IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 11 September 2023, accepted 30 October 2023, date of publication 13 November 2023,
date of current version 21 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3332141

RESEARCH ARTICLE

CloudUP–Upsampling Vibrant Color Point Clouds Using Multi-Scale Spatial Attention

YONGJU CHO[1], RIMSHA TARIQ[2], USAMA HASSAN[3], JAVED IQBAL[4], ABDUL BASