Adapting 2D ViTs for 3D Point Cloud Understanding

Bastian Berle - Christian Traxler

Agenda

1. Introduction, Motivation, Outline (1:30 min)
2. Pix4Point (3 min)
3. Adapt Point Former (2 min)
4. Diff Renderer (2:30 min)
5. Final Comparison / Outlook (1 min)

Introduction & Motivation

We train models built for **Point Cloud Understanding** using **Pretrained 2D ViTs**.

Why ViTs?

1. Transformer Scalability & Inductive Bias Flexibility
2. Utilize 2D Pre-Trained Knowledge
3. Although, Transformers are Data Hungry

In the lecture, we have classified models into:

1. Projection-based Models → Our Differential Renderer
2. Voxel-based
3. Point-based → Pix4Point & AdaptPointFormer

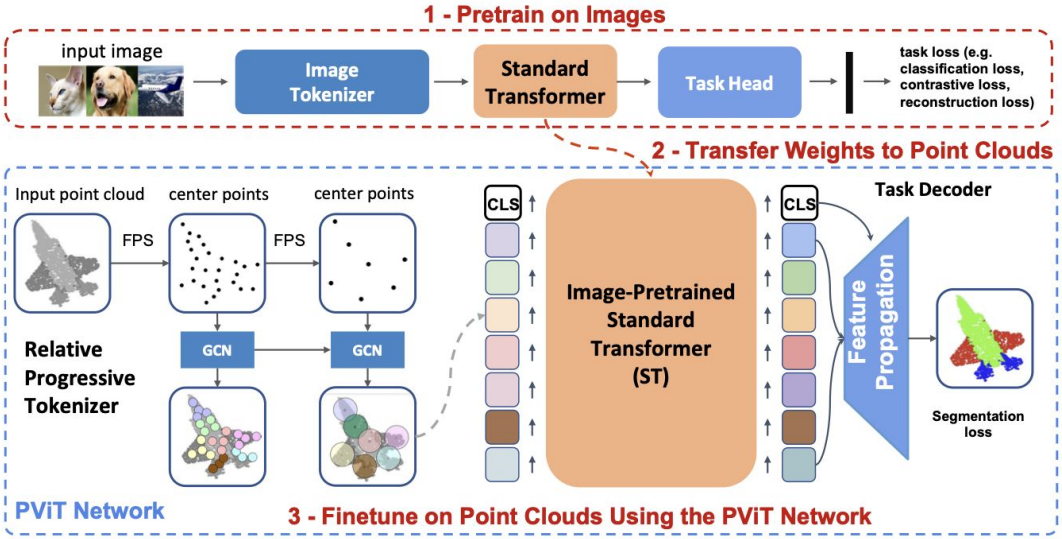
ScanObjectNN

- 15,000 Objects across 15 Categories
- Each Object contains ~2,048 Points
  - only geometry, i.e. x,y,z, and no RGB
- Real World Data: Noisy, Incomplete, and Misaligned
- Available in different Cuts:
  - Clean Segmented, Incomplete Noisy, etc…

We focus on **3D Object Classification**

[Revisiting Point Cloud Classification: A New Benchmark Dataset and Classification Model on Real-World Data (2019)]

