3D Reconstruction Occupancy Networks

Team Noisy Pixels:
Shubham Dokania (2020701016)
Shanthika Naik (2020701013)
Sai Amrit Patnaik (2020701026)
Madhvi Panchal (2019201061)

OUTLINE

INTRODUCTION & PROPOSAL

PROBLEM STATEMENT

No well-defined structured representation method in 3D which allows arbitrary topology to represent high-resolution geometry.

Exitance of non-linear classifier for mapping of 3D space to an implicit function

The Task: 3D reconstruction from a single view image

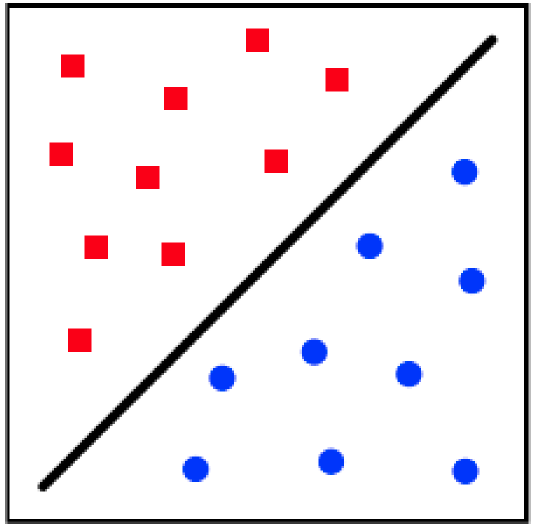
- Input: An image of an object (without camera pose)
- Output: 3D geometry of the object and it's reconstruction in 3D space

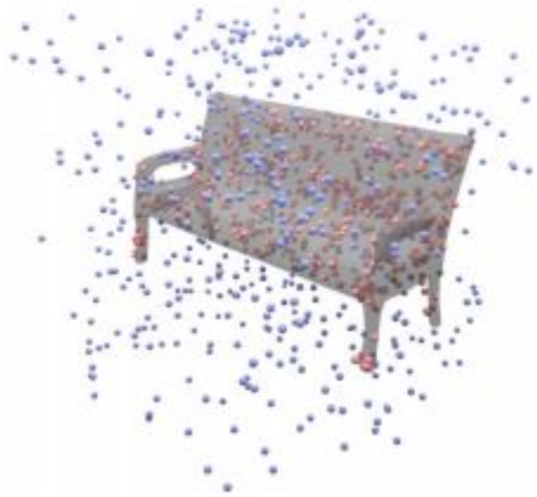
GOALS

1. Implicit learning of 3D surface as a continuous decision boundary of a non-linear classifier
2. Learning the entire occupancy function that can be evaluated at any arbitrary resolution.
3. Extraction of mesh from occupancy function by using a Multi-resolution IsoSurface Extraction(MISE) technique and Marching cubes at inference time.
4. Refining of Final mesh using Fast-Quadric-Mesh-Simplification algorithm and first and second order (i.e., gradient) information.

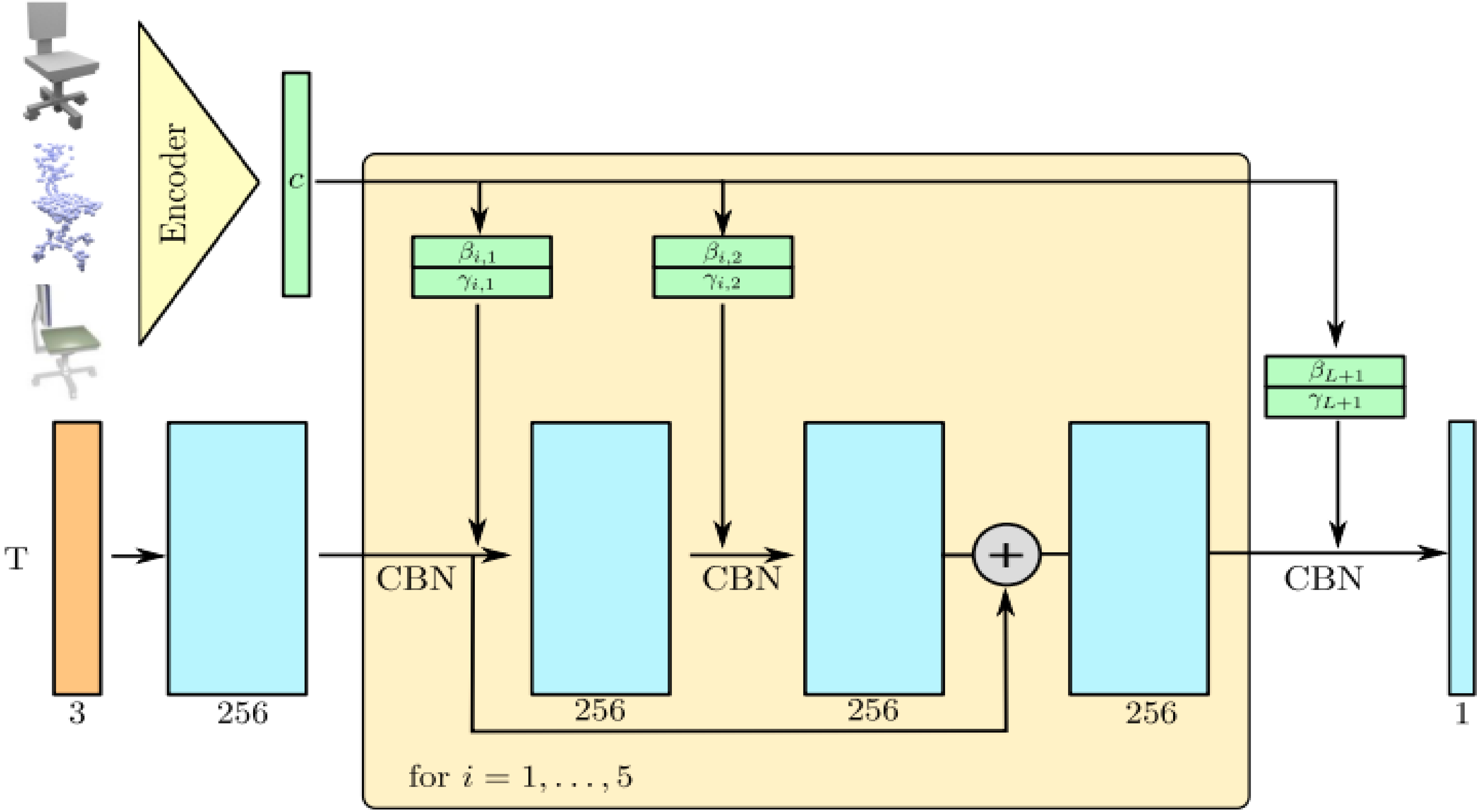
OCCUPANCY NETWORK

- Model 3D surface as a decision boundary of a non-linear classifier.
- Returns an Occupancy probability of a 3D point conditioned on an input image.
- Model 3D surface as a decision boundary of a non-linear classifier.
- Returns an Occupancy probability of a 3D point conditioned on an input image.



