TopNet: Structural Point Cloud Decoder

Supplementary Material

Lyne P. Tchapmi[1] Vineet Kosaraju[1] S. Hamid Rezatofighi[1,2] Ian Reid[2] Silvio Savarese[1]

[1]Stanford University, [2]The University of Adelaide, Australia

This document provides further analysis for the method presented in our main paper.

1. Design analysis

In section 4 of our main paper, we demonstrated that any topology $\mathbb{T}$ on a set $\mathcal{S}$ can be embedded in a rooted tree in which each node of the tree represents a non-empty subset of $\mathbb{T}$ or a singleton of $\mathcal{S}$. We proved this hypothesis for any arbitrary rooted tree. However, the decoder D which we propose here, is a specific rooted tree structure with the following features:

- Every node in D has at most one parent
- All nodes at the same level have the same number of children
- All leaves are at the same level.

While G is not guaranteed to meet the conditions above, it can be transformed into a new equivalent tree G' for which the conditions above are met by performing a series of node duplication on G outlined below and illustrated in Fig 1:

- Let x be a node in G with $k > 1$ parents and $E(p_i, x), i \in 1 \dots k$ be the corresponding edges from parent p_i to node x. We duplicate x and it's associated subtree and create k equivalent new nodes $x_1, \dots x_k$ and edges $E(p_i, x_i), i \in 1 \dots k$. The result of this duplication is new tree G' where the new nodes x_i have a unique parent p_i (Fig. 1a).
- If a leaf x is at level $i < L$ where L is the highest level of a leaf in G, duplicate x and add the duplicate as a new child of x. Repeat this procedure until the desired depth is reached (Fig..1b).
- Let K be the maximum number of children for a node at level i. For a node x at level i having less than K children, duplicate one of the children of x to reach K children and create edges from x to the duplicated children (Figure 1c, 1d).

Since the operations above are node duplications they do not delete or add to the information contained in the original tree representing topology $\mathbb{T}$. Therefore the new tree G also embeds topology $\mathbb{T}$. The duplication operations suggest that our learned decoder may generate duplicate point set embeddings at different nodes. This does not affect the final point set generated since point sets are invariant to duplication of set elements. Still, it would be interesting to explore methods to reduce potential duplicate nodes in the learning process. We leave this as an open research question.

2. Ablation studies

Design choices involved in our decoder include choosing the number of features F generated for each node embedding and the number of tree levels L. We analyze the effect of these parameters for an output cloud size $N = 2048$ by varying F in $\{8, 16, 32, 64\}$, and L in $\{2, 4, 6, 8\}$. This ablations study was used to pick the model's final number of layers and number of features. One important thing to note is that since the number of output points is fixed at 2048 in this experiment, increasing the number of levels requires decreasing the number of children per level. This operation is therefore not similar to adding a new layer in conventional networks and a deeper tree may not necessarily improve performance.

In Figure 2a, we plot the Chamfer distance as a function of the number of levels L for different values of F. For $F \in \{8, 32, 64\}$ the graphs exhibit different local minima but for $F = 6$, the performance oscillates around an average value. In Figure 2b, we plot the Chamfer distance as a function of the number of features F for different values of L). The graphs for $L \in \{4, 6, 8\}$ exhibit slightly similar trend though the pattern is non-convex. The graph for $L = 2$ exhibits a very different pattern compare to the others. In all experiments, regardless of the value for L and F above, our method outperforms all previous method.