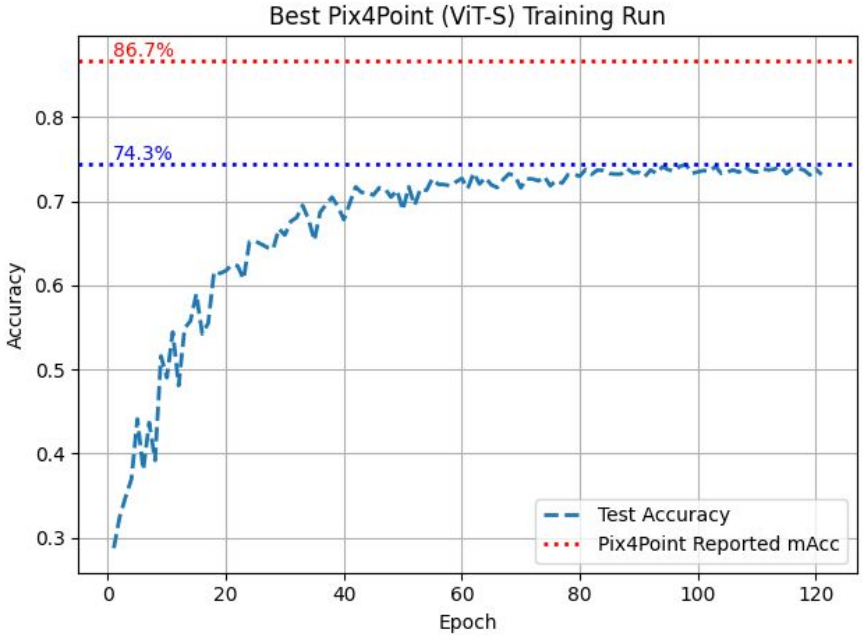
Pix4Point

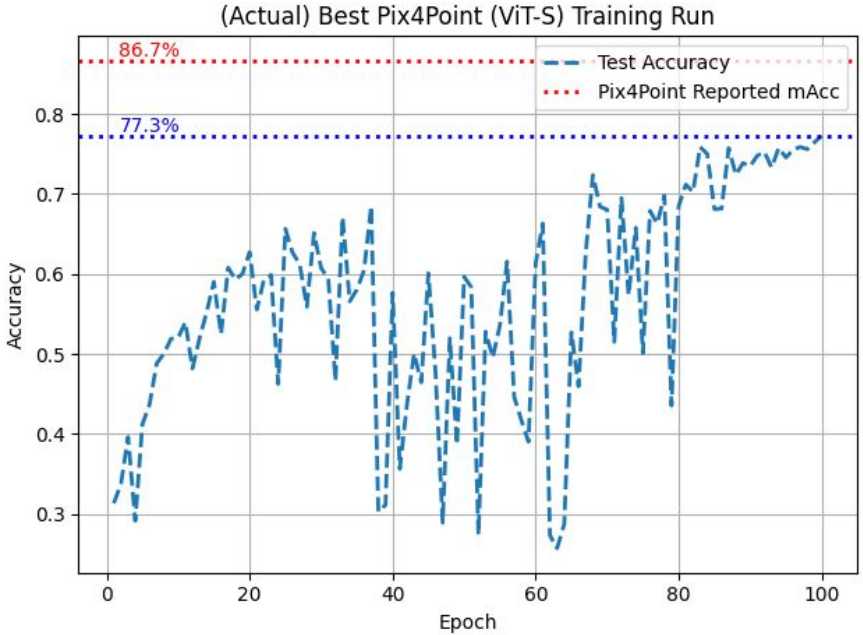


- Progressive Point Patch Embedding (P3Embed)
- Pretrained ViT-S using MAE on ImageNet
- Point Vision Transformer (PViT)
- (Full) Finetune of PViT (not Frozen)

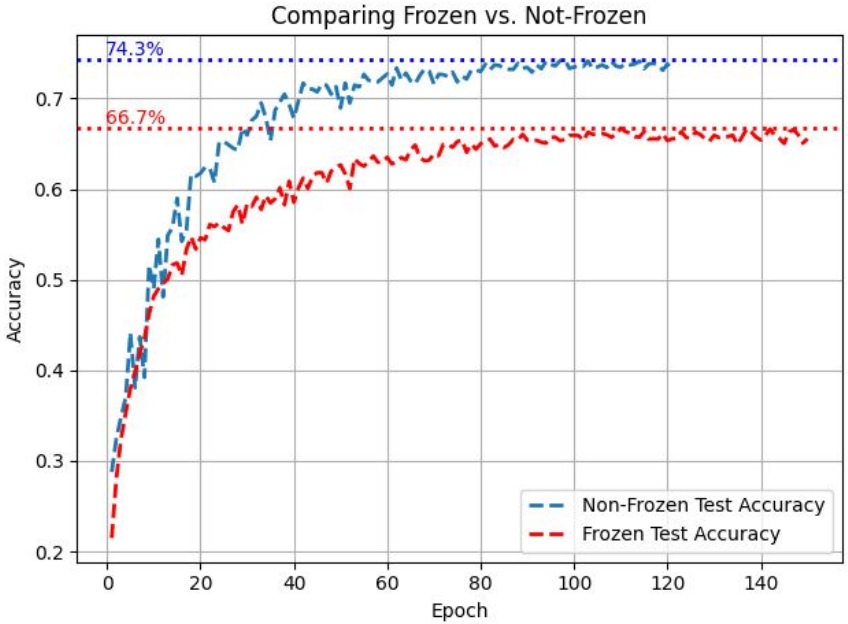
Pix4Point: Image Pretrained Standard Transformers for 3D Point Cloud Understanding (2024)