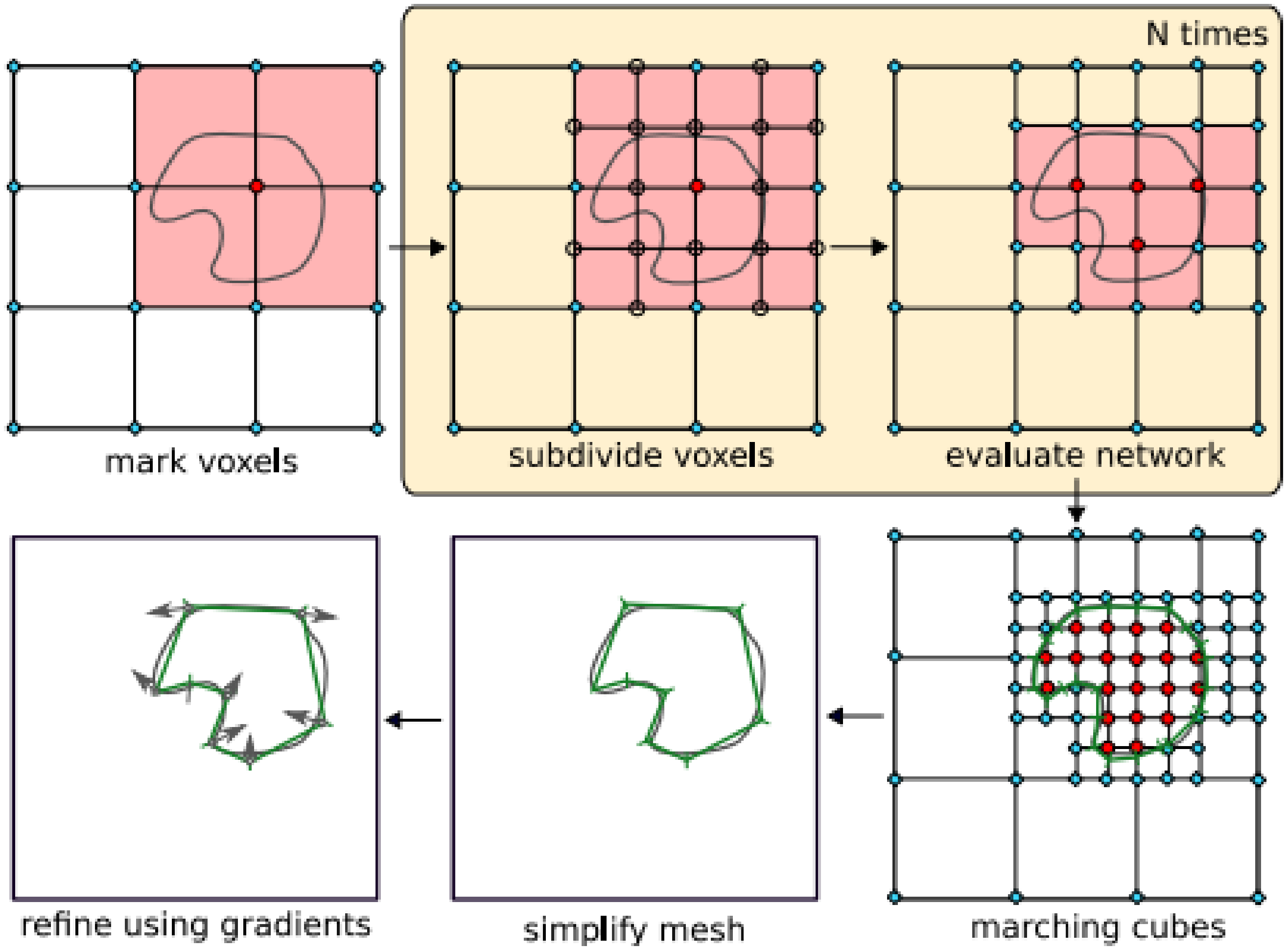


ARCHITECTURE



SURFACE EXTRACTION

- Multi-resolution Iso-Surface Extraction(MISE):
  - Hierarchical method
  - Incrementally builds an octree.



MESH EXTRACTION

- Once the desired resolution is reached, we use the Marching Cube algorithm to extract an approximate Isosurface :

$$\{p \in \mathbb{R}^3 \mid f_\theta(p, x) = \tau\}.$$

- The mesh obtained is further simplified using Fast-Quadric-Mesh-Simplification algorithm.

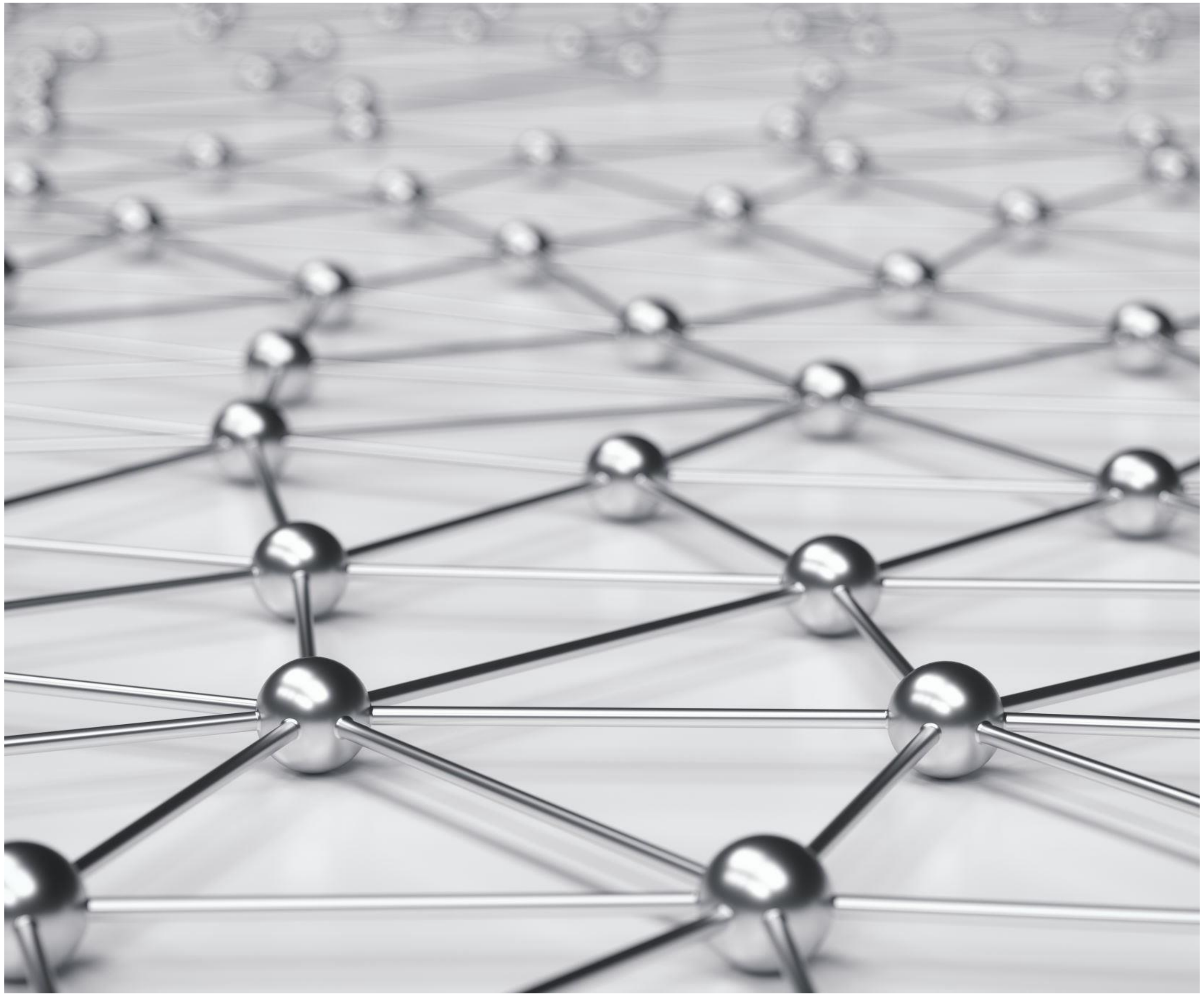
$$\sum_{k=1}^{K} (f_\theta(p_k, x) - \tau)^2 + \lambda \left\| \frac{\nabla_p f_\theta(p_k, x)}{\|\nabla_p f_\theta(p_k, x)\|} - n(p_k) \right\|^2$$

PROJECT EXPECTATIONS

- Analysis of the ShapeNet3D Dataset relevant to 3D reconstruction.
- PyTorch Lightning implementation of Occupancy Network for Single View Reconstruction.
- Visualization of point clouds as inside/outside the surface as output of network.
- Demo for 3D mesh reconstruction.
- Comparison between different performance metrics such as Chamfer distance, IoU, and Normal consistency.
- Open-source release.

