

DEEP-LEARNING-BASED POINT CLOUD UPSAMPLING OF NATURAL ENTITIES AND SCENES

Tian Zhang*, Sagi Filin

Mapping and Geo-Information Engineering, Technion – Israel Institute of Technology, Haifa, Israel
(tianz, filin)@technion.ac.il

Commission II, WG 3

KEY WORDS: Geometric deep learning, Point cloud upsampling, Graph convolutional networks, Natural scenes.

ABSTRACT:

Limited accessibility, occlusions, or sensor placement, can generate unevenly sampled laser scanning based point clouds. Such uneven coverage and partial lack of detail can affect the computation of geometric features therein and generate a visually unpleasant site description. The application of 3-D interpolation-driven solutions has been demonstrated to generate oversmoothed results as such algorithms ignore local patterns and variations within the surface. In that respect, the introduction of deep neural networks (DNN) has the potential to learn more complicated forms, typical of the rich morphological patterns that natural landforms and entities therein tend to exhibit. While existing research has focused on the upsampling of man-made objects, little has been devoted to natural scenes and the entities therein. To address that, we propose in this paper a DNN based approach that utilizes the self-similarity of geometric details as a means to address this generally ill-posed problem. Specifically, we treat two key elements that stand at the root of point-DNN-related design, the definition and selection of neighboring points, and the interpolation at a high dimensional feature space. We show how the introduction of a graph convolutional network and an attention unit helps address these matters and demonstrate how knowledge of densely sampled regions can be learned and transferred to sparsely sampled ones through geometric learning methods.

1. INTRODUCTION

Because of topographic variations, occlusion, or use of wide baselines, point cloud depiction of natural scenes is oftentimes unevenly sampled. This may leave regions in the site partially covered or even void of data. Such coverage-related problems are not only visually unpleasant but as the literature show hamper the modeling and interpretation of the scene (Boulton and Stokes, 2018; Wang et al., 2020; Metzger et al., 2021; Singh et al., 2021; Yan, 2021). One solution that can avert artifacts caused by partial coverage is to upsample the point cloud to yield a more even distribution throughout the site. In natural scenes, upsampling poses a unique challenge because of the complex morphology and the varying scales. This suggests that the generated points should describe the underlying geometry of a latent target object at multiple scales while conforming to both local and global trends. The common solution has been applying 3-D interpolation-based techniques (Lipman et al., 2007; Huang et al., 2013), where new points were planted in void parts as a natural extension of the observed surface. However, such approaches are suited for structured shapes and tend to generate an overly smoothed outcome (Wang et al., 2020).

To address these challenges, we propose in this paper a data-driven deep neural network (DNN) upsampling framework. Our aim is to utilize the self-similarity of geometric details in the point cloud as a means to address this generally ill-posed problem. Our assumption is that knowledge of densely sampled regions can be learned and transferred to sparsely sampled parts. While most approaches tackle this interpolation problem through a simple feature expansion or by progressive upsampling (Yu et al., 2018b; Yifan et al., 2019), we place great focus on the local contextual information when planting new

points. We also demonstrate how the introduction of an attention unit helps to decorrelate the enriched features, thereby allowing an informative and unclustered upsampled outcome. Finally, we consider a better-suited cost function to avoid the need to describe the correspondence between prediction and ground-truth, thereby providing a simple and efficient evaluation process. Our results demonstrate better performance in challenging cases with lower error measures compared to the ground-truth data on both object and complete-site levels.

2. RELATED WORK

Early upsampling-related research has focused on the development of 3-D interpolation-driven methods. As an example, Fleishman et al. (2003) upsampled a point set by interpolating points at vertices of a Voronoi diagram in the local tangent space. Lipman et al. (2007) presented a locally optimal projection (LOP) operator for points resampling and surface reconstruction based on the L_1 median. This operator performs well even when the input point set contained noise and outliers. Successively, Huang et al. (2009, 2013) proposed an improved weighted LOP and its variant to edge-aware cases as a means to address the point set density problem while preserving sharp features. Although these models have yielded good results, they made a strong assumption about the smoothness of the underlying surface, thus restricting their scope.

