Patch by Patch Coarse-to-Fine 3D Shape Refinement

Kristiyan Sakalyan
TUM
Arcisstraße 21 80333 München
kristiyan.sakalyan@tum.de

Stephan Schmiedmayer
TUM
Arcisstraße 21 80333 München
stephan.schmiedmayer@tum.de

Abstract

We propose a novel architecture designed to enhance the geometric detail in 3D shape generation by refining sparse and noisy point clouds. Our approach combines the strengths of Point-Voxel Convolution (PVConv), Point Transformer, and PU-Net to refine shapes generated by Point-Voxel Diffusion (PVD) in a patch-by-patch manner. The refinement process is designed to enrich geometric details, resulting in 3D models of higher realism and improved spatial resolution. Empirical results highlight the architecture's capability to improve the quality of generated 3D shapes.

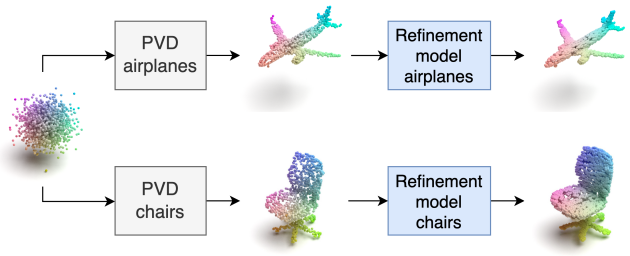


Figure 1. **Pipeline Overview.** Our 3D shape generation pipeline consists of two stages: starting with random noise input for the PVD model to generate a coarse shape, followed by refinement that adds geometric details and upsamples the point cloud for a detailed, high-resolution shape.

1. Introduction

Our research focuses on enhancing 3D shape generative modeling, crucial for vision, graphics, and robotics, by addressing the lack of geometrical details and sparsity of current state-of-the-art generative models [1, 4, 5, 10, 14]. We introduce a novel method, "Patch by Patch Coarse-to-Fine 3D Shape Refinement," which incorporates advanced techniques from PVCNN [7], Point Transformer [13], and PU-Net [11] to refine coarse shapes produced by PVD [14].This approach aims to significantly improve detail and realism in 3D shape refinement.

We focus on training separate models for each unique kind of shape, focusing on chairs and airplanes (see Figure 1).

2. Related Work

2.1. Point Cloud Upsampling

PU-Net introduces an innovative approach for point cloud upsampling, targeting the enhancement of sparse point clouds into denser, more detailed representations. It employs a unique feature duplication and expansion strategy to increase the point set, followed by a sophisticated refinement process using Multi-Layer Perceptrons (MLPs) to reconstruct the denser point cloud accurately. As our work is influenced by the PU-Net architecture and employ similar techniques, we will provide further details in Section 3.4.

2.2. Point Cloud Refinement

Point Diffusion-Refinement (PDR) [8] focuses on generating point clouds based off incomplete 3D point clouds. Similar to our work, their overall generation pipeline first queries a coarse shape from a diffusion model which is refined by a separate model. However both initial shape generation and refinement are conditioned on a partial point cloud, which alleviates the ground truth problem we encountered, however requires additional input to the model.

2.3. Point Cloud Upsampling and Refinement

Point Cloud Upsampling via Disentangled Refinement [6] addresses very sparse, non-uniform point clouds with the main goal of very high upsampling (up to 16X). In two sequential steps, they first upsample and then refine the point cloud with separate models. For training and inference, they also work on patches, and achieve some invariance to noise, however do not train to refine the very noisy input point clouds our work is focusing on.

2.4. Point Cloud Shape Generation

Point-Voxel Diffusion [14] introduces a new model to performs 3D shape generation and completion by integrating point clouds and voxel representations, enhancing detail and processing efficiency. While it advances the diffusion process for 3D data, enabling the creation of high-fidelity structures, the framework is not without limitations. Specifically, the use of 2048 points for shape generation results in coarse shapes lacking in geometrical details. This constraint highlights a significant challenge in generative models, pointing to a need for solutions that can generate more detailed and complex 3D geometries beyond the low-resolution point sets currently employed.

Latent Point Diffusion Models for 3D Shape Generation (LION) [12] introduces an approach to 3D shape generation by leveraging the capabilities of latent point diffusion models. This method significantly advances the generative modeling of 3D shapes by producing detailed and diverse structures with a high degree of fidelity. A key feature of LION is its generation process, which, similar to previous models, typically outputs 2048 points. While this establishes LION as a capable tool for creating complex 3D shapes, the limitation to generating only 2048 points remains a notable concern.

