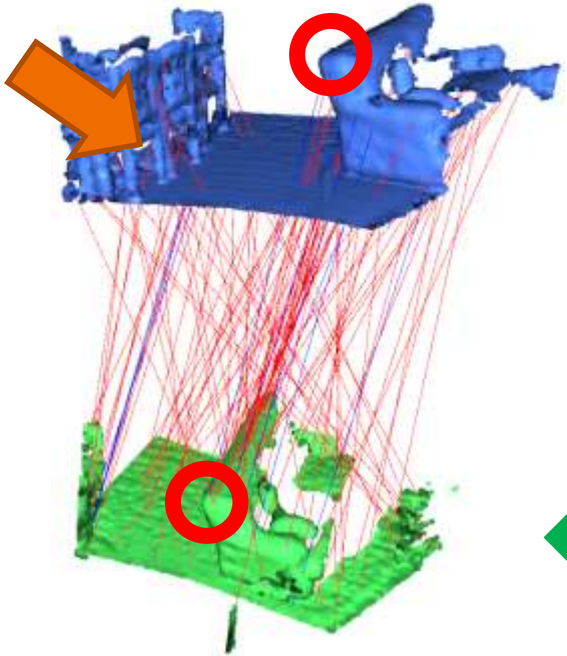
- 3D Fragments
- Feature Extraction
- Correspondence
- Global Registration

OANet, Zhao et al., 2019
LFGC, Yi et al., 2018

- Feature Extraction
- Nearest Neighbor
- ((x,y,z), (x', y', z'))
- Dimensionless data
- Approximate P(correspondence correct)



# Geometry of 3D Correspondence



Inliers: Blue, Outliers: Red

Fragment B

(x,y,z), (x', y', z')

Fragment A

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# 3D Correspondences and 6D Surface

(x,y,z), (x', y', z')

- (x,y,z) → Fragment A
- (x',y',z') → Fragment B

Concatenate

- (x, y, z, x', y', z')
- First 3 follow A, last 3 follow B
- Inliers follow the common geometry

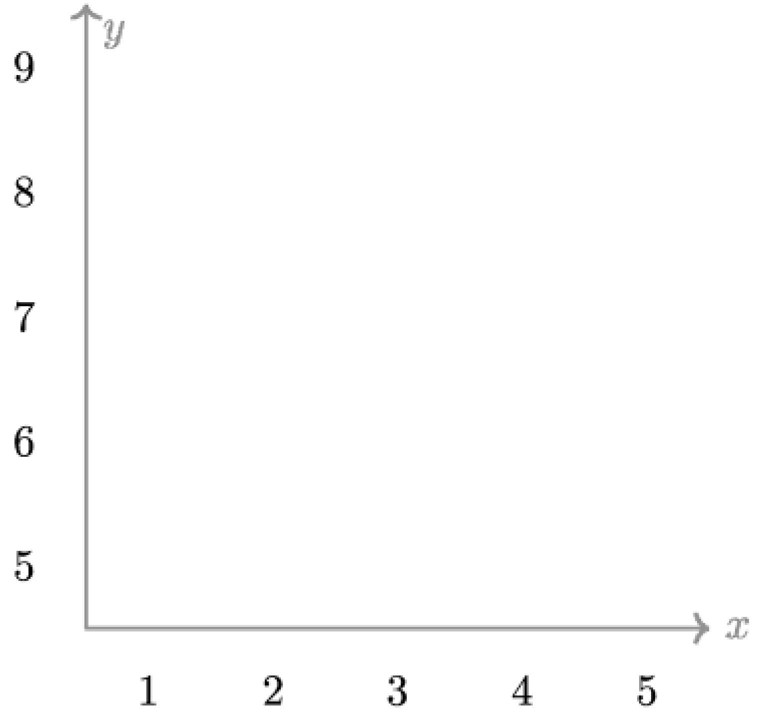
**6D Hyper Surface**

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# Correspondences form high-dimensional geometry

- X = {1,2,3,4,5}
- Y = T(X) where T(x) := x + 4
- Correspondence
  - {(1, 5), (3, 7), (4, 8), (5, 9), (2, 9)}