Deep learning methods now achieve state-of-the-art results in point cloud upsampling. Rather than interpolating in a Euclidean space, the network learns to plant new points using high-dimensional features from the training data. Neural point processing was pioneered by the PointNet and PointNet++ networks (Qi et al., 2017a,b), where the problems of irregularity and the lack of structure were addressed by applying shared

* Corresponding author

multilayer perceptrons (MLPs) for the feature transformation of individual points, and a symmetric function, e.g., max pooling, for global feature extraction. Yu et al. (2018b) introduced the first point set upsampling network, PU-Net, where both the input and the output were the 3-D coordinates of a point set. PU-Net extracted multiscale features based on PointNet++ and concatenated them to obtain aggregated multi-scale features for each input point. These features were expanded by replication, then transformed to a uniformly distributed upsampled pointset of the underlying surface. Although multiscale features were gathered, the correspondence between points and their feature similarity was neglected. Therefore, the network suffered the loss of local sharp features. In a later work, Yu et al. (2018a) proposed the EC-Net, an edge-aware network for pointset consolidation. Because of the PU-Net tendency to penalize the accumulation of points, sharp transitions were smoothed. To encourage the preservation of edges, an edge-aware joint loss was introduced. Nonetheless, the EC-Net is very complicated in the data and training preparation phases. Yifan et al. (2019) proposed the 3PU, a progressive network that learns different levels of detail in multiple steps, where each step focuses on a local patch from the output of the previous step. Due to its progressive nature, the 3PU network is computationally expensive and requires more data to supervise the middle-stage outputs of the network. Li et al. (2019) proposed the PU-GAN, a generative adversarial network (GAN) designed to learn upsampled point distributions. Inspired by the advances in the attention mechanism in recurrent neural networks, the PU-GAN adopts a self-attention module to the upsampling by introducing the interactions of low-level and high-level features. Recently, Qian et al. (2021) proposed the PU-GCN, a graph convolutional network (GCN), where multi-level features are aggregated by an inception-like block. The advantage of the GCN is in the reduction of parameters to learn, making it also more computation efficient compared to the PointNet++ architecture. The PU-GCN approach performs well when handling samples generated from watertight meshes. Nonetheless, when handling complex shapes with open boundaries or natural complex-shaped objects, this network tends to fail (Zhou et al., 2021).

Most upsampling frameworks adopt the PointNet++ architecture as the front-end to learn high-dimensional features. Nonetheless, such a framework is limited in its representative power due to the utilization of a static graph to exploit local geometric structures. Generally, the upsampling is performed through a simple feature expansion by replication (Yu et al., 2018b; Qian et al., 2021) and tends to create a clustered outcome. In this paper, we demonstrate how the alteration of this base PointNet++ related concept helps resolve this matter. We propose a neighborhood definition that effectively encodes spatial information and allows for effective feature extraction. We also demonstrate how alteration of the feature interpolation concept allows resolving the clustering of points and generating a truer to reality upsampling outcome. Such a concept places great focus on the surrounding entities when planting new points, thereby adhering better to the underlying structure.

3. METHODOLOGY

Our network is based on the architecture proposed by Yu et al. (2018b) that also sets the base for many upsampling applications (e.g., Yu et al., 2018a; Yifan et al., 2019; Li et al., 2019, 2021). These neural upsampling applications consist of a feature extraction component and a feature-space interpolation (expansion) component. The first maps the points from a

Euclidean space to a high dimensional space as a means to capture intrinsic properties of the local geometric structure. The second performs interpolation at this high-dimensional space and then maps the interpolated features back to the Euclidean space. Shared drawbacks of these applications are the over-smoothness around edges and the tendency to generate clutter around layered surfaces and complex structures (cf. Fig. 4). The causes for such phenomena are rooted in the manner by which features are extracted by their networks as well as in the interpolation process. As Sec. (2) noted, the neighborhood definition for exploiting local geometric structures is driven by Euclidean distance measures. Therefore, it limits the effective receptive field to a predefined value assigned to the network. Secondly, a simple interpolation through aggregation or duplication of features does not necessarily enforce the interactions of features and is limited in differentiating the contributions of local surfaces and global trends when planting a new set of points.

Our approach addresses these challenges by the following modifications to the base architecture (Fig. 1). Firstly, we extend the receptive field to the entire point set by modifying the common shard MLPs in the feature extraction component by introducing a dynamic graph convolution unit. Secondly, we weigh the contribution of local and global features when performing interpolations, thereby allowing us to generate a local structure-attentive outcome while being coherent to the global trend. As noted, we also modify the cost function to avoid the need to describe the correspondence between predicted and ground-truth points, thereby making the evaluation more efficient.

3.1 Feature extraction

The use of the PointNet++ for feature extraction produces a static graph that is based on Euclidean distance measures as a means to drive neighborhood relations. Such a local neighborhood is not guaranteed to define the local surface structure, as it does not necessarily imply geodesic proximity. As noted by Wang et al. (2019), such a setting also has a limited receptive field, bounded by the largest querying distance to organize neighbors per point.