3. Background

3.1. PVConv Layer

Point-Voxel Convolution (PVConv) is a fundamental operation in our proposed architecture. It efficiently processes 3D data by combining voxel-based and point-based methods to optimize feature extraction. It voxelizes point clouds, applies 3D convolution for feature aggregation, and then devoxelizes the results. In parallel, it extracts fine-grained features directly from points using a Multi-Layer Perceptron (MLP). The final output is a fusion of the coarse-grained voxel and fine-grained point features, optimizing the layer for both accuracy and computational efficiency. The layer can be seen in Figure 6.

3.2. Point Attention Layer

Point Attention dynamically assigns significance to individual points in a point cloud based on their context and the task at hand. It operates by computing attention scores for each point, which are used to scale the point's features, thereby emphasizing features that are more relevant for the given processing task. This mechanism enables the model to focus on critical areas within the point cloud, enhancing its ability to capture complex spatial relationships and geometric structures. The exact mechanism is illustrated in Figure 8.

3.3. Global Attention Layer

The global attention layer assesses and emphasizes the importance of each point in relation to all other points across the entire dataset, thereby capturing global dependencies. It incorporates positional encoding to retain spatial information, ensuring that the model recognizes the original positioning of points within the 3D space. This approach enables the network to understand and leverage the spatial context, significantly enhancing feature representation.

3.4. Upsampling Layer

In **PU-Net** [11], a feature expansion strategy is employed to increase the density of point clouds, detailed in Figure 7. Similarly to this, our upsampling layer multiplies the point cloud size by the up_ratio, enhancing point cloud resolution. Additionally, one-hot encoded vectors are appended to provide unique identifiers for points, which help with the dispersion of duplicates and enriching geometric detail.

4. Proposed Approach

In this section, we present our novel architectural framework that aims to refine coarse 3D shape point clouds, in a parallel patch by patch manner. Our approach combines principles from PVCNN, Point Transformer, and PU-Net to enhance the realism and accuracy of 3D shapes generated by existing models.

4.1. Architecture

Our proposed architecture, as illustrated in Figure 2 utilizes a sequence of Point-Voxel Convolution (PVConv) layers across multiple resolutions to capture shape features at varying scales. This multi-scale feature extraction is analogous to and inspired by the hierarchical learning approach of PointNet++ [9]. Integrating Point Transformer layers, the network is capable of learning and emphasizing salient features specific to the geometry at each hierarchical level.

The architecture incorporates a Max Pooling layer designed to retain spatial dimensions by broadcasting the maximum activation across the respective field. This approach is inspired by the spatial preservation strategy utilized in PVCNN [7], and used in PVD [14]. The resulting feature maps are concatenated and introduced to a classifier composed of successive Conv2D layers, coupled with a Global Attention mechanism that refines the feature representation.

Subsequently, the feature set is expanded in accordance with the predefined upsampling ratio (up_ratio), and one-hot encoding is incorporated to differentiate the duplicated points, facilitated by the Upsample Layer. Finally, Conv2D and Global Attention Layers are applied to accurately reconstruct the coordinates, resulting in an expanded point cloud by the factor of up_ratio.