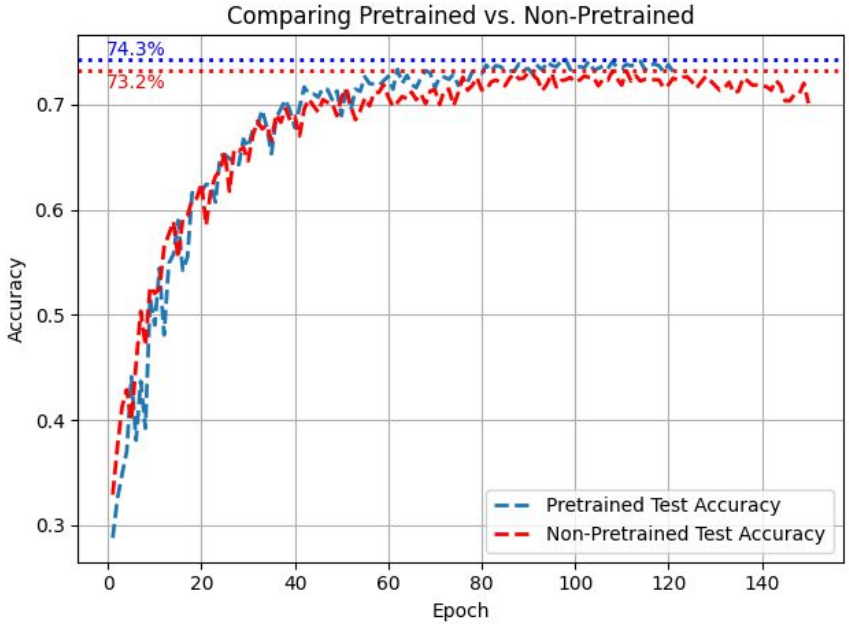
Pix4Pix Results



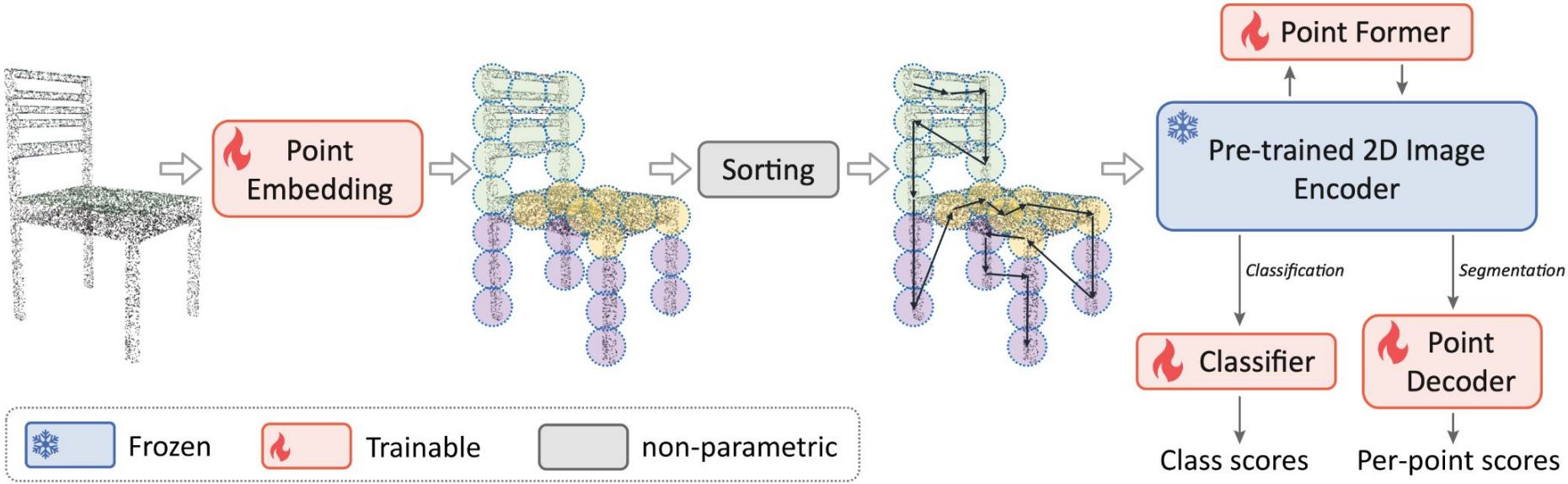


Pix4Pix Results

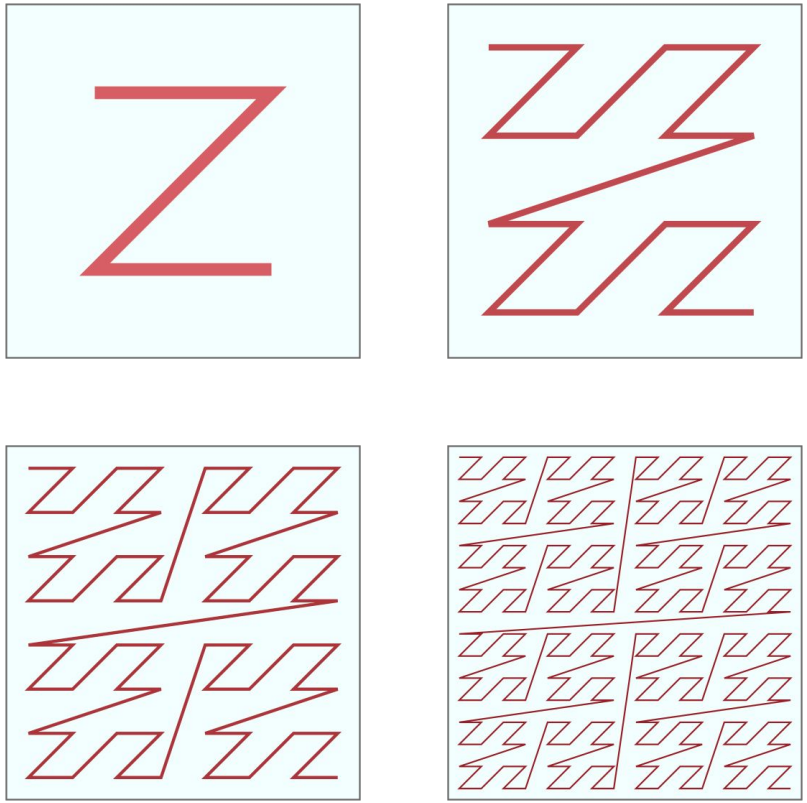




Adapt PointFormer



Adapt PointFormer: 3D Point Cloud Analysis via Adapting 2D Visual Transformers (2024)