Instead, we consider a dynamic graph approach to query for neighbors (Fig. 1). The proximity term is determined initially in the first layer by Euclidean distance, while in the following ones it is derived by the closeness of the learned features (closeness in the sense of intrinsic properties). This approach, known as dynamic graph convolution (aka graph convolution network, GCN, Wang et al., 2019), has a receptive field that covers an entire graph. Given a point p_i and its neighbor p_j , the convolution is defined by:

$$e_{ijm} = \text{Relu}(\theta_m \cdot (p_j - p_i) + \phi_m \cdot p_i) \quad (1)$$

where θ_m and ϕ_m are MLPs, and the final activation x_{im} of p_i at the m -th layer is:

$$x_{im} = \max_{j \in N_i} e_{ijm} \quad (2)$$

where N_i is a set of neighboring points for the i -th point. Note that the max function is permutation invariant. Therefore, it is not related to the order of points. We denote this using as Per-point GCN Feature Extraction Unit (Fig. 1).

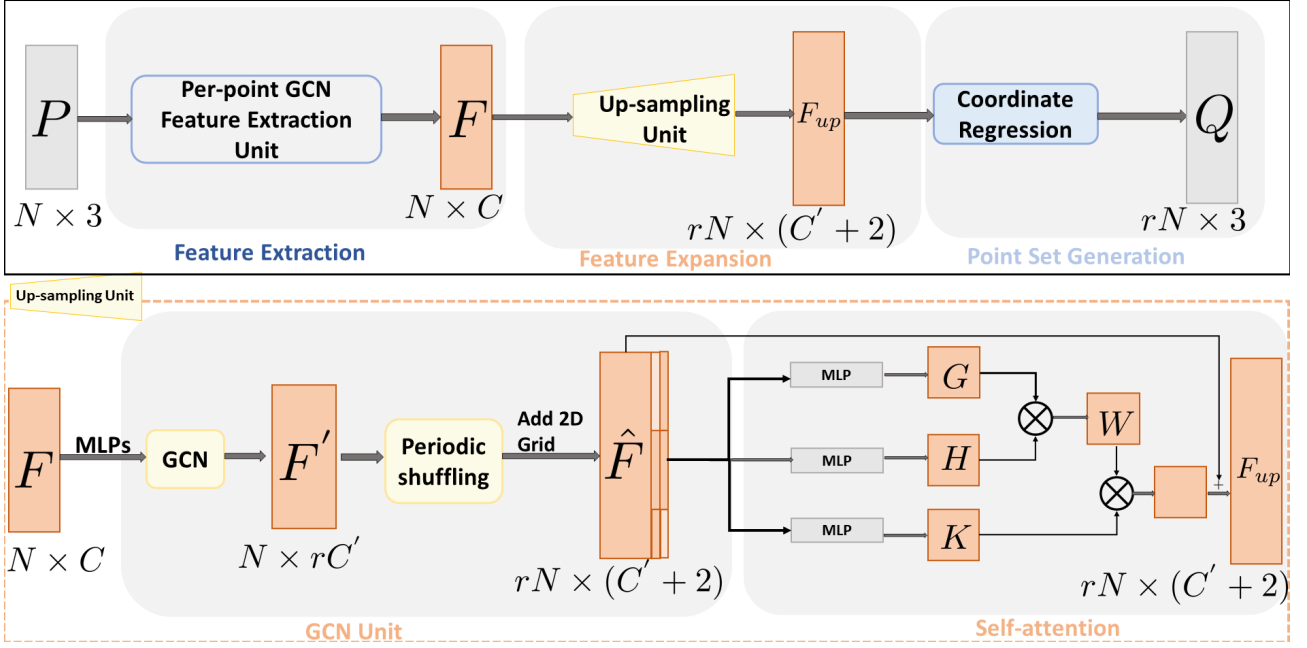


Figure 1. Overview of our point cloud upsampling framework.

3.2 Context-aware feature expansion

Earlier studies concatenated features extracted from each layer along feature dimensions and the interpolation was performed by a set of shared MLPs (Yu et al., 2018b; Yifan et al., 2019). This strategy creates a highly correlated feature representation that tends to cluster points in the final prediction. This correlation was partially resolved by utilizing a non-linear activation function after each MLP.

Here, given a high dimensional feature, F , we aim to expand it r times. This expansion is being made by two components, first, a GCN unit to expand the features to $N \times rC'$, and second a periodic shuffling unit to rearrange the output. In this manner, the expansion is no longer an operation per point but considers the spatial information in the latent space. The periodic shuffling (Qian et al., 2021), a matrix reshuffling operator, is applied here as a means to rearrange features, particularly for the convenience of regressing them back to an Euclidean space in the subsequent phase. Hence, the dimensions of our features change from $N \times rC'$ to $rN \times C'$. From the reshuffled features, we added a 2-D grid to encode the position of each feature point as input $\hat{F}$ for the self-attention unit.

We consider a solution that can effectively embed the neighborhood information in a high-dimensional space when planting new points. The proposed method seeks a mechanism to better incorporate contextual information, similar to the attention concept in image captioning and recurrent neural networks (Vaswani et al., 2017). For that, we introduce a self-attention unit (Li et al., 2019) that effectively learns the contribution or weights of each feature vector when generating outputs. This setting is justified as it encourages incorporating contextual information by looking at the entire point cloud and evaluating the importance per point for the prediction. Our attention includes three components, query sets G , key sets H , and values K , where:

$$G = \hat{F}W_G \quad H = \hat{F}W_H \quad (3)$$

where W_G and W_H are weights of the MLPs. Note here the

dimensions of G and H are the same. We derive the score W , of dimension $rN \times rN$, from the alignment of G and H by applying the softmax function, which provides the weights for each feature vector in K . Here, the output dimension of K is $rN \times (C' + 2)$. Note here the three components G, H, K are derived from $\hat{F}$, hence the self-attention definition. In addition, to avoid over-fitting we also add a skip connection from $\hat{F}$ to F_{up} .

3.3 Regression

We adopt a common coordinate regression unit, similar to Yu et al. (2018b); Li et al. (2019); Qian et al. (2021), which consists of several MLPs. The coordinate regression unit maps the features to our final prediction outcome Q .

3.4 Cost Function

The cost function to evaluate the quality of the network prediction would usually measure the closeness of the newly established point to the ground truth P . The common choice is Earth mover distance (EMD, Yu et al., 2018b; Li et al., 2019). Such distance needs to define a bijection mapping from Q to P , $\beta(Q) : Q \rightarrow P$. Differing from the EMD, we introduce a metric that is easier to perform and less rigid. To position points on the underlying object surfaces, we aim to obtain a predicted set Q from the sparse samples that are close to ground truth P by minimizing the Chamfer distance (Qian et al., 2021):

$$L(P, Q) = \frac{1}{|P|} \sum_{p \in P} \min_{q \in Q} \|p - q\|_2^2 + \frac{1}{|Q|} \sum_{q \in Q} \min_{p \in P} \|p - q\|_2^2 \quad (4)$$

This distance is differentiable and also invariant to point order, which avoids the need to define correspondence. In the training time, the network evaluates this loss by measuring the similarity of the predicted outcome and manually cropped ground truth. The weight parameters in each layer are updated using gradient descent through back-propagation.

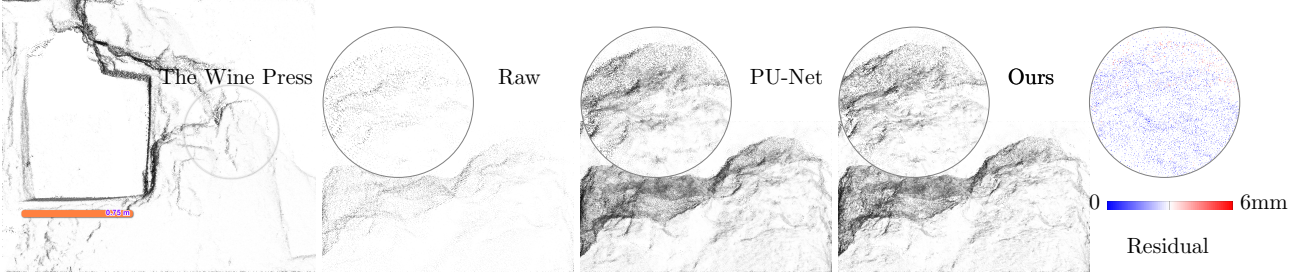


Figure 2. A typical terrain in our dataset, upsampled by Yu et al. (2018b) and our approach. Points are shaded by their normal orientations. Our approach provides an outcome of richer features coherent to the original scene (see the residual plot).

We introduce a dynamic weight decaying scheduler to encourage weight updating during training. At inference time, the network is fed with spare samples and uses its output as a final, predicted, form.

4. RESULTS

Implementation details – We trained the proposed upsampling network for 100 epochs on one GPU (NVIDIA TITAN 2080Ti) in all experiments. We optimized the network parameters using the Adam optimizer with a learning rate of 0.001 and a beta of 0.9. This selection is justified by the fact that momentum introduces a better convergence given complex loss surfaces. As a training set we use the PU1K dataset, published by Qian et al. (2021), which consists of 1020 training samples for all models. Following Qian et al. (2021) convention, we cropped 50 patches from each 3-D model (51,000 patches in total) as the input texture to the network. By using textures from another distribution, different from that of natural scenes, our aim was to test the generalizability of our network and also to establish a fair comparison with common architectures. The upsampling ratio, r , was set to 4 in all experiments. We compared our approach to the recent models by Yu et al. (2018b) and Yifan et al. (2019). We used the implementation by Qian et al. (2021) for these models, available in the public domain.

Upsampling application on natural scenes We demonstrate the application of our network on two datasets. One is a scan of a tree, and the other is of an ancient wine-press site. The tree scan features a non-linear complex geometric structure with many discontinuities and shape variations. The rock-quarried ancient wine press, prevalent across the Mediterranean basin, particularly during the late Roman and Byzantine times (Stavi et al., 2018), provides an extended rock exposure, forming a seemingly large-area drainage basin, which – upon rain events – drained into the vat. This site is of great complexity, featuring an abundance of elevation changes (up to 3 meters) and voids in the point cloud, due to occlusions and a relatively wide baseline between posts (cf. Fig. 2 and Fig. 6). Both datasets were acquired by a Leica C10 scanner with both low and high-resolution settings, the wine press consists of five scans and consists of approx. 30 million points.

We first study the performance of our proposed model in a region featuring sharp transitions within the wine press site. Of focus (Fig. 2) is one blindspot area of one of the scans, where we used points acquired from others, and evaluated how the upsampling performs against a lower resolution version of the data. We measured these differences to the ground truth using the distance of each predicted point to the closest ground truth. To quantify the upsampling results we computed the nearest

neighbor distance between the upsampled version of the point cloud to the ground truth. Results show that we successfully upsample a sparse set of input points to a dense one exhibiting rich geometric details, similar to the rock face morphology and well embedded within it. Analysis of the residuals of our interpolated set of points compared to the original scan shows that they are mostly below 5 millimeters. A comparison of our results to that of the PU-Net (Fig. 2) shows also how finer details are highlighted in a much sharper manner.

Generating an even distribution of point density within the scene is critical to providing a visually pleasing outcome. We demonstrate the application of our model in contributing to obtaining such an outcome. Given a region of interest, the blindspot of one individual scan, covered by a much sparser point set from neighboring scans, we clustered the original data into two subsets according to their point density (lower density within the blindspot vs. higher outside) and performed the upsampling task on the low-resolution section. Though more advanced partitioning approaches for the point set may exist, here we use this simple strategy as a proof of concept. As demonstrated in Fig. 3, our approach distributes points within that region in a more uniform pattern compared to using the PU-Net model.

Turning to the tree scan, the value of the proposed GCN network becomes apparent. As noted, the GCN is better designed for embedding spatial information and is therefore expected to improve the quality of the final prediction, especially in the mixture of entities. In Fig. (4) a visual demonstration of the upsampling performance is provided, comparing also our approach to the PU-Net and the MPU networks. Here the differences to the ground truth are computed again and analyzed as a histogram with 5mm spacing. While the PU-Net and MPU blurred the branches and bifurcation points, our model was capable of producing a high-fidelity outcome (cf. Fig. 4 & Fig. 5). The histogram plotting of the differences (Fig. 5) demonstrates how our results are more accurate with approx. 70% of the data having differences lower than 5 mm and approx. 90% with a sub-centimeter level of accuracy.

An application of our upsampling solution at a site level is presented in Fig. 6. Our model is capable to bridge the voids in the five blind spots with the majority of the differences lower than 3 mm. Note our model is designed for upsampling rather than shape completion. Therefore, at the bottom of the vat, where no data exists at all, it did not complete the large void region, as it was not aimed to. For quantitative evaluation of the wine-press data, we used the low-resolution version of the scan as input and compared the outcome with the high-resolution set. Following the convention in previous upsampling architectures, we used the Chamfer distance (CD) and Hausdorff distance (HD). Here, the selection of HD is utilized for a fair