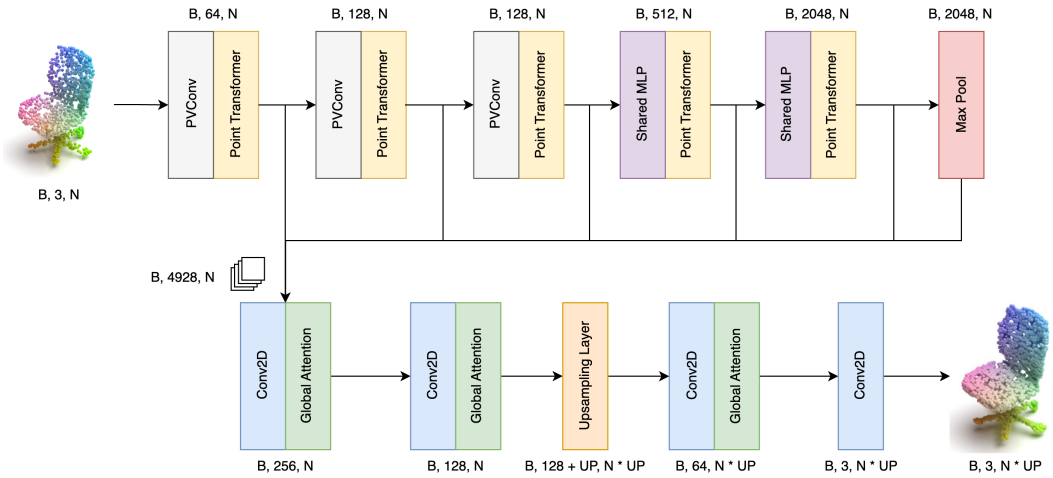


Figure 2. **Proposed architecture** combining layers of existing networks in a first hierarchical feature extraction and second refinement + upsampling pipeline.

5. Experiments

Our findings support that the proposed architecture significantly enhances the geometric detail of coarse shapes. However, it is important to note that the efficacy of this enhancement is contingent upon the quality of the output generated by the PVD [14] model. We upsample the point clouds of size 2048 generated by PVD with a *up_ratio* of 3 to 6114 points in 4 non-overlapping patches.

5.1. Training

5.1.1 Dataset

Refining novel shapes generated by PVD presents a challenge due to the absence of suitable ground truth values for direct comparison. As such we approximate the refinement step during training based on the assumption that the population of point clouds generated by PVD has the same distribution as point clouds sampled from the ShapeNet dataset [2] with added noise. Given that PVD training involved noise removal from ShapeNet point clouds, this assumption is deemed reasonable. [14]

During training, the model is conditioned to refine and upsample fixed-sized patches. As provided by the dataset of PointFlow [10], 6144 points are randomly sampled from a given ShapeNet 3d model. To generate the ground truth patch, the 1536 nearest neighbors of a randomly chosen point are extracted. To simulate the imperfections observed in PVD's output and thus generate the model's input point cloud, we first sub-sample the patch to 512 points and then augment each point with noise drawn from $\mathcal{N}(0, 0.05)$. This training protocol is illustrated in Figure 3.

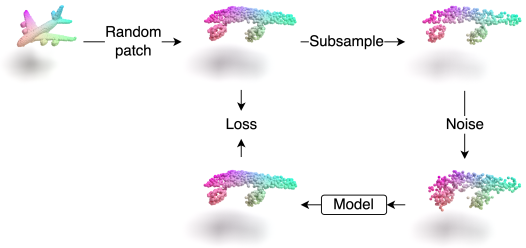


Figure 3. **Training Overview**: We begin by selecting a random patch containing $512 * up\_ratio$ points. This patch is then augmented by subsampling down to 512 points and introducing noise. Subsequently, it is fed to our model to produce a refined patch. The loss is then computed between the original (pre-augmentation) and the model's output.

5.1.2 Loss

Our approach adopts the loss function utilized in PU-Net [11], comprising the Earth Mover's Distance (EMD) [3] and repulsion loss, however with a weighting of 10-to-1 instead of 100-to-1. The EMD facilitates the preservation of the global shape integrity of the point cloud, while the repulsion loss ensures the dispersion of points, thereby enabling the generation of new points to augment geometric detail. The repulsion loss is especially important to ensure the feature duplication of the Upsampling layer does not lead to duplicate output points.

5.2. Inference

5.2.1 Patch creation

Having been trained on patches, during inference, the point cloud generated by PVD needs to be partitioned into patches with roughly 512 points each.

To create geometrically cohesive patches we adopted spectral clustering. By first creating a k-NN graph from the point cloud and using spectral clustering we generate patches of similar size. A correct choice of k required to trade off between a focus on creating the patches clustered close to the inherent point cloud geometry (small k) and a graph with less variation in patch point cloud size (big k). We achieved best results with $k = 10$, transparently increased for disconnected graphs.

Spectral clustering is not used during training due to the fact that it does not produce point clouds of exactly equal size and thus prevents alignment in memory.

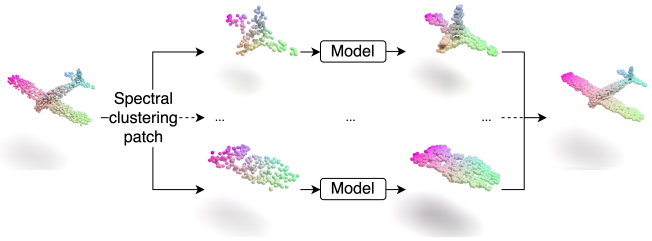


Figure 4. **Inference Overview**: By utilizing spectral clustering on a kNN graph, non-overlapping patches of similar size with geometric coherence are generated before being independently refined by the model. The refined patches can be combined by simple patch concatenation.

5.2.2 Refinement

The generated patches are subsequently refined individually before being aggregated to form the complete, enhanced shape. As the patches are non-overlapping, a more advanced patch fusion methodology is not necessary. This integration process, pivotal for the reconstruction, is depicted in Figure 4.

5.3. Quantitative Results

To quantitatively evaluate our results, we utilize both Chamfer Distance and Earth Mover's Distance. Following the methodology proposed in PointFlow, we employ 1-nearest neighbor accuracy (1-NNA) to assess the similarity of two distributions. If two point clouds have the same original distribution the metric will evaluate to approximately 50%. [10]

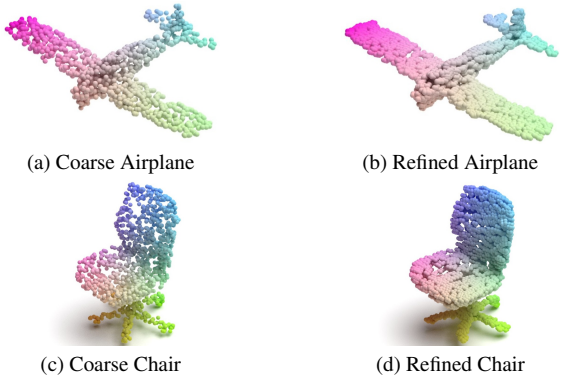
With distribution of point clouds based off ShapeNet as ground truth, both metrics are computed on $N = 400$ point clouds sampled from PVD as well as the same point clouds refined by our model (PVD + ours) and shown in Table 1.

5.4. Qualitative Results

Our qualitative evaluation revealed instances of both highly successful refinements (see Figure 5) and notable failures (see Figure 9), underscoring the variability in our model's performance This variability can be attributed to the quality of the input provided to our model. Specifically, when the shape output from the PVD model is excessively coarse, it undermines our refinement process. This issue arises from a fundamental assumption in our approach: the output distribution of the PVD model can be approximated by sub-sampling and noising ShapeNet point clouds.

	Airplane		Chair	
Networks	CD	EMD	CD	EMD
PVD	**75.49%**	**68.51%**	58.37%	**56.01%**
PVD + Ours	79.87%	73.37%	**56.49%**	58.87%

Table 1. Performance Comparison on Shape Retrieval. We use 1-NNA as an evaluation metric. CD and EMD are used to measure the distance as done in PVD [14]. A score closer to 50% indicates better performance. As mentioned in [14] the EMD score might varry between implementation. We use the implementation provided in PVD [14] to enforce consistent results. The results show that our approach managed to generate chairs of a distribution more closely resembling ShapeNet than the raw PVD output, according to Chamfer Distance, however did not quantitatively improve the results for other metrics.



(a) Coarse Airplane (b) Refined Airplane

(c) Coarse Chair (d) Refined Chair

Figure 5. Examples of successfully refined shapes, demonstrating the model's ability to enhance geometric detail in well-defined inputs with less noise.

6. Conclusion

In conclusion, we introduce a novel architecture for refining coarse point clouds derived from 3D shapes, leveraging advancements in Point-Voxel Convolution, Point Transformer, and PU-Net technologies through a patch-by-patch approach. Our model systematically enhances geometric detail and realism in 3D models, demonstrating significant visual improvements in shape quality. Future research directions include exploring global refinement layers that combine patches and can potentially further improve model performance.