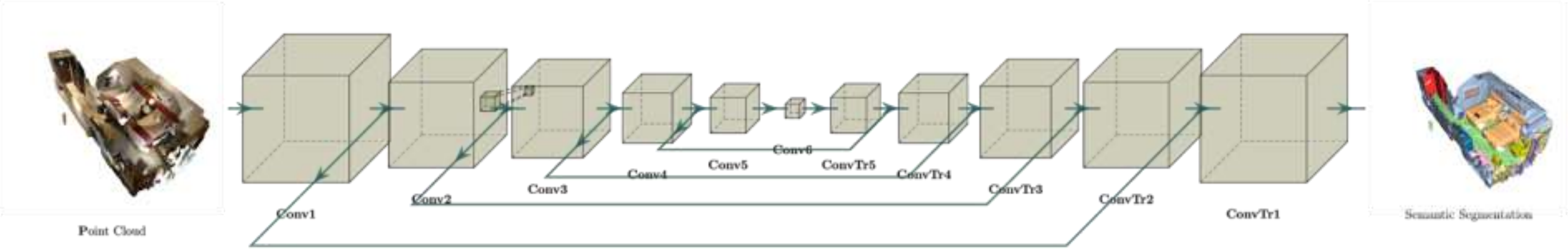
- Correct correspondences
  - Follow the common geometry
  - Inliers
- Incorrect correspondences
  - Outliers




Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

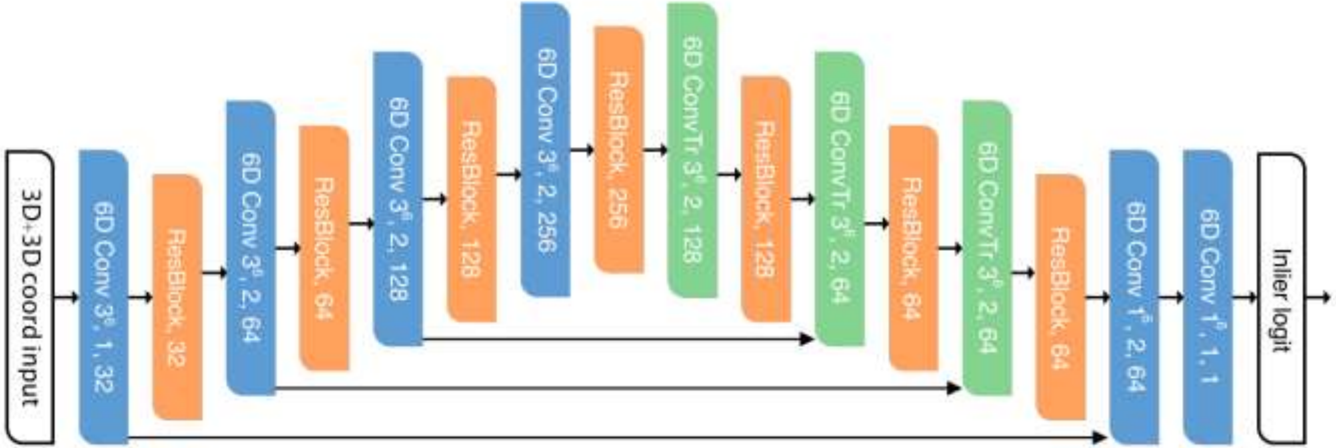
# Inlier vs. Outlier

Label each correspondence as Inlier vs. Outlier
→ Label each 6D point as an Inlier vs. Outlier
→ Label each 3D point as chair, bed, …




Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

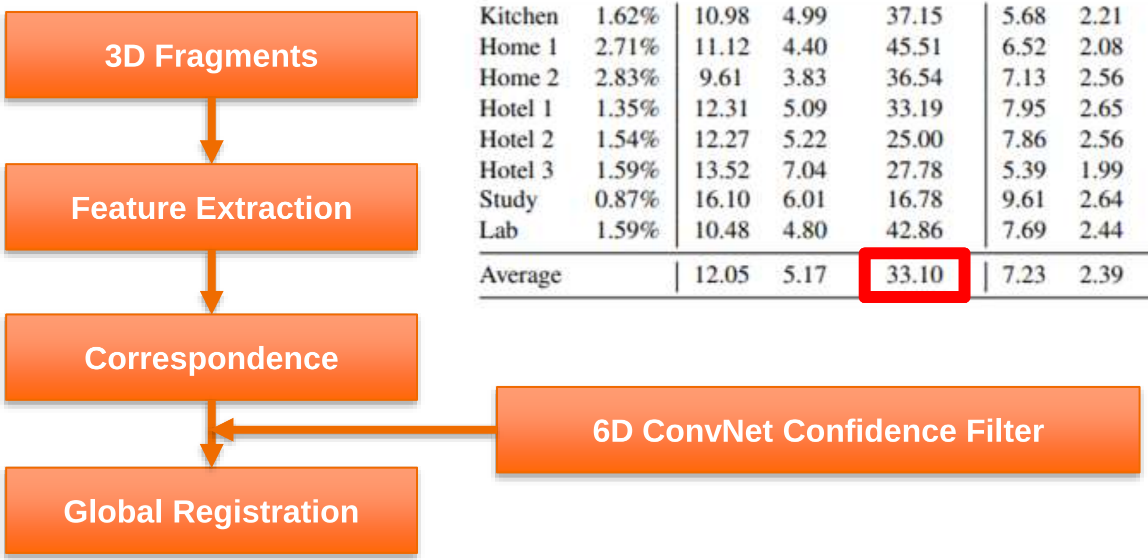
# 6D Convolutional Neural Network




Translation invariance: Fragments can be located anywhere in 3D space
Multi-resolution (large receptive field, less sparse)

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# Results: 3D Correspondence Segmentation



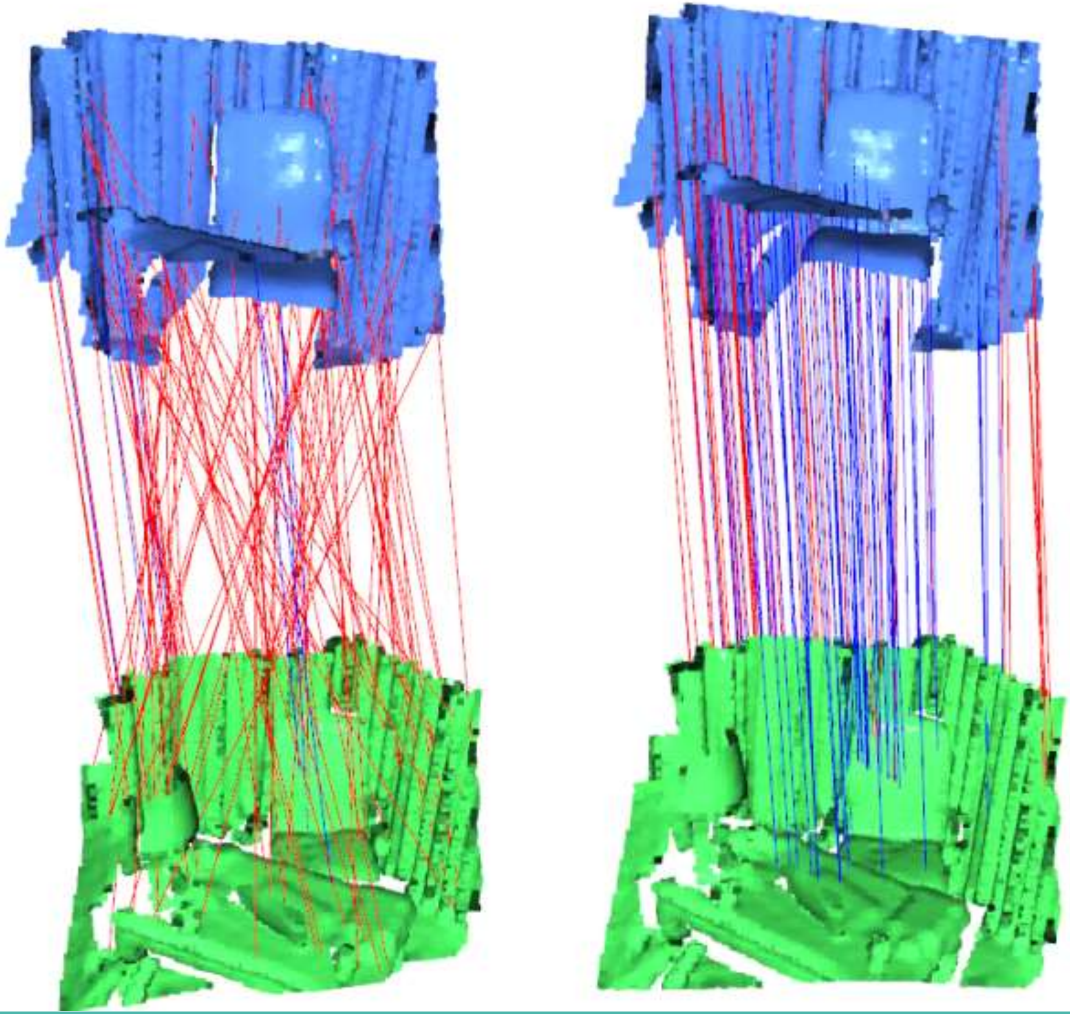

| | SNR | FPFH + FGR TE | FPFH + FGR RE | FPFH + FGR Succ. Rate | FPFH + Ours + FGR TE | FPFH + Ours + FGR RE | FPFH + Ours + FGR Succ. Rate | FPFH + RANSAC TE | FPFH + RANSAC RE | FPFH + RANSAC Succ. Rate | FPFH + Ours + RANSAC TE | FPFH + Ours + RANSAC RE | FPFH + Ours + RANSAC Succ. Rate |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Kitchen | 1.62% | 10.98 | 4.99 | 37.15 | 5.68 | 2.21 | 65.61 | 6.25 | 2.17 | 44.47 | 5.90 | 1.98 | 69.57 |
| Home 1 | 2.71% | 11.12 | 4.40 | 45.51 | 6.52 | 2.08 | 80.77 | 7.07 | 2.19 | 61.54 | 6.00 | 1.87 | 80.13 |
| Home 2 | 2.83% | 9.61 | 3.83 | 36.54 | 7.13 | 2.56 | 64.42 | 6.47 | 2.40 | 50.00 | 7.86 | 2.56 | 69.71 |
| Hotel 1 | 1.35% | 12.31 | 5.09 | 33.19 | 7.95 | 2.65 | 76.11 | 7.48 | 2.75 | 48.67 | 7.38 | 2.38 | 80.09 |
| Hotel 2 | 1.54% | 12.27 | 5.22 | 25.00 | 7.86 | 2.56 | 69.23 | 9.54 | 3.18 | 47.12 | 6.40 | 2.25 | 70.19 |
| Hotel 3 | 1.59% | 13.52 | 7.04 | 27.78 | 5.39 | 1.99 | 72.22 | 5.91 | 2.46 | 59.26 | 5.85 | 2.36 | 81.48 |
| Study | 0.87% | 16.10 | 6.01 | 16.78 | 9.61 | 2.64 | 53.42 | 10.05 | 3.01 | 30.48 | 8.51 | 2.23 | 56.16 |
| Lab | 1.59% | 10.48 | 4.80 | 42.86 | 7.69 | 2.44 | 61.04 | 8.01 | 2.31 | 45.45 | 6.64 | 2.12 | 68.83 |
| Average | | 12.05 | 5.17 | 33.10 | 7.23 | 2.39 | 67.85 | 7.60 | 2.56 | 48.37 | 6.82 | 2.22 | 72.02 |

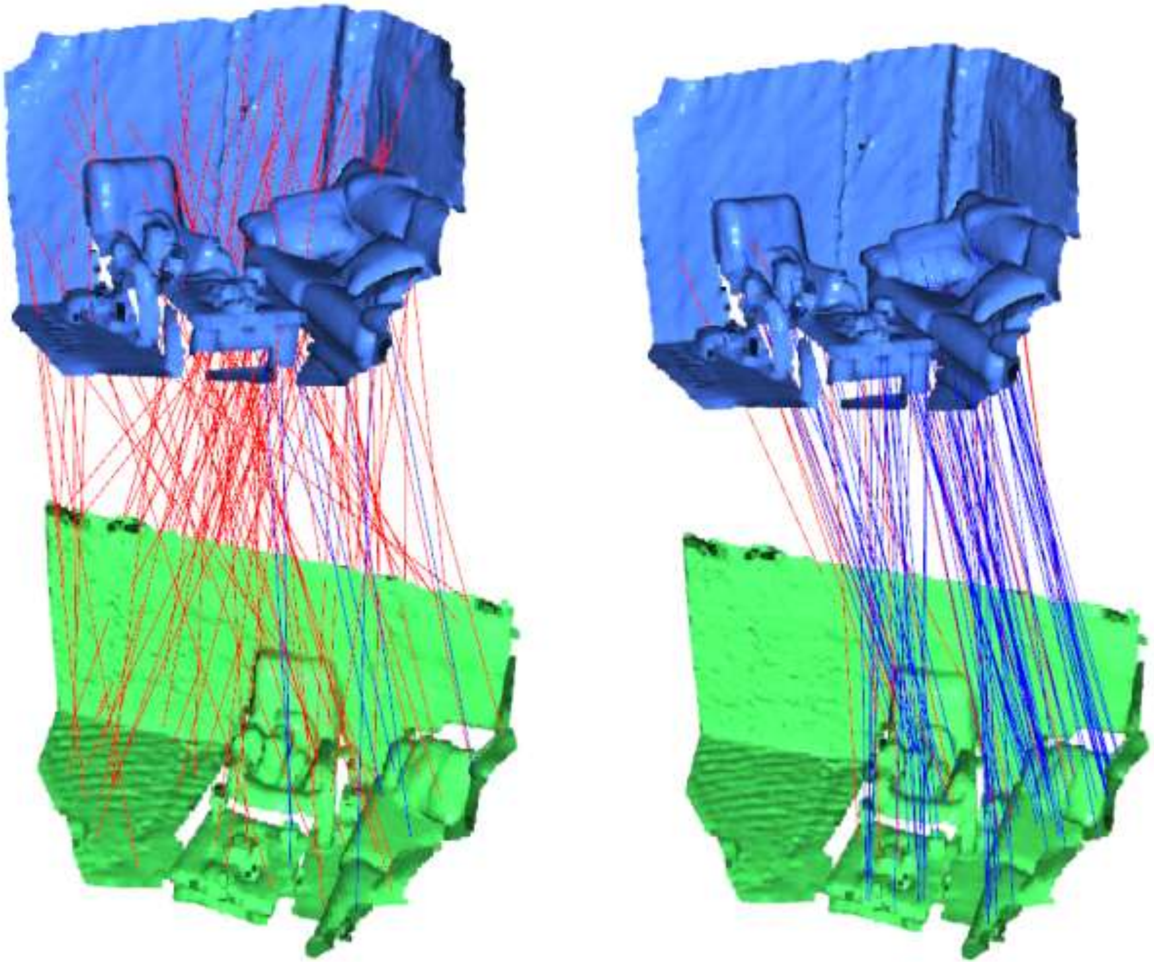
Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# Results: 3D Correspondence Segmentation

| | SNR | FPFH + FGR | | | FPFH + Yi *et al.* [2] + FGR | | | FPFH + Yi *et al.* [2] + RANSAC | | | FPFH + Ours + RANSAC | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | TE | RE | Succ. Rate | TE | RE | Succ. Rate | TE | RE | Succ. Rate | TE | RE | Succ. Rate |
| Kitchen | 4.90% | 9.32 | 3.92 | 44.86 | 8.06 | 3.36 | 55.53 | 9.10 | 3.65 | 57.71 | 5.90 | 1.98 | 69.57 |
| Home 1 | 7.50% | 9.13 | 3.53 | 51.92 | 8.76 | 3.23 | 64.10 | 9.28 | 2.99 | 67.31 | 6.00 | 1.87 | 80.13 |
| Home 2 | 6.65% | 9.02 | 3.58 | 36.54 | 7.96 | 3.13 | 45.19 | 10.02 | 3.71 | 53.85 | 7.86 | 2.56 | 69.71 |
| Hotel 1 | 5.22% | 10.20 | 3.86 | 46.02 | 9.14 | 3.46 | 57.52 | 11.25 | 3.80 | 61.95 | 7.38 | 2.38 | 80.09 |
| Hotel 2 | 4.75% | 10.69 | 4.82 | 35.58 | 9.74 | 3.82 | 50.00 | 11.06 | 4.52 | 56.73 | 6.40 | 2.25 | 70.19 |
| Hotel 3 | 5.20% | 13.10 | 4.69 | 46.30 | 10.36 | 3.86 | 57.41 | 10.59 | 4.05 | 68.52 | 5.85 | 2.36 | 81.48 |
| Study | 3.83% | 14.20 | 4.74 | 27.40 | 12.95 | 4.01 | 37.67 | 12.88 | 4.09 | 48.63 | 8.51 | 2.23 | 56.16 |
| Lab | 4.98% | 9.33 | 3.60 | 46.75 | 7.51 | 3.26 | 49.35 | 8.85 | 2.94 | 50.65 | 6.64 | 2.12 | 68.83 |
| Average | | 10.62 | 4.09 | 41.92 | 9.31 | 3.52 | 52.10 | 10.38 | 3.72 | 58.17 | 6.82 | 2.22 | 72.02 |

Yi et al., **Learning to find good correspondences**, 2018
Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020





# 3D Correspondences and 6D Geometry

3D Fragments

↓

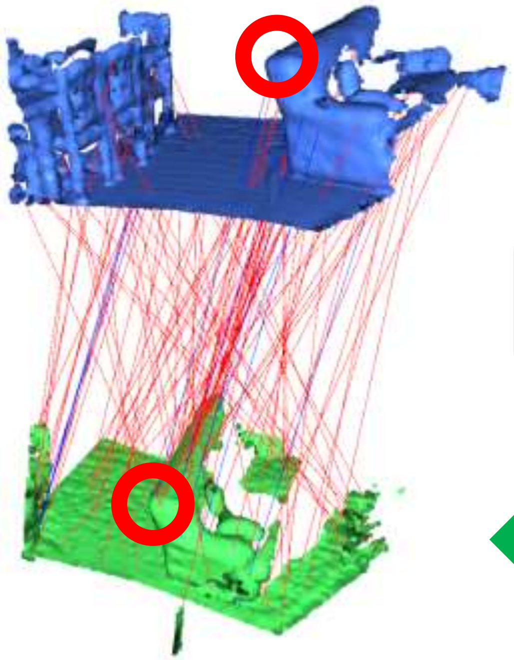
Feature Extraction

↓

Correspondence

↓

Global Registration



Fragment B

(x,y,z), (x’, y’, z’)

Fragment A

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# 2D Correspondences and 4D Geometry