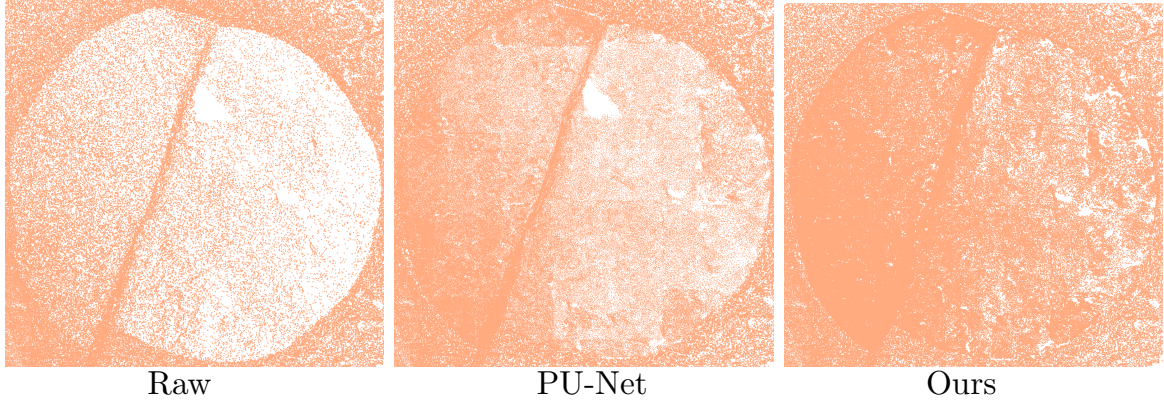


Figure 3. Comparison of upsampling low resolution data, with outcome generated by the PU-Net and our network. Our approach helps to generate a more uniformly distributed pointset.

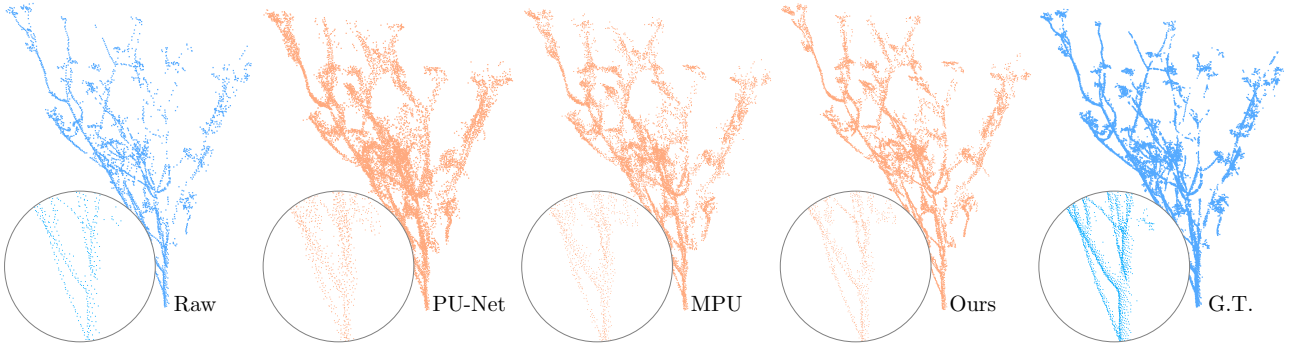


Figure 4. Upsampling of low resolution tree pointcloud. Outcome generated by PU-Net, MPU, and Ours and evaluated against ground truth (G.T.) data. Our approach produces sharper edges and richer features consistent with the original high density point cloud.

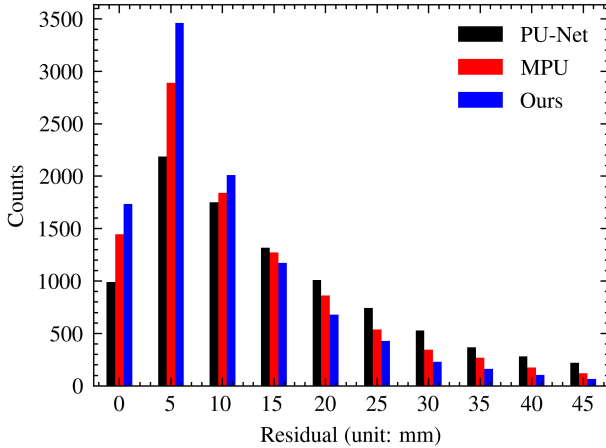


Figure 5. Evaluation of residuals of the tree model upsampled by the PU-Net, MPU and our model, while compared to ground truth.

evaluation and comparison as the model was trained by the CD. The CD measures the average of all the distances from a point in one set to the closest point in the other set, while the HD measures the largest of all distances. Both metrics measure the closeness of two clusters in the same metric space, smaller is better. We divided the whole dataset into 512 patches (each with 8192 points) and analyzed the average CD and HD of all patches. As shown in Table (1), compared to PU-Net and MPU, our proposed model yields the smallest CD and HD on the wine

press dataset. Note though being trained on the dataset from other domains, our model is capable to generalize the learned knowledge and leading to the performance it exhibited in the upsampling of both the wine press and tree models.

Network	PU-Net	MPU	Ours
CD 10^{-3}	3.083	2.124	1.995
HD 10^{-3}	50.368	40.189	37.427

Table 1. Quantitative comparison of our proposed model vs. the PU-Net and MPU on the wine press dataset. Bold-face text denotes the best performance.