References

[1] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge J. Belongie, Noah Snavely, and Bharath Hariharan. Learning gradient fields for shape generation. *CoRR*, abs/2008.06520, 2020. 1

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3

[3] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. *CoRR*, abs/1612.00603, 2016. 3

[4] Hyeongju Kim, Hyeonseung Lee, Woo Hyun Kang, Joun Yeop Lee, and Nam Soo Kim. Softflow: Probabilistic framework for normalizing flow on manifolds. *CoRR*, abs/2006.04604, 2020. 1

[5] Roman Klokov, Edmond Boyer, and Jakob Verbeek. Discrete point flow networks for efficient point cloud generation. *CoRR*, abs/2007.10170, 2020. 1

[6] Ruihui Li, Xianzhi Li, Pheng-Ann Heng, and Chi-Wing Fu. Point cloud upsampling via disentangled refinement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 344–353, 2021. 1

[7] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel CNN for efficient 3d deep learning. *CoRR*, abs/1907.03739, 2019. 1, 2, 6

[8] Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3d point cloud completion. *arXiv preprint arXiv:2112.03530*, 2021. 1

[9] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017. 2

[10] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge J. Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. *CoRR*, abs/1906.12320, 2019. 1, 3, 4

[11] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. *CoRR*, abs/1801.06761, 2018. 1, 2, 3, 6

[12] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*, 2022. 2

[13] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point transformer. *CoRR*, abs/2012.09164, 2020. 1, 6

[14] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. *CoRR*, abs/2104.03670, 2021. 1, 2, 3, 4

A. Figures of used architectures.

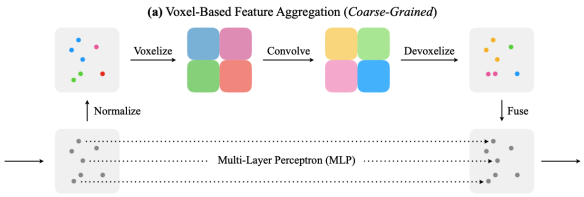


Figure 6. Point-Voxel Convolution Layer. Image credits: Zhijian Liu et al. [7]

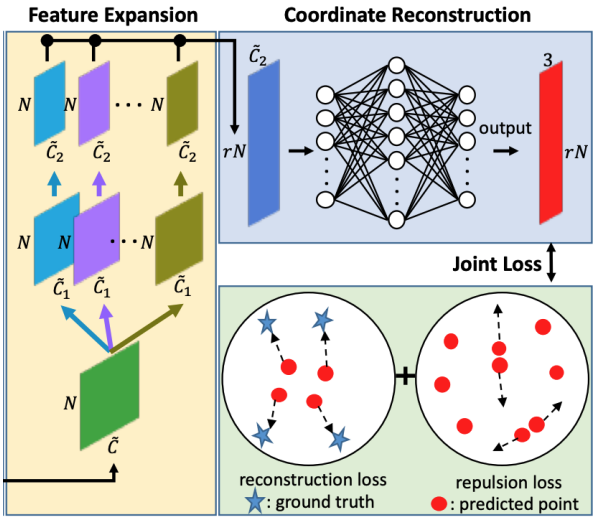


Figure 7. Feature Expansion, Coordinate Reconstruction and Joint Loss used in PU-Net. Image credits: Lequan Yu et al. [11]

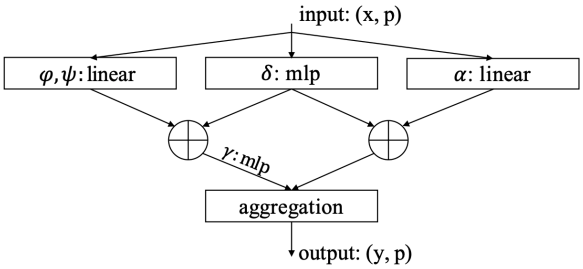
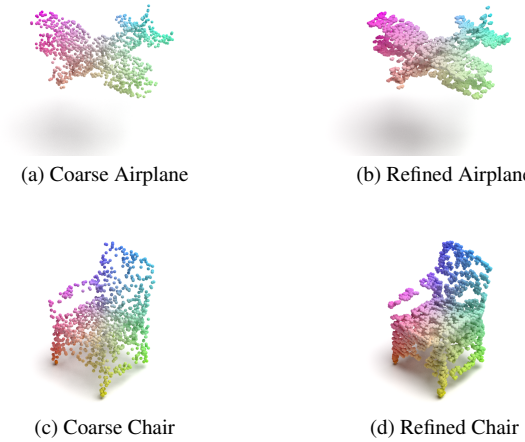


Figure 8. Point attention. Image credits: Hengshuang Zhao et al. [13]

B. Visual results



(a) Coarse Airplane (b) Refined Airplane

(c) Coarse Chair (d) Refined Chair

Figure 9. Illustrations of sub-optimal shape refinements due to sparse inputs with insufficient geometric detail and noise, impacting model performance.