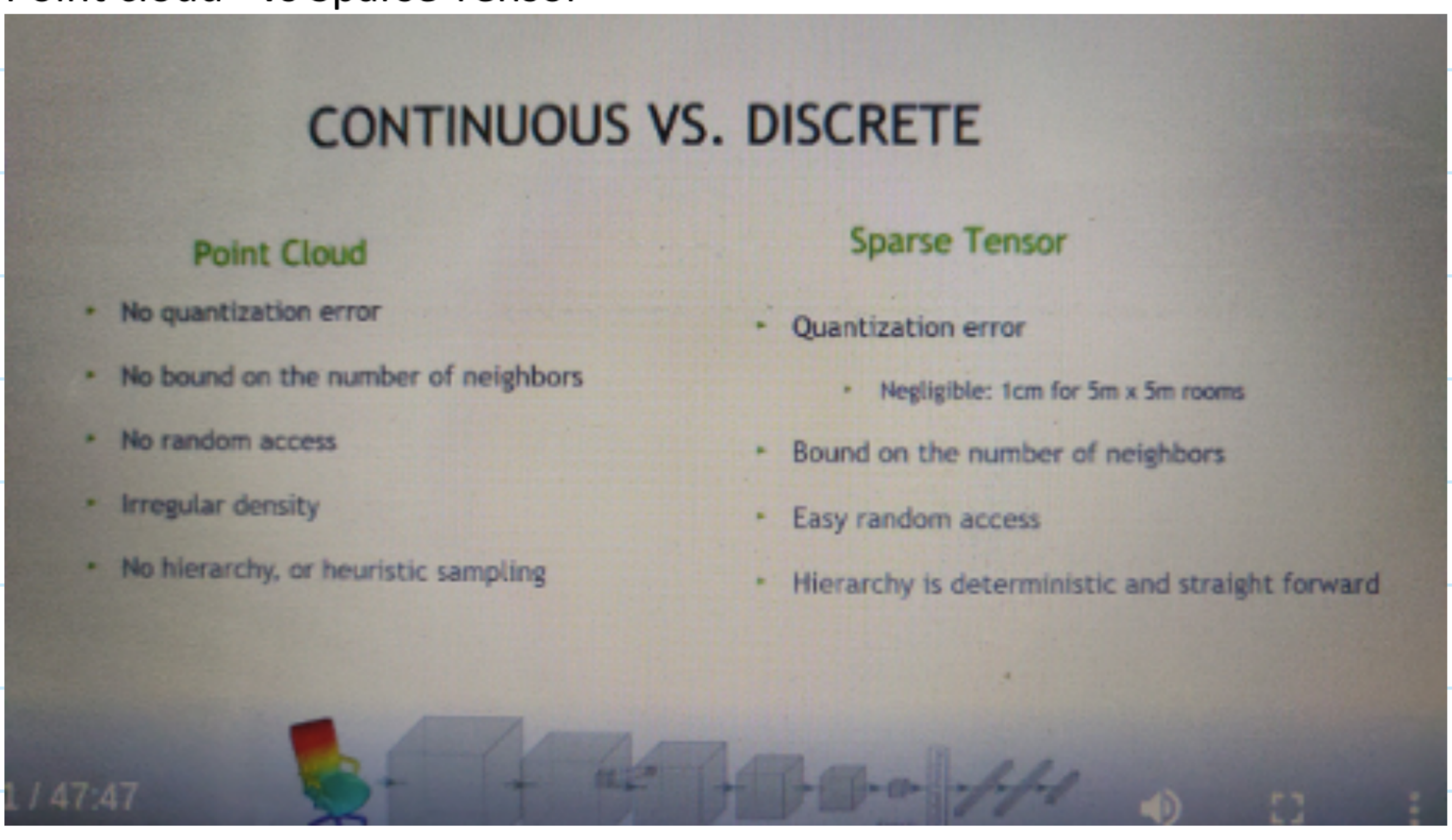
# Deep Representation and Estimation of State for Robotics - 1
# 3D perception with sparse tensors- Nvidia Research

27 October 2020 10:58

Sparse matrix: save only non-zero elements; sparse tensor : N dimensional extension

Point cloud –vs Sparse Tensor




**Sparse Tensor and Convolution**
Convolution for 3D: Conv is most general invariant operation,
CNN- Best inductive bias and regularization, hierarchical representation

1. Dense Convolution Convnet : Dense grid, too large and inefficient, directional weights
2. MLP or PointNet: PointNet, Fast efficient, no directional weights
3. Graph "Conv"Net: Mesh, preserves geometry by edges, no directional weights
4. Cont. ConvNet, Point cloud, Expensive neighbour search, it could be slow, conts func
5. Sparse conv: sparse Tensor, discretized directional weight , best combination of efficiency
   Sparse tensor- *MinkowskiEngine (library for sparse tensor)*

**3D/4D Semantic Segmentation**
Partition 3d scans or data into sematic parts, label them
*4D Spatio-Temporal ConvNets: Minkowski CN(CVPR'19)*
Sparse tensor for input output feature
Good jump on scannet 3D semantic segmentation benchmark

3D to 4D Spatio-temporal perception:
4D DATA: temporal consistency, novel viewpoint, dynamics/action
But challenges like weak 3d perception, complexity is higher in memory and computation,
Complexity has been reduced with generailized convolution used with incorporating sparse tensor kernel
Spatially aligned 3D video, synthetic dataset:synthia. Network: 4D U-Shaped Net for semantic segmentation,sparse tensor kernel

3D Geometric Features:
Early hand-designed features, now learned Features
Extract a small 3d patch->features extracted separately

Before learning, processing done

*Fully convolutional Metric Learning*
*Sparse Fully convolutional Metric Learning*

| | |
|---|---|
| | Dense Image -> spatially Sparse Tensor |

*Fully convolutional hardest contrastive loss*
Geometric correspondences

Evaluation done on 3D Match benchmarks: their method achieves improvement than other learning based methods

**3D Registration**
Pipelines when no camera extrinsics are given . Feature matching -> outlier filtering -> transformation Estimation -> Fine Tuning
Feature Matching: nearest neighbour
Outlier filtering- 6D conv network (to segment the inliers)
Inliers lie in 3d subspace in 6d, outlier lie as noise
Foreground segmentation,:foreground-background segmentation
Transformation Estimation: procrustes analysis :
Fine-tuning : gradient based optimization, conts 6D representation

Better alignment on small objects than ransac and fast global registration

**3D Detection**
Single stage (single-shot)vs 2 stage

Many stag: input image -> object/region proposals/deep learning region classifier-> region classification, box registration
One Stage : tend to be faster and efficient but at the cost of detection matrix

1. 3D semantic instance segmentation cvpr 19
one stage 3 detection on a dense grid
2.low resolution ConvNet 3 dense grid
one stage 3 detection on a dense grid
3.Votenet

4. Non-conv-net on surface

Now : nvidia"s single shot object detection :
High resolution ConvNet on sparse Grid of Surface
***Generative Networks : generating geometry/sparsity Pattern***
Object detection and generating bounding box "Anchors"
Deep U shaped network created
Ultimately high-resolution convnet
Runtime: faster than votenet
However problem in detecting white boards, items with similar colors, similar desks identified as one.

**Conclusion:**
- **Sparse Tensor is a poweful representation**
  - **Direction weight with computational efficiency**
  - **Segmentation, representation learning, registration detection**
- **Minkowski Engine – open source library for sparse tensot networks (should check it on github)**