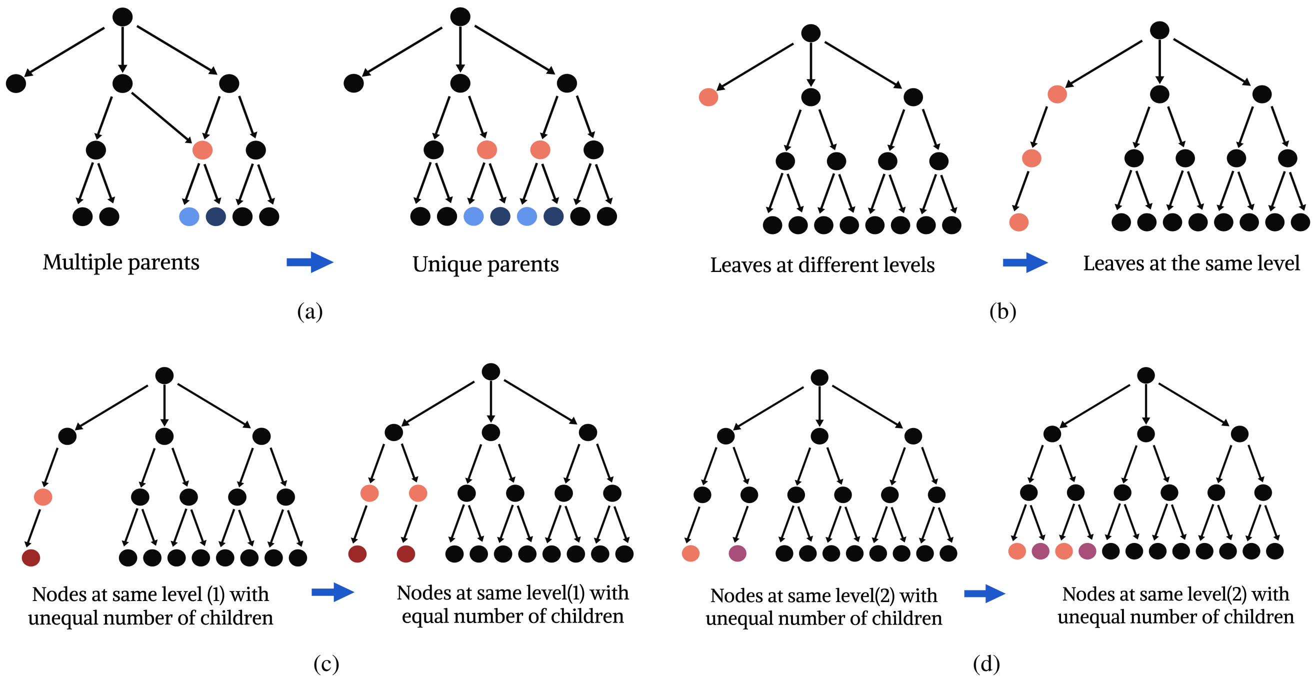
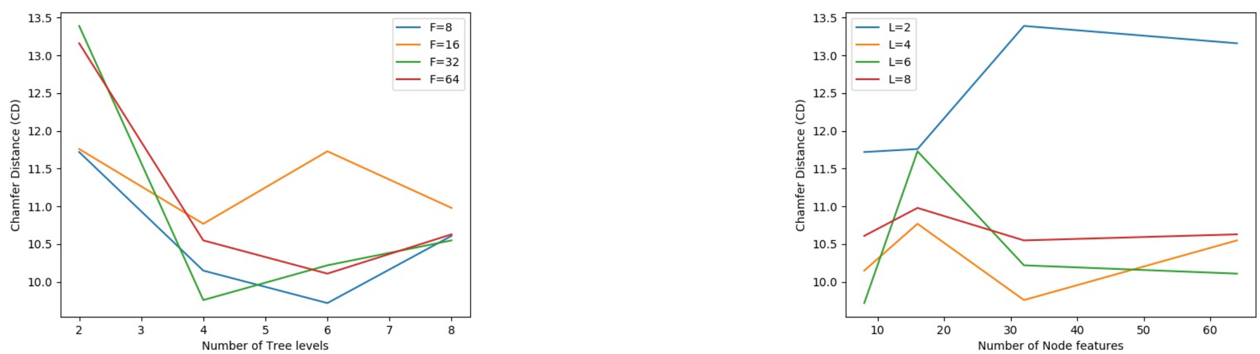


Figure 1: Illustration of duplication operation that can be performed on any rooted tree to transform it into a full tree



(a) (b)

Figure 2: **Ablation Experiments:** We analyze the effect of varying different parameters in our network. We vary the number of node features $\{8, 16, 32, 64\}$, and the number of tree levels $\{2, 4, 6, 8\}$, while keeping the number of outputs constants. For instance when the number of levels L=2, the number of children per level is 32-64. When L=4, the number of children per level is 4, 4, 4, 8. All instantiations of our method outperform previous works. The number of levels seem to suggest a local minimum, but the number of features does not show a noticeable pattern. The Chamfer distance is reported multiplied by 10^4.

3. Visualizing learned structure

By design, each node in our decoder embeds and generates a subset of S made of all its descendant leaves. We can roughly visualize learned structure by plotting each node's descendant leaves. In Figure 3 we show visualizations of several nodes of the decoder. We notice that several clustering patterns emerge. Some clusterings seem geometric (edges of plane and table, center of table), others semantic (legs of table, front and back of car) while others appear random but most are consistent across similar objects. This can be seen as a consequence of our adoption of the more general definition of topology which does not enforce the generated clustering to be smooth.

4. Note on previous works comparison

There exists a large number of works on point cloud generation for 3D shape completion and 3D reconstruction.

We chose a select number of recent works for comparison based on code availability and replicability. Some works such as [1] were not included in our comparison due to issues such as dataset unavailability and convergence issues on our dataset. Due to the lack of consistency in previous works evaluation (different dataset splits, optimizers, number of epochs, loss functions, etc.) we will be releasing a public large scale point cloud completion benchmark based on the setup and dataset presented in this paper to allow for consistent comparison across different works.

References

[1] M. Gadelha, R. Wang, and S. Maji. Multiresolution tree networks for 3d point cloud processing. In *The European Conference on Computer Vision (ECCV)*, September 2018. 3

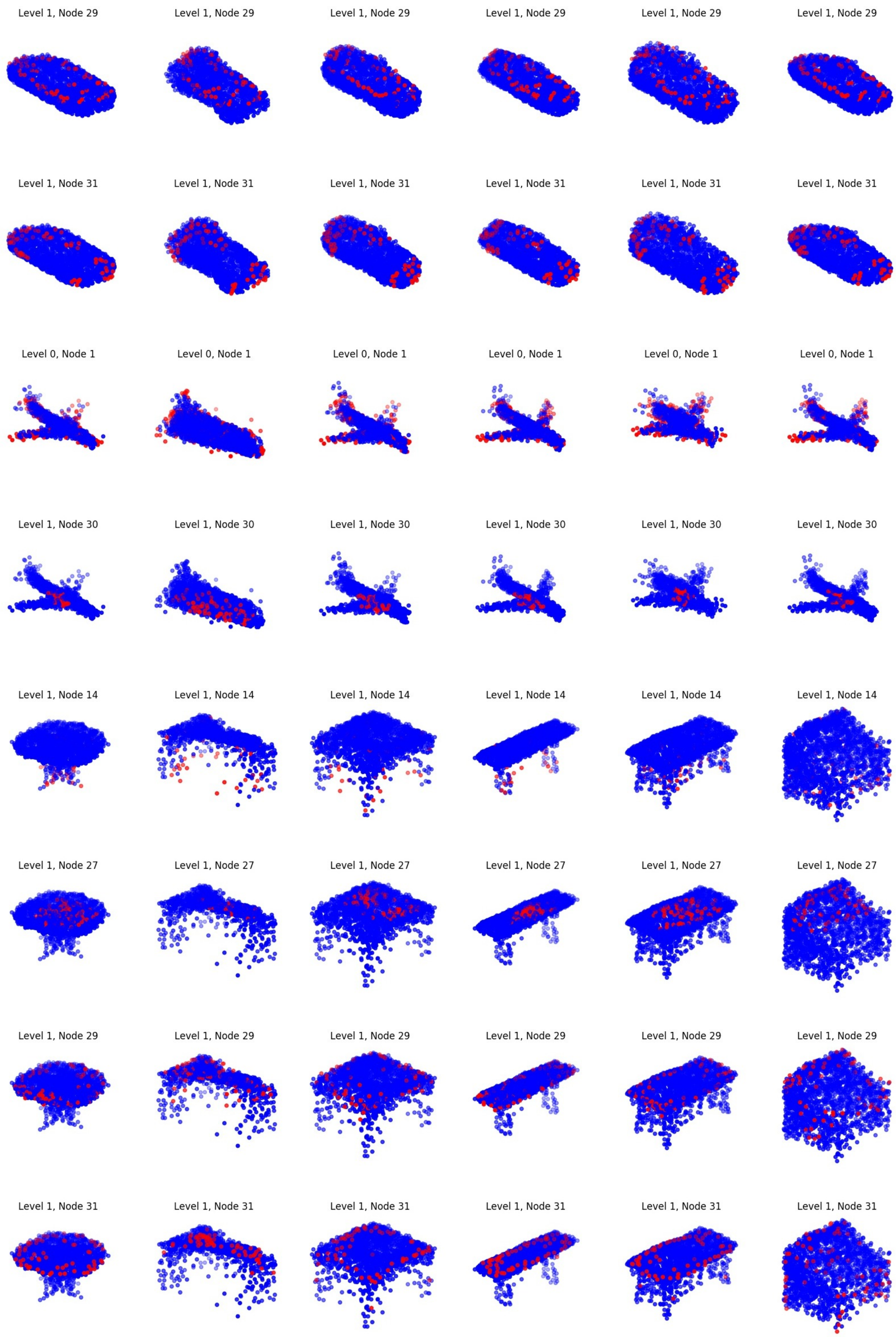


Figure 3: Selected point grouping patterns emerging from our structural decoder.