Images

Feature Extraction

Correspondence

Global Registration

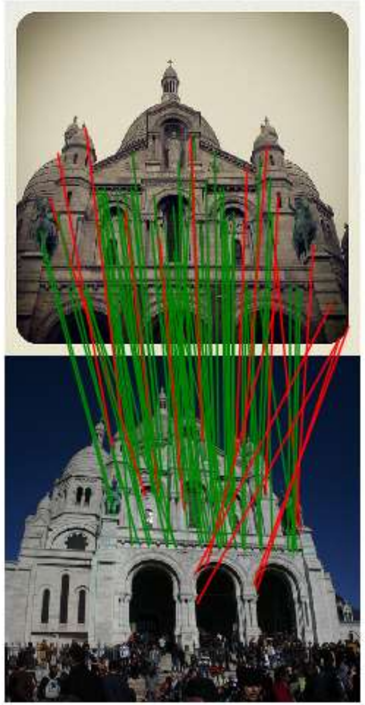


Image B

(x,y), (x', y')

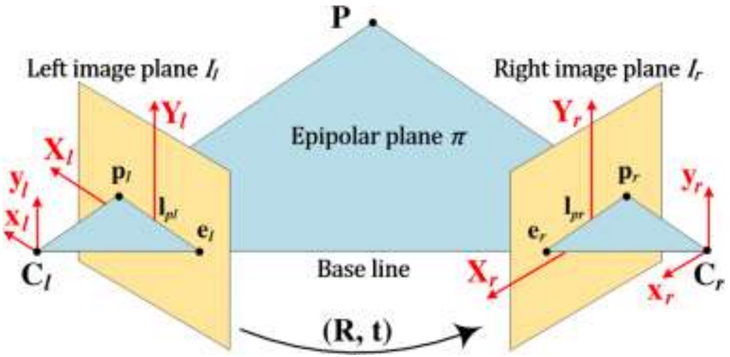
Image A

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# 2D Correspondences and 4D Geometry

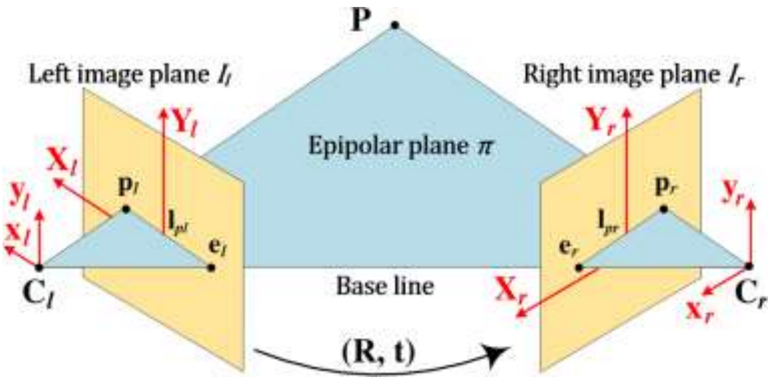
(x,y), (x', y')

- (x,y) → Image A
- (x',y') → Image B




Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# 2D Correspondences and 4D Geometry




$$\mathbf{E} = \mathbf{R}\,[\mathbf{t}]_\times$$

$$\mathbf{x}^T\mathbf{E}\mathbf{x}' = 0$$

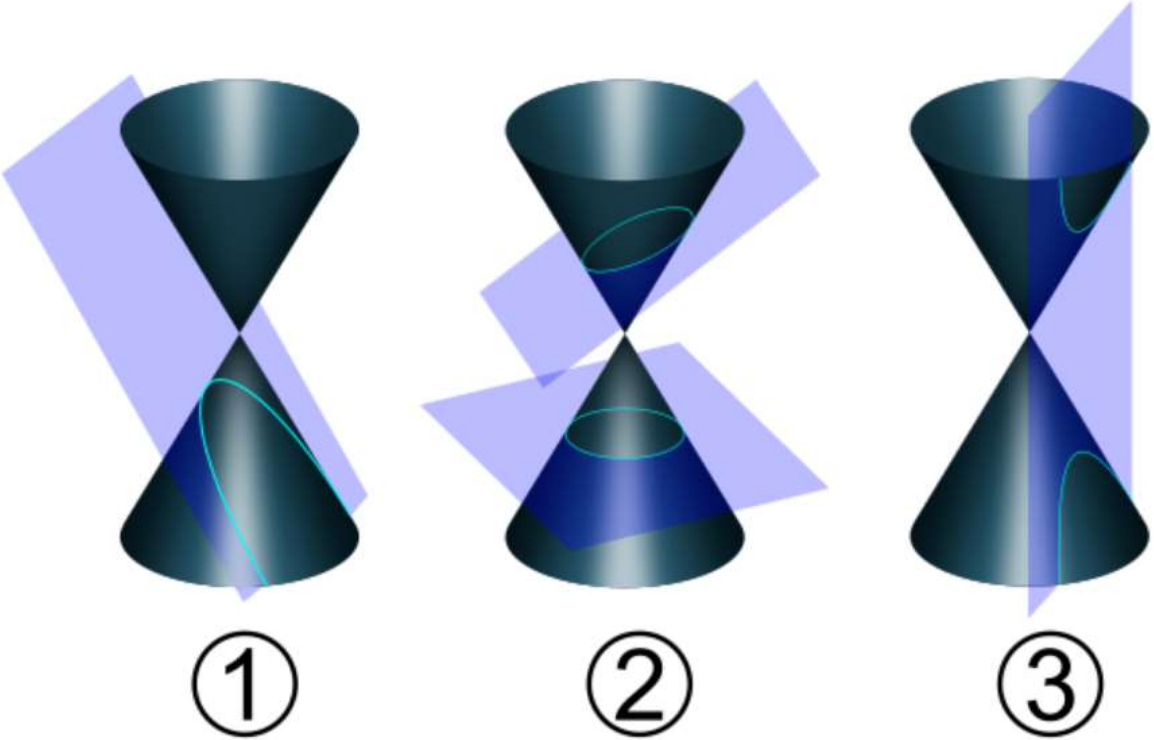
**Second degree polynomial (x, y, x’, y’) = 0**

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# Conic Sections

| conic section | equation | eccentricity ($e$) | linear eccentricity ($c$) | semi-latus rectum ($\ell$) | focal parameter ($p$) |
|---|---|---|---|---|---|
| circle | $x^2 + y^2 = a^2$ | $0$ | $0$ | $a$ | $\infty$ |
| ellipse | $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ | $\sqrt{1 - \frac{b^2}{a^2}}$ | $\sqrt{a^2 - b^2}$ | $\frac{b^2}{a}$ | $\frac{b^2}{\sqrt{a^2 - b^2}}$ |
| parabola | $y^2 = 4ax$ | $1$ | N/A | $2a$ | $2a$ |
| hyperbola | $\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$ | $\sqrt{1 + \frac{b^2}{a^2}}$ | $\sqrt{a^2 + b^2}$ | $\frac{b^2}{a}$ | $\frac{b^2}{\sqrt{a^2 + b^2}}$ |

# 4D Hyper Conic Section of 5D Hyper Cones



# 2D Correspondences and 4D Geometry

(x,y), (x', y')

- (x,y) → Image A
- (x',y') → Image B

$$\mathbf{x}^T \mathbf{E} \mathbf{x}' = 0$$

- 2-nd degree polynomial = 0

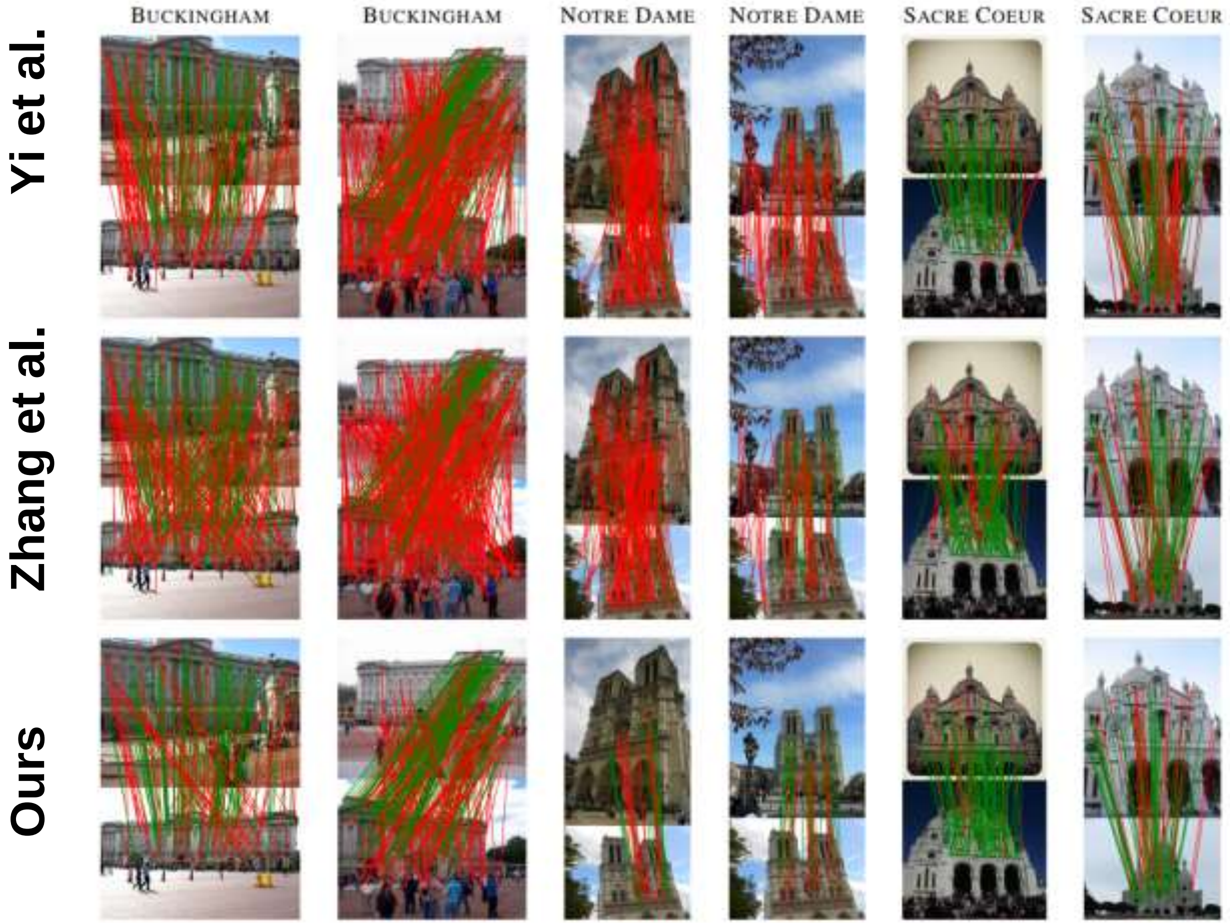
**4D hyper conic section**

Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# YFCC 100M dataset

| | LMeDs [34] | | | MLESAC [40] | | | Yi *et al.* [44] | | | Zhang *et al.* [47] | | | Ours | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Prec. | Recall | F1 | Prec. | Recall | F1 | Prec. | Recall | F1 | Prec. | Recall | F1 | Prec. | Recall | F1 |
| BUCKINGHAM | 0.213 | 0.178 | 0.194 | 0.294 | 0.299 | 0.297 | 0.490 | 0.767 | 0.598 | 0.535 | 0.804 | 0.642 | 0.589 | **0.830** | **0.689** |
| NOTRE DAME | 0.335 | 0.197 | 0.248 | 0.489 | 0.422 | 0.453 | 0.568 | 0.890 | 0.693 | 0.679 | 0.901 | 0.774 | **0.701** | **0.922** | **0.796** |
| REICHTAG | 0.380 | 0.217 | 0.276 | 0.573 | 0.441 | 0.498 | 0.736 | 0.876 | 0.800 | **0.808** | 0.878 | **0.842** | 0.772 | **0.903** | 0.832 |
| SACRE COEUR | 0.203 | 0.104 | 0.137 | 0.418 | 0.292 | 0.344 | 0.653 | 0.865 | 0.744 | 0.724 | 0.902 | 0.803 | **0.726** | **0.921** | **0.812** |
| Average | 0.283 | 0.174 | 0.214 | 0.444 | 0.364 | 0.398 | 0.612 | 0.849 | 0.709 | 0.686 | 0.871 | 0.765 | **0.697** | **0.894** | **0.782** |

Yi et al., **Learning to find good correspondences**, 2018
Zhang et al., **Learning Two-View Correspondences and Geometry Using Order-Aware Network**, 2019
Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020




Yi et al., **Learning to find good correspondences**, 2018
Zhang et al., **Learning Two-View Correspondences and Geometry Using Order-Aware Network**, 2019
Choy et al., **High-dimensional Convolutional Networks for Geometric Pattern Recognition**, 2020

# Conclusions

3D Convolutional Networks
4D Convolutional Networks

4D Convolutional Networks
6D Convolutional Networks

7D Convolutional Networks

32D Convolutional Networks

**3D Reconstruction**

Supervised Reconstruction

**3D Perception**

3D Semantic Segmentation

3D Feature Learning

**Perception on a Set of 3D Data**

4D Spatio-Temporal Perception

4D and 6D for Registration

# Conclusions and Future Work

- Many more high-dimensional problems
  - Geometric structure
- Expand the high-dimensional pattern recognition problems to
  - 3D object detection
  - Tracking
  - Reconstruction

# Thank you

# Thank you

Vladlen Koltun

Jaesik Park

Manmohan Chandraker

JunYoung Gwak

Iro Armeni

Lyne Tchapmi

Kevin Chen

Kuan Fang

# Thank you

Benjamin Van Roy

Leonidas Guibas

Gordon Wetzstein

Tsachy Weissman

# Thank you

Danfei Xu, Yuke Zhu, Animesh Garg, Andrey Kurenkov, Manolis Savva, Angel Chang, Namhoon Lee, Yu Xiang, Junha Lee, Michael Stark

# Thank you for your attention