5. CONCLUSIONS

Neural upsampling approaches provide a promising avenue to address the defects and partial coverage in point clouds depicting natural objects and scenes. In that respect, this paper proposed an architecture by which we compensated for the lack of coverage and recovered detail directly from scans. Recognizing the deficiency of the common applied PointNet++ based architecture, one objective was to address the insufficient receptive field for feature computation and the loss of detail around a mixture of entities. Recognizing that feature proximity represents the similarity of entities in upsampling, we first constructed a dynamic graph rather than an Euclidean graph for efficient feature extraction and then expanded these features to perform upsampling guided by a self-attention mechanism. In this manner, the dynamic graph allowed an incremental grouping of similar entities and an extension of the receptive field to

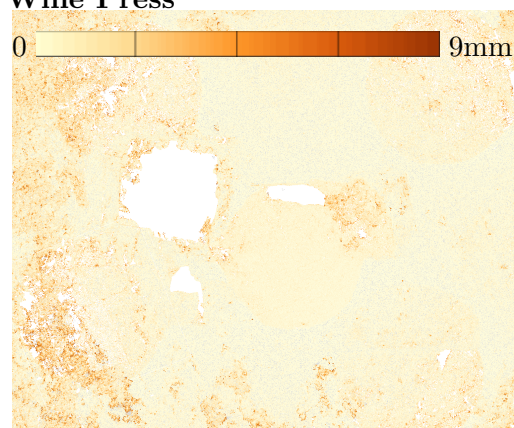
the entire graph. This way the learned features captured both local and global trends. These enriched features were evaluated by their contribution to each point in the final prediction. Therefore, new points were planted under the guidance of contextual information. Results demonstrated that the contribution of the proposed approach in improving the density distribution at voids and recovering details conformed to the ground truth. This strategy may help in further applications in attribute estimation or visualization-related applications.

REFERENCES

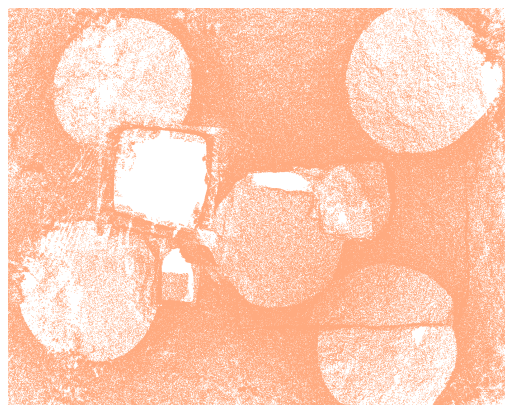
- Boulton, S. J., Stokes, M., 2018. Which DEM is best for analyzing fluvial landscape development in mountainous terrains? *Geomorphology*, 310, 168–187.
- Fleishman, S., Cohen-Or, D., Alexa, M., Silva, C. T., 2003. Progressive point set surfaces. *ACM Transactions on Graphics (TOG)*, 22(4), 997–1011.
- Huang, H., Li, D., Zhang, H., Ascher, U., Cohen-Or, D., 2009. Consolidation of unorganized point clouds for surface reconstruction. *ACM transactions on graphics (TOG)*, 28(5), 1–7.
- Huang, H., Wu, S., Gong, M., Cohen-Or, D., Ascher, U., Zhang, H., 2013. Edge-aware point set resampling. *ACM transactions on graphics (TOG)*, 32(1), 1–12.
- Li, R., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2019. Pungan: a point cloud upsampling adversarial network. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7203–7212.
- Li, R., Li, X., Heng, P.-A., Fu, C.-W., 2021. Point cloud upsampling via disentangled refinement. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 344–353.
- Lipman, Y., Cohen-Or, D., Levin, D., Tal-Ezer, H., 2007. Parameterization-free projection for geometry reconstruction. *ACM Transactions on Graphics (TOG)*, 26(3), 22–es.
- Metzer, G., Hanocka, R., Giryas, R., Cohen-Or, D., 2021. Self-sampling for neural point cloud consolidation. *ACM Transactions on Graphics (TOG)*, 40(5), 1–14.
- Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.
- Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*.
- Qian, G., Abualshour, A., Li, G., Thabet, A., Ghanem, B., 2021. Pu-gcn: Point cloud upsampling using graph convolutional networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11683–11692.
- Singh, S. K., Raval, S., Banerjee, B., 2021. A robust approach to identify roof bolts in 3D point cloud data captured from a mobile laser scanner. *International Journal of Mining Science and Technology*, 31(2), 303–312.
- Stavi, I., Chocron, M., Filin, S., Arav, R., Ackermann, O., Zissu, B., 2018. Intentional, dual purpose of ancient wine presses as cisterns for runoff water harvesting in drylands. *The Holocene*, 28(7), 1107–1112.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Wang, Y., Li, H., Liao, L., 2020. DEM Construction Method for Slopes Using Three-Dimensional Point Cloud Data Based on Moving Least Square Theory. *Journal of Surveying Engineering*, 146(3), 04020013.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Solomon, J. M., 2019. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5), 1–12.
- Yan, W. Y., 2021. Scan line void filling of airborne LiDAR point clouds for hydroflattening DEM. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, 6426–6437.
- Yifan, W., Wu, S., Huang, H., Cohen-Or, D., Sorkine-Hornung, O., 2019. Patch-based progressive 3d point set upsampling. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5958–5967.
- Yu, L., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2018a. Ecnnet: an edge-aware point set consolidation network. *Proceedings of the European Conference on Computer Vision (ECCV)*, 386–402.
- Yu, L., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2018b. Pu-net: Point cloud upsampling network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2790–2799.
- Zhou, K., Dong, M., Arslanturk, S., 2021. "Zero Shot" Point Cloud Upsampling. *arXiv preprint arXiv:2106.13765*.