IMPLEMENTATION DETAILS

- Data Preprocessing
- Input data visualizations
- Encoder TSNE Visualizations.
- Networks Implemented
- Other Implementation details
- Mesh Extraction Pipeline
- Metrics Implementation



DATA PREPROCESSING

VOXELIZATION AND IMAGE RENDERINGS (32^3).

TSDF-FUSION ON RANDOM DEPTH RENDERINGS OF THE OBJECT.

CREATING WATERTIGHT VERSIONS OF THE MESHES.

CENTERING AND RESCALING OF THE MESHES.

SAMPLING 100K POINTS IN THE UNIT CUBE CENTERED AT 0 AND DETERMINING WHERE THE POINTS LIE.

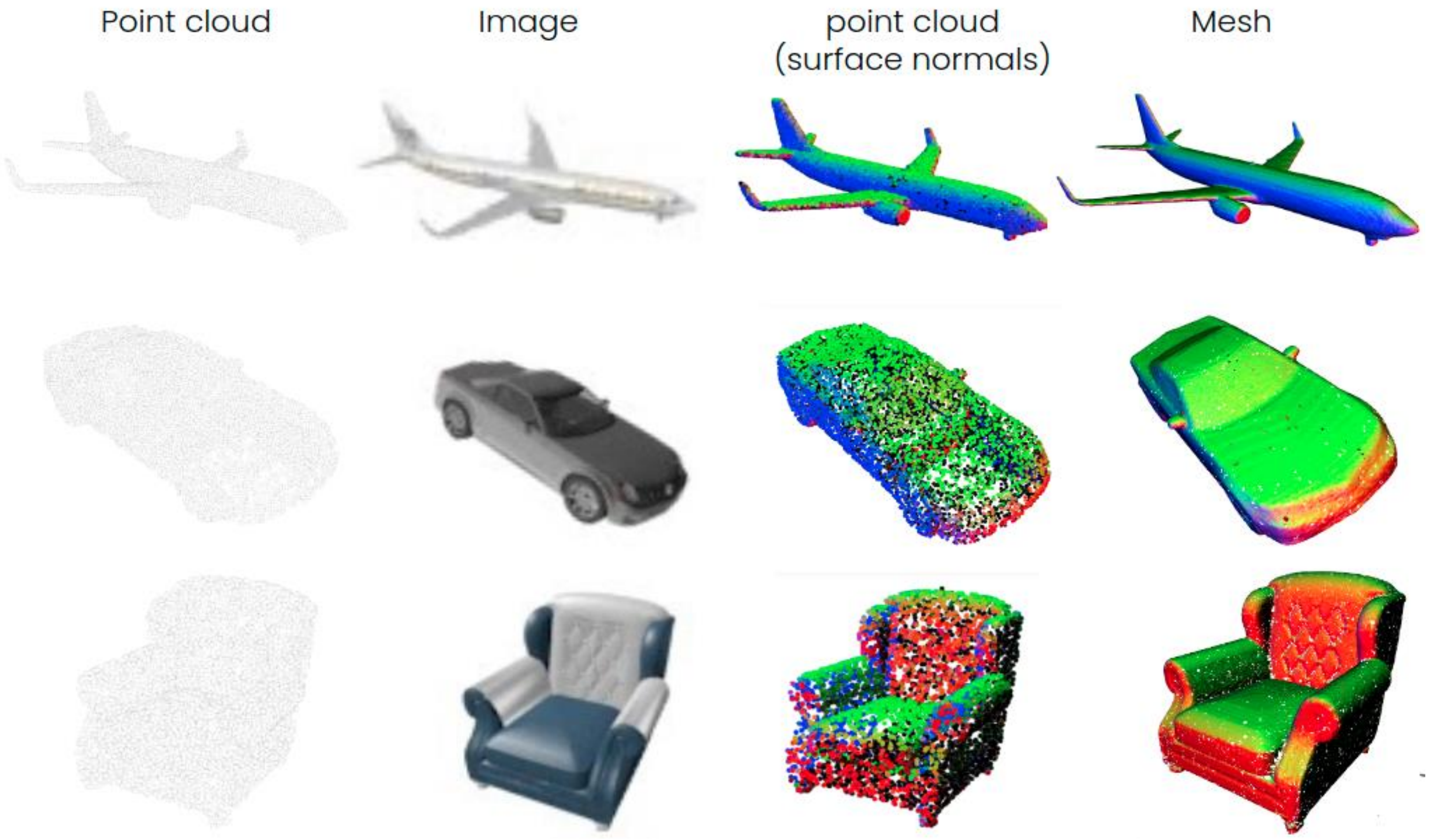
STORING POSITIONS OF THESE POINTS AND THEIR OCCUPANCIES TO A FILE.

VISUALIZATION EXPERIMENTS

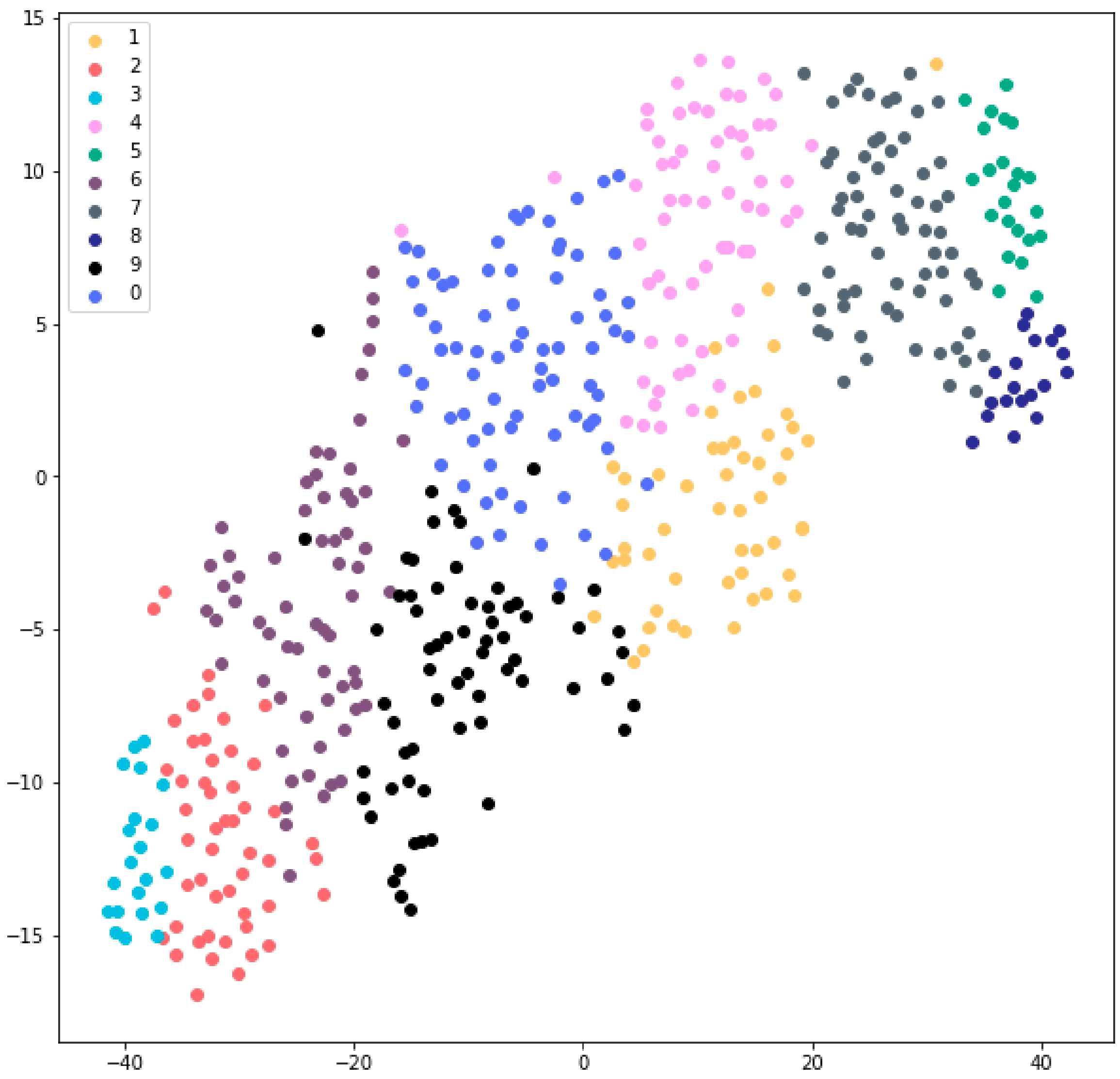
Input point clouds and mesh visualizations