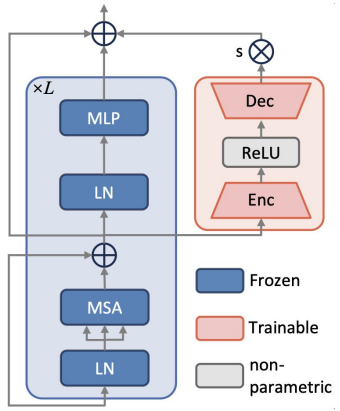


Results

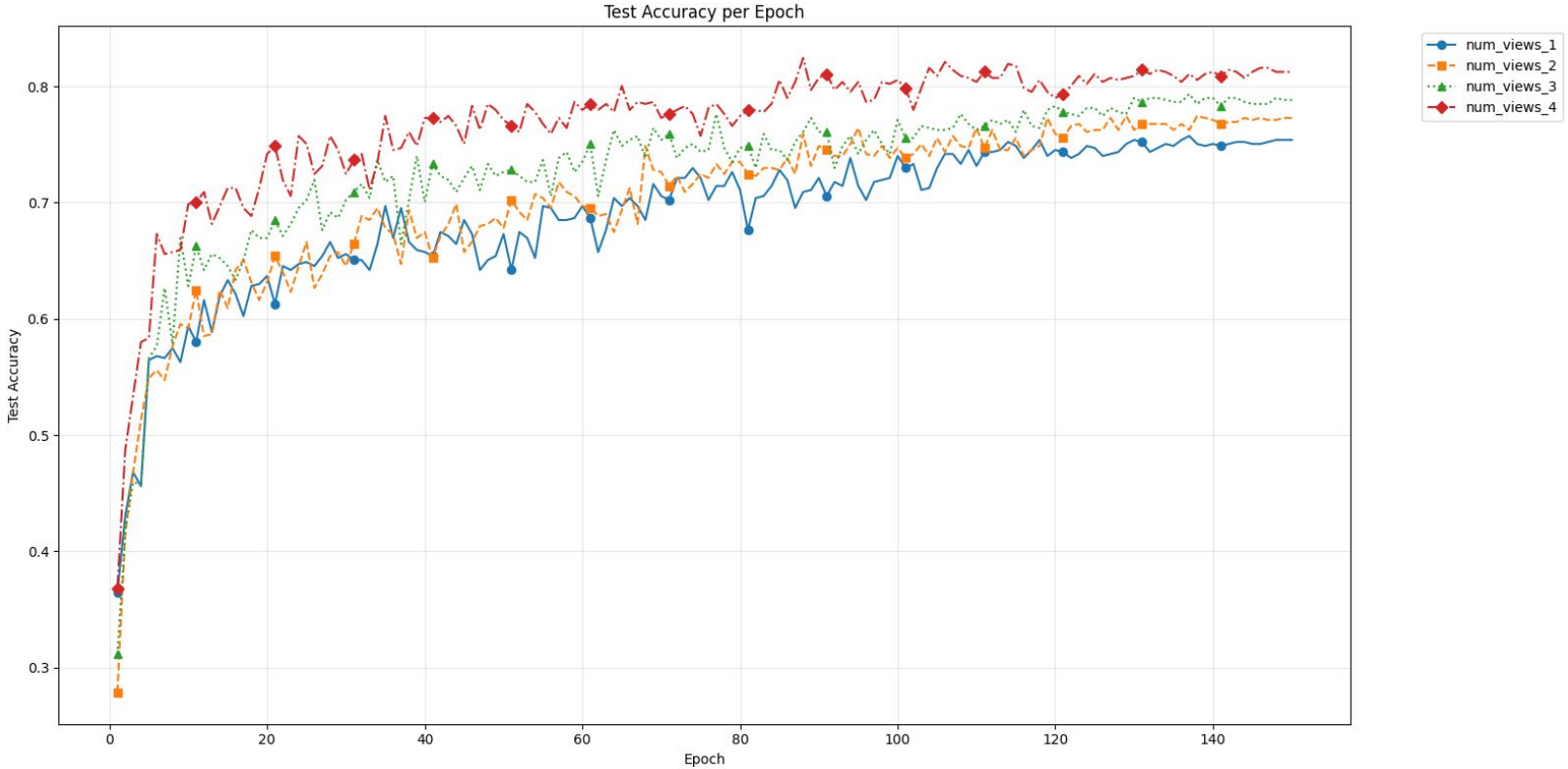

Comparing APF with XYZ & Height vs. APF with XYZ
83.133%
82.272%
Accuracy
Epoch
0.45
0.50
0.55
0.60
0.65
0.70
0.75
0.80
0
20
40
60
80
100
APF with XYZ & Height
APF with XYZ

Custom Concept

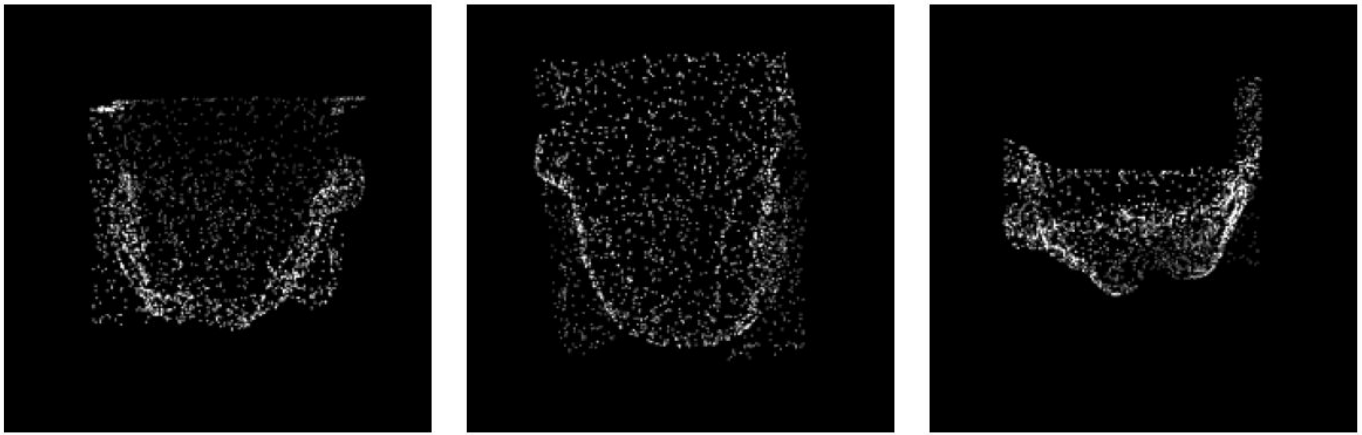
- Go back to more intuitive approach… projections



Results


Test Accuracy per Epoch
Test Accuracy
Epoch
0.3
0.4
0.5
0.6
0.7
0.8
0
20
40
60
80
100
120
140
num_views_1
num_views_2
num_views_3
num_views_4

Results - Differential Renderer



Number of Views		*1*	*2*	*3*	*4*
Renderer	*Fix*	0.7453	0.7659	0.7935	0.8055
	Diff	0.6248	0.7108	0.7091	0.7263

MVTN: Multi-View Transformation Network for 3D Shape Recognition (2021)

Discussion

Base Models: No Access to Same Base Models

Hyperparameters:

- Unreported Values
- Trustworthiness (Different Values in Paper vs Repository)
- Led to Bad Results (Instability in Training)

Dataset: Trained only on Classification vs Multiple Tasks