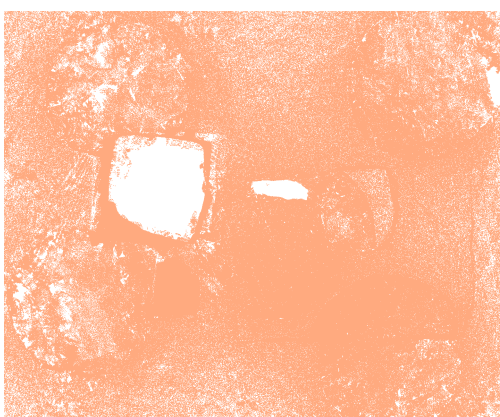
Wine Press



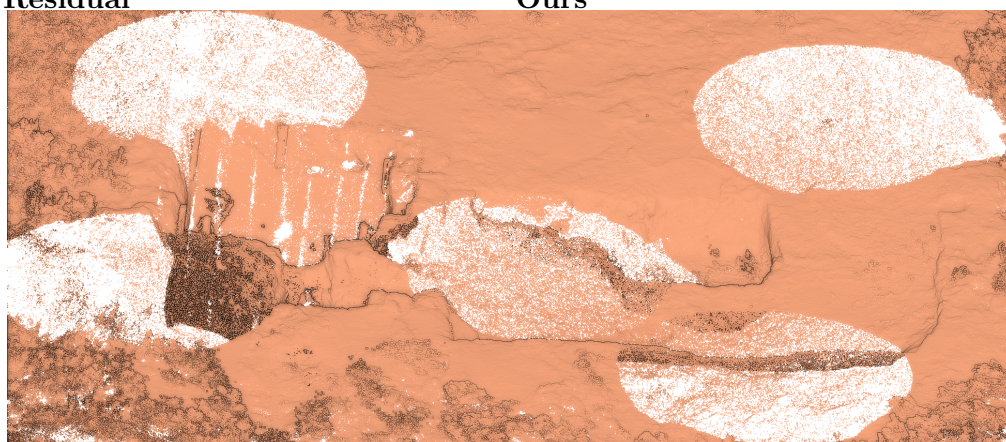
Residual



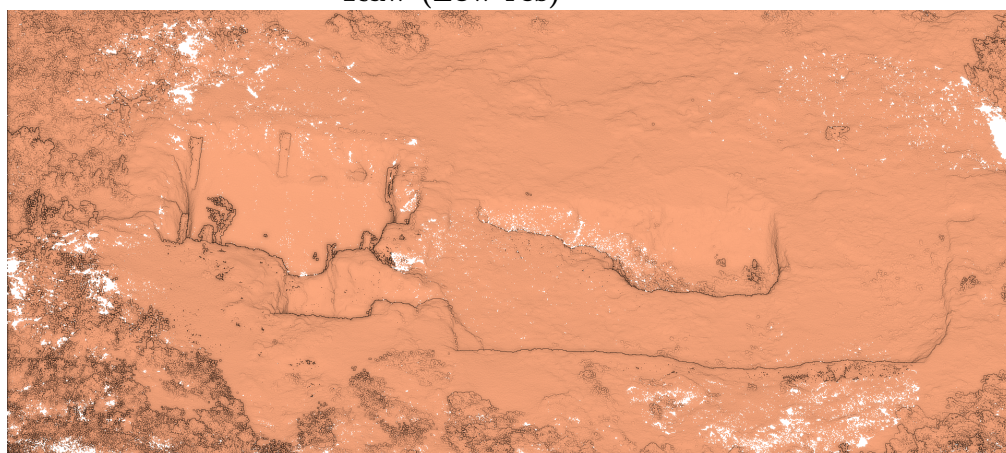
Raw (Low res)



Ours



Raw (Low res)



Ours

Figure 6. Upsampling of the wine press point cloud, the proposed method generates a point cloud with rich geometric details.