TSNE visualization of encoder embeddings

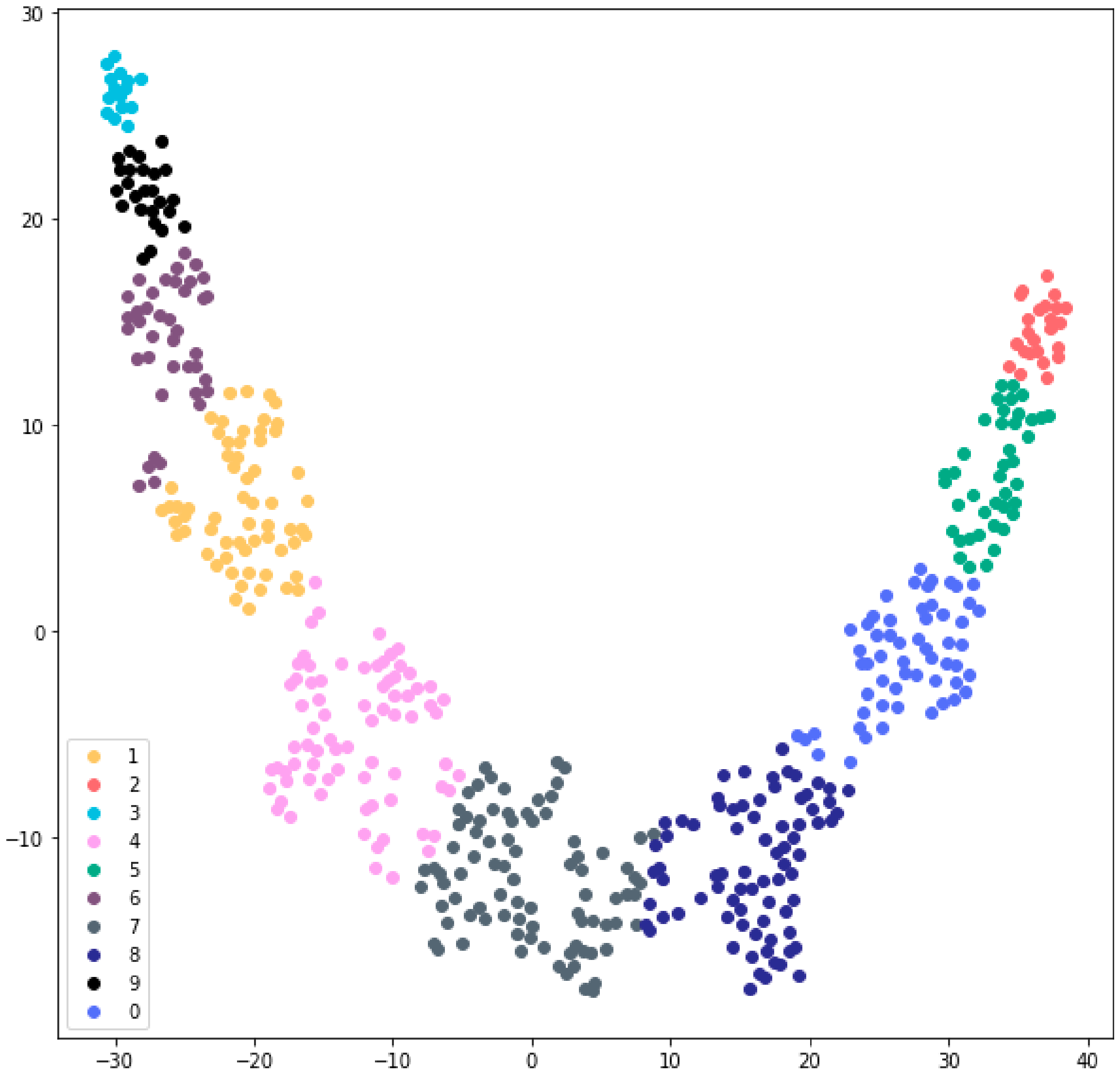
INPUT VISUALIZATIONS



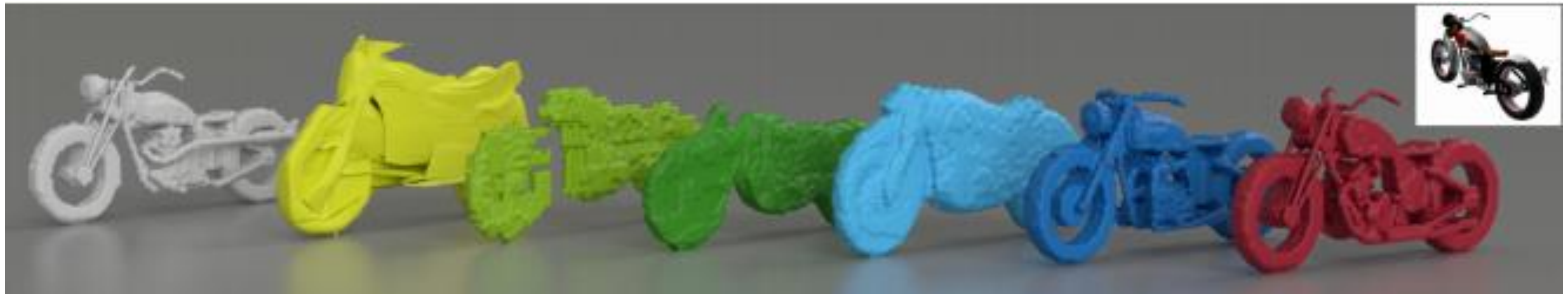
ENCODER VISUALIZATIONS (RESNET 18 ENCODINGS)



ENCODER VISUALIZATIONS (RESNET 50 ENCODINGS)



WHAT SVR NETWORKS LEARN?



Example: From the SVR paper. Demonstrating similarities between SVR networks.

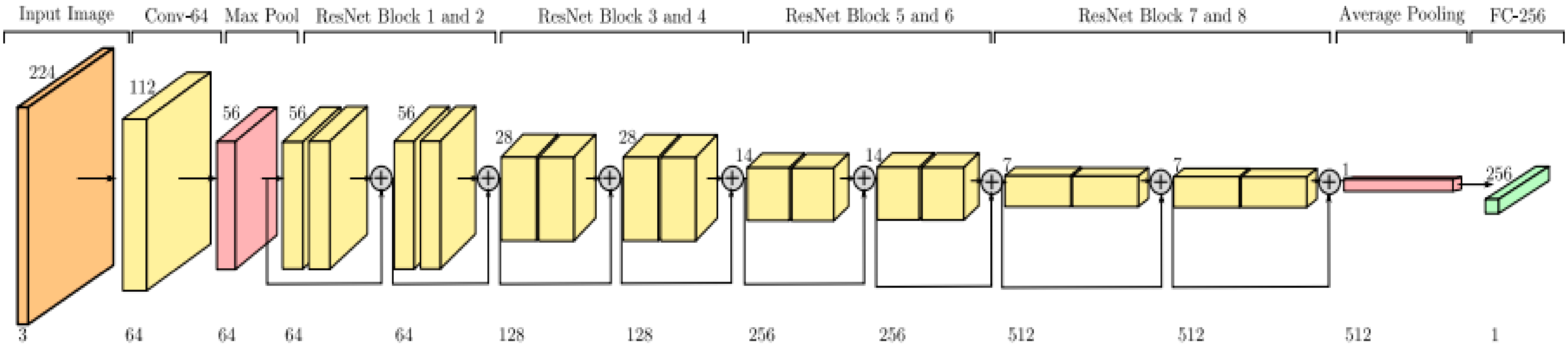
We understand from the paper that SVR (Single-View-Reconstruction) Networks work as a combination of classifier-refiner networks. Here, one part of the network internally acts as a classifier and represents features from a class (hence, networks perform poorly on unseen classes). While, the refiner network takes the features, assumes spatial coherence in the features and tries to model the 3D output. The base 3D structure is learned in the refiner network, and the encoder features work as a class conditional latent variable for this. The refiner network then uses the 3D representation of the base model of the class and applies refinement on the structure in order to minimize the loss.
The authors in [*] also prove this via a series of experiments comparing popular SVR networks to an oracle baseline which works similar to a Nearest neighbour ranking method for model retrieval.

* Tatarchenko M, Richter SR, Ranftl R, Li Z, Koltun V, Brox T. What do single-view 3d reconstruction networks learn?. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2019 (pp. 3405-3414).

ENCODERS

Experimented with:

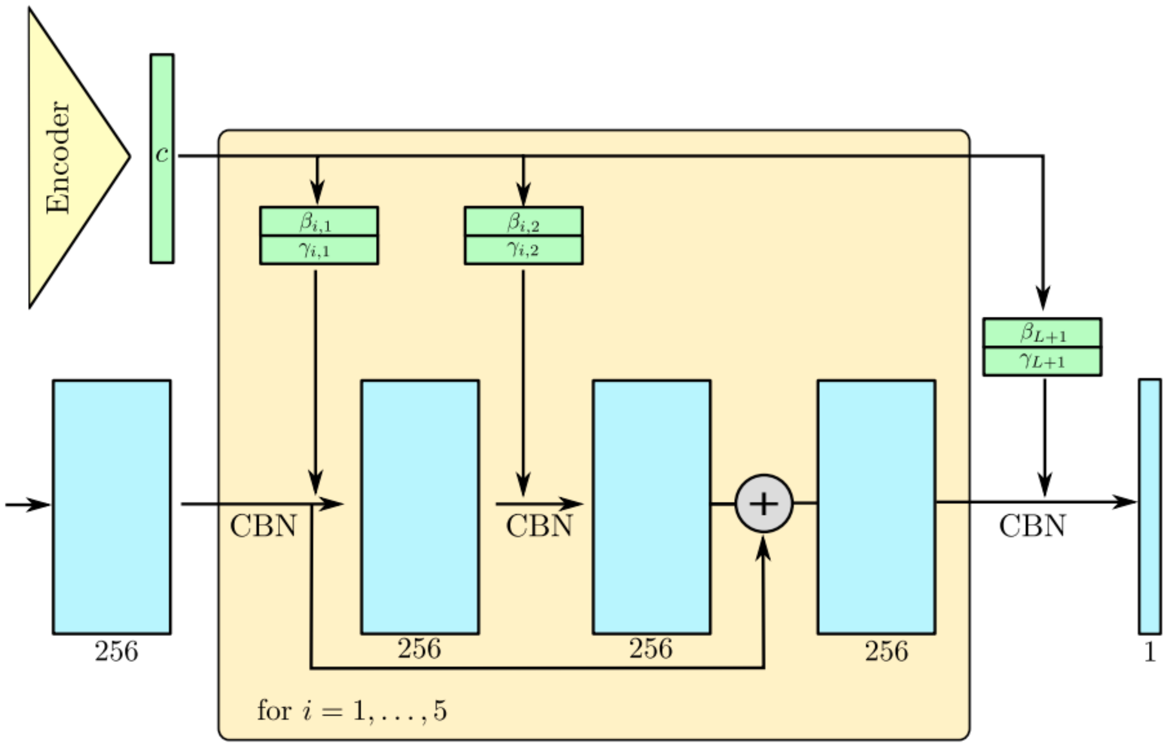
- ResNet:
  1. ResNet18
  2. ResNet50
- EfficientNet:
  1. EfficientNetB0
  2. EfficientNetB1
  3. EfficientNetB5
  4. EfficientNetB7



DECODER

- Consists of a series of ResNet-blocks, each with Conditional Batch Normalization(CBN) and an activation.
- Last block transforms the outputs to a scalar value representing the probability of occupancy.
- CBN Implementation: The conditional encoding c is passed from the encoder through two FC layers to obtain 256-d vectors β(c) and γ(c). CB as follows:

$$f_{out} = \gamma(c)\frac{f_{in} - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta(c)$$



EVALUATION METRICS

Volumetric IOU: defined as quotient of the volume of the two meshes' union and the volume of their intersection.

Chamfer-L1 Distance: Defined as the mean of an accuracy and and a completeness metric.

Normal Consistency Score: Defined as the mean absolute dot product of the normals in one mesh and corresponding nearest neighbors in the other mesh.

IMPLEMENTATION DETAILS

01

The entire network is implemented using Pytorch

02

The training and logging scripts are wrapped inside Pytorch Lightning modules.

03

Binary cross entropy loss is used.

04

The entire preprocessed data is converted to HDF format for efficient memory usage and faster I/O.

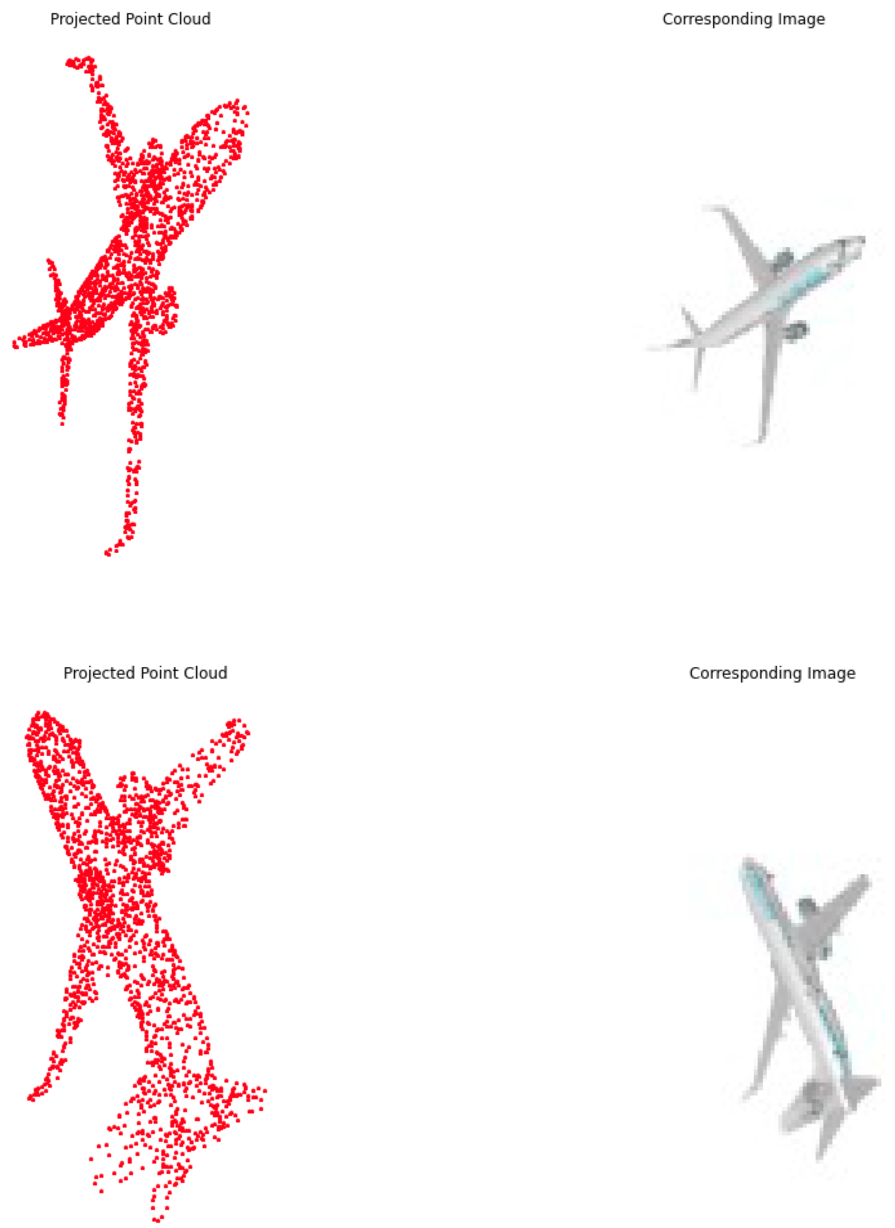
05

All losses and metrics are logged and visualized using TENSORBOARD.

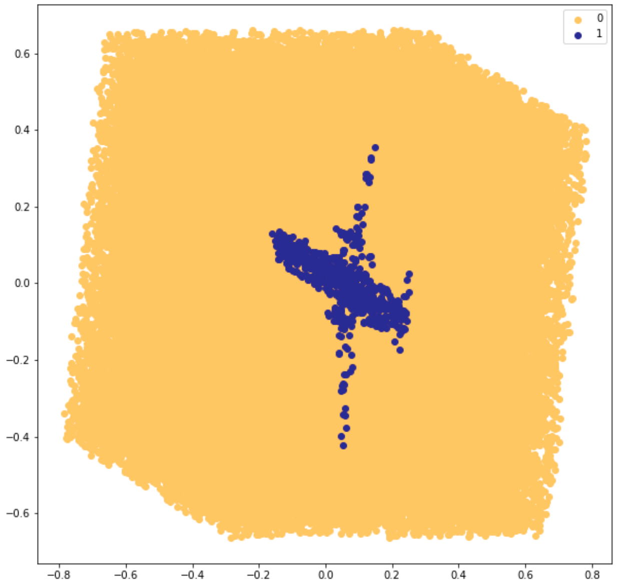
06

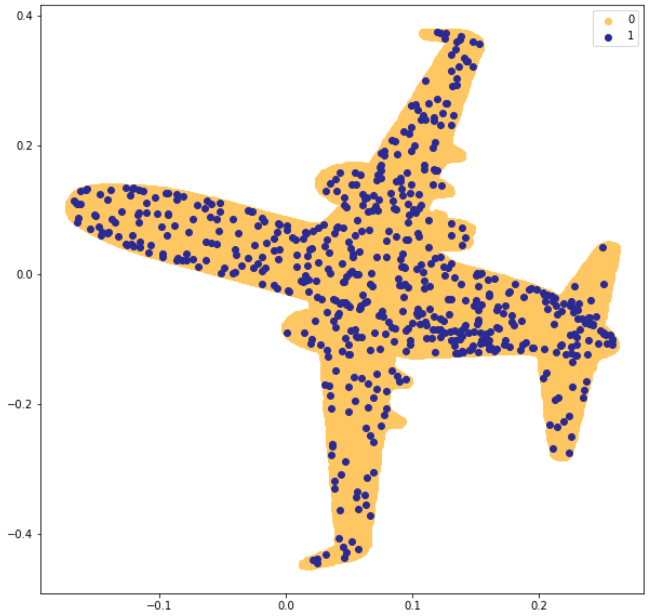
The Mesh extraction and Refinement module is implemented from scratch.

PROJECTING THE POINT CLOUD USING CAMERA MATRIX OF THE IMAGE

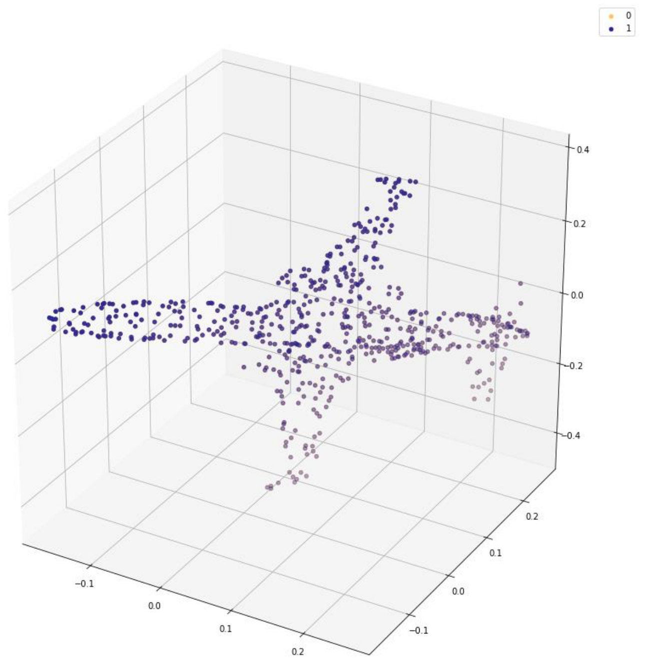


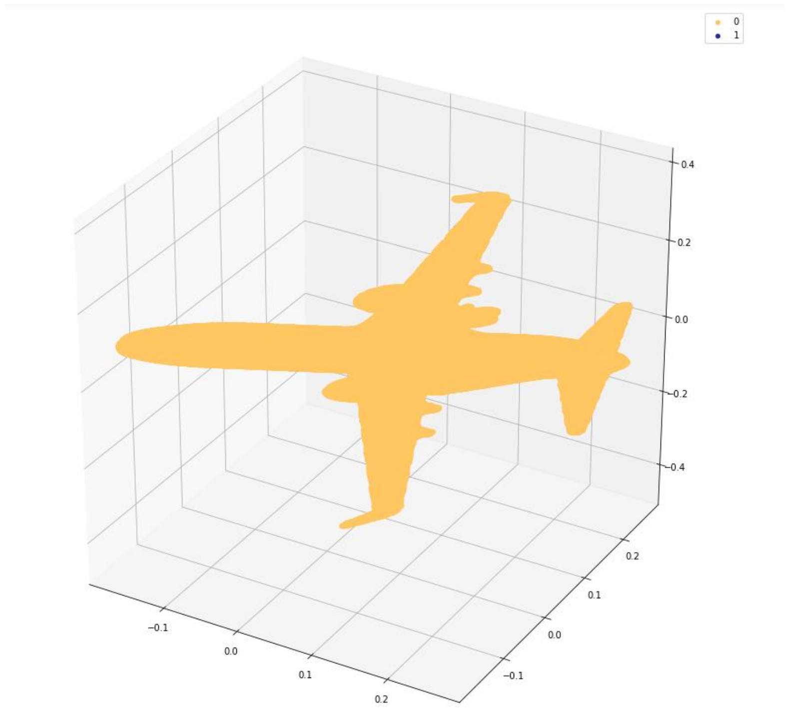
PROJECTING THE POINTS AND POINT CLOUDS TO 2D USING CAMERA MATRIX



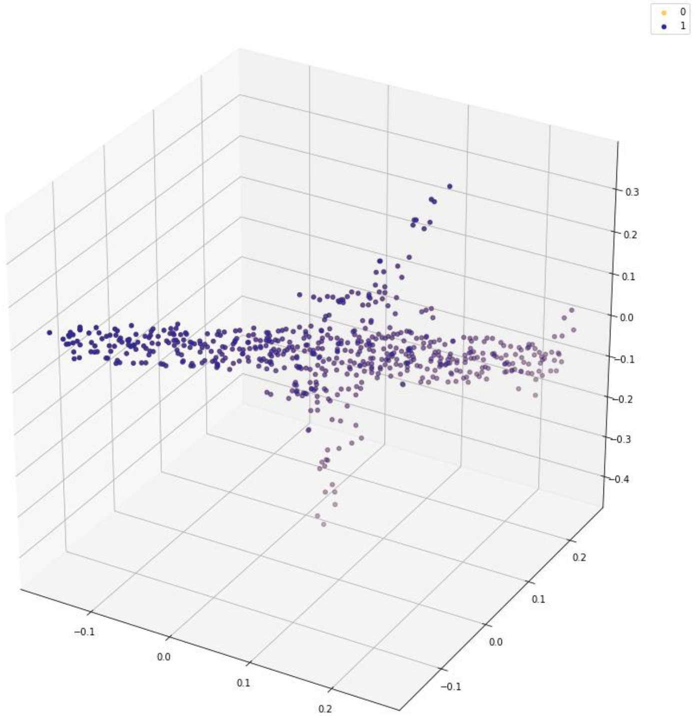


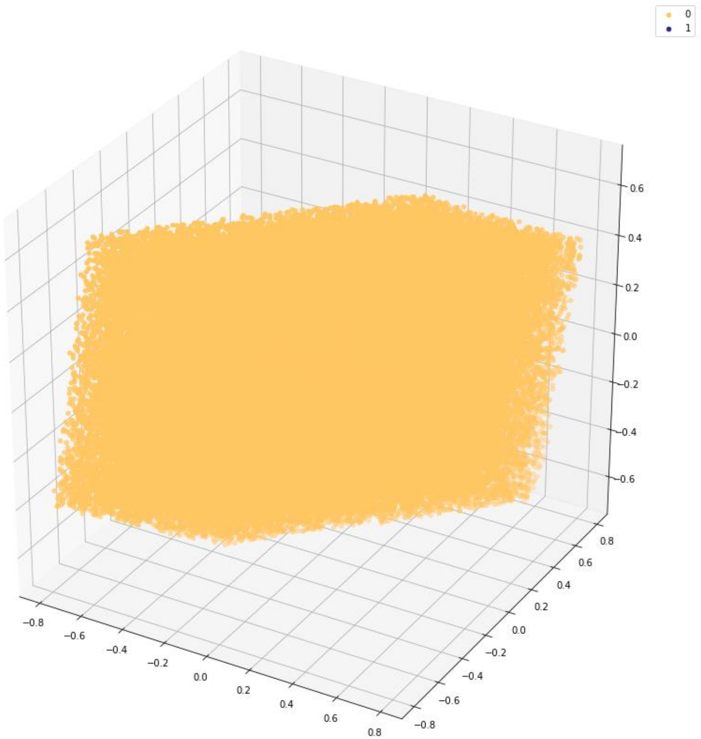
PLOTTING BY PROJECTING THE POINT CLOUD ALONG WITH OCCUPANCY VALES



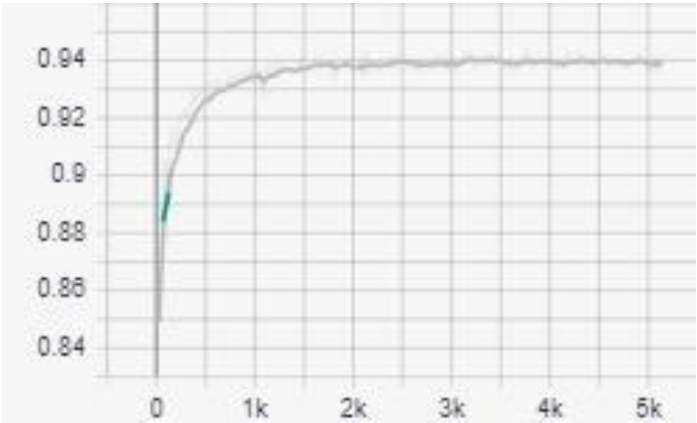


PLOTTING BY PROJECTING THE POINTS ALONG WITH OCCUPANCY VALUES

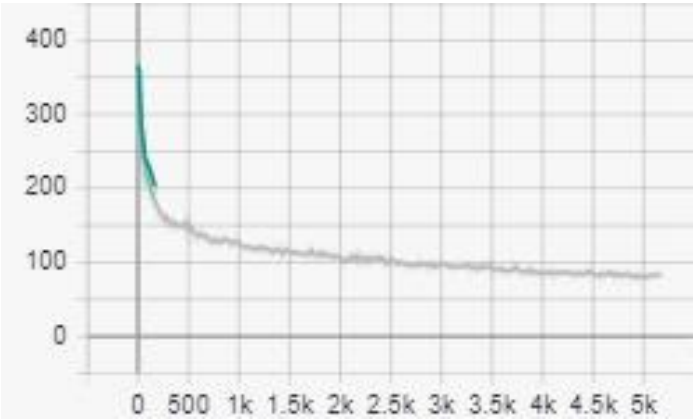




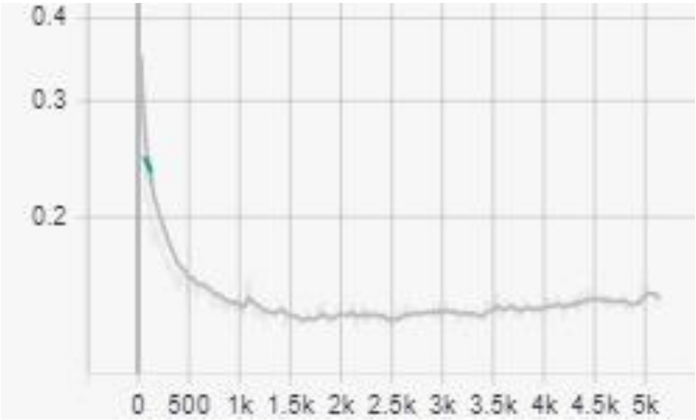
LOSS CURVES (RESNET-18 ENCODER)



Accuracy

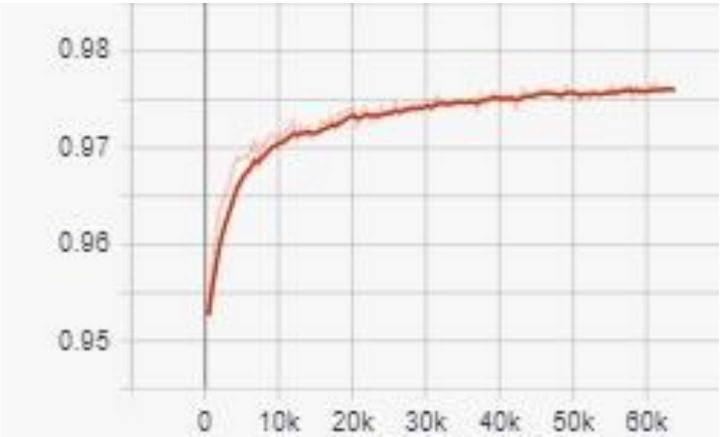


Training Loss

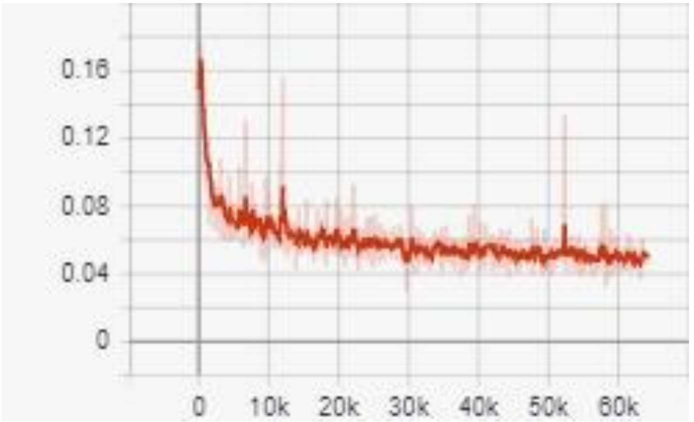


Validation Loss

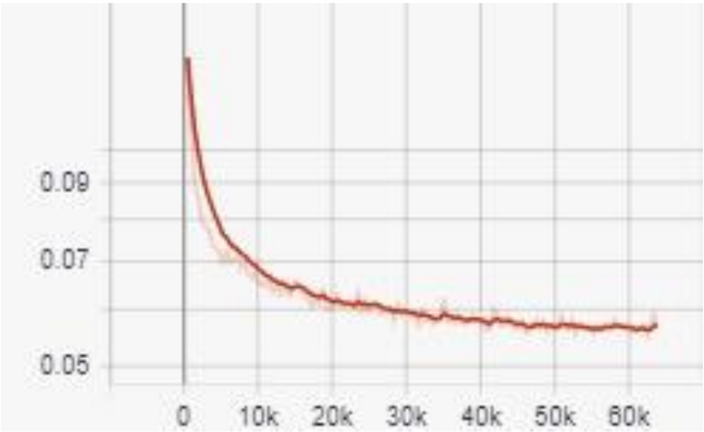
LOSS CURVES (EFFICIENT-NET ENCODER)



Accuracy



Training Loss

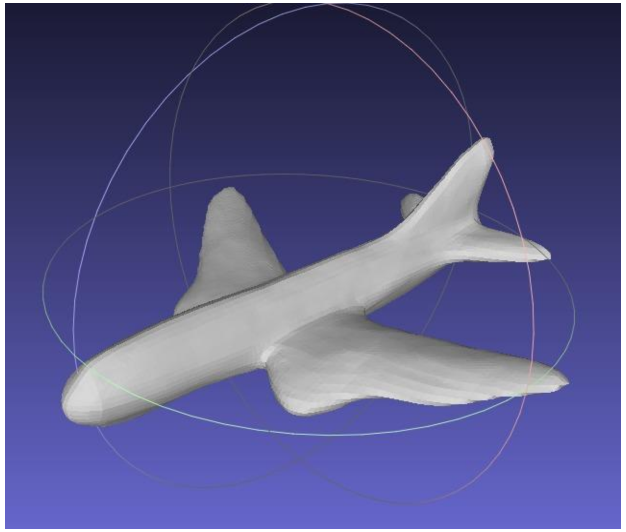


Validation Loss

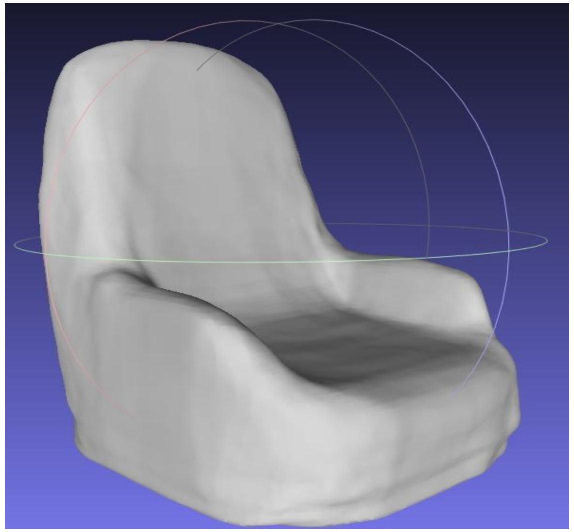
QUANTITATIVE RESULTS

Model	IoU Value	Chamfer-L1 Value	Normal Consistency
Encoder: EfficientNet-B0 Decoder: FC	0.52	0.199	0.82
Encoder: EfficientNet-B0 Decoder: CBN	0.542	0.18	0.844
Encoder: ResNet-18 Decoder: CBN(256 dim)	0.48	0.24	0.802

QUALITATIVE RESULTS



Aeroplane Class



Chair Class

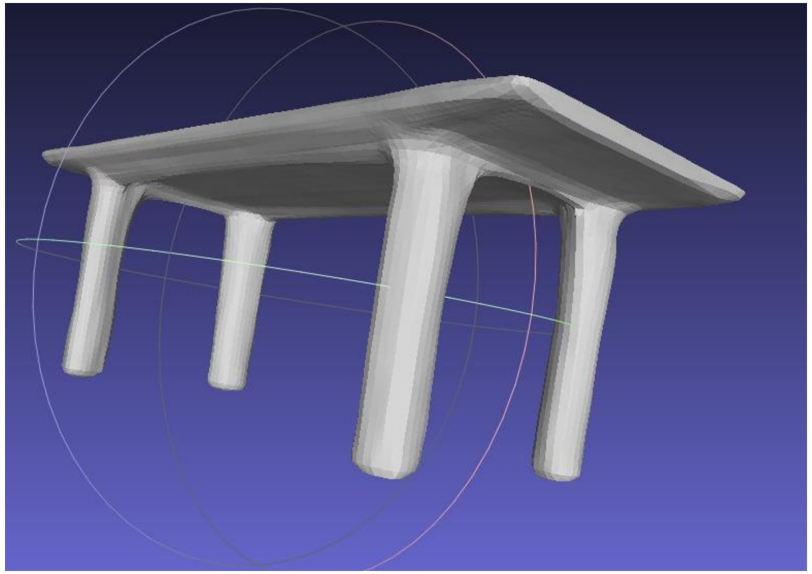


Table Class

ACHIEVEMENTS

- Train End-to-End encoder-decoder network for Occupancy detection over the full dataset.
- Compare multiple variants of encoder-decoder network models
- Use camera matrix for better orientation of occupancy points for better accuracy
- Use MISE and Marching Cubes along with surface normals to ensure smooth mesh generation and recovery.
- Understand working of SVR (Single-View-Reconstruction) Networks.
- Build Visualizations to for better analysis of models and results.

FURTHER STEPS

- For future work, we plan to continue on the work and explore more variations such as:
  - Uncertainty in occupancy predictions in different poses (explore most informative object pose)
  - Role of positional-encoding in improvement of occupancy detection
  - Prediction of camera pose for multi-view loss function generation
  - Mesh refinement via graph neural network prediction of surface normals and comparison with gradient based method.

REFERENCES

- Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S. and Geiger, A., 2019. **Occupancy networks: Learning 3d reconstruction in function space**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4460-4470).
- Tatarchenko M, Richter SR, Ranftl R, Li Z, Koltun V, Brox T. **What do single-view 3d reconstruction networks learn?**. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2019 (pp. 3405-3414).
- Vakalopoulou, M., Chassagnon, G., Bus, N., Marini, R., Zacharaki, E.I., Revel, M.P. and Paragios, N., 2018, September. **AtlasNet: multi-atlas non-linear deep networks for medical image segmentation**. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 658-666). Springer, Cham.
- Bartsch, M., Weiland, T. and Witting, M., 1996. **Generation of 3D isosurfaces by means of the marching cube algorithm.** *IEEE transactions on magnetics*, *32*(3), pp.1469-1472.
- M. Garland and P. S. Heckbert. **Simplifying surfaces with color and texture using quadric error metrics**. In Visualization'98. Proceedings, pages 263–269. IEEE, 1998.
- Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H. and Xiao, J., 2015. **Shapenet: An information-rich 3d model repository**. *arXiv preprint arXiv:1512.03012*.
- H. de Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. C. Courville. **Modulating early visual processing by language**. In Advances in Neural Information Processing Systems (NIPS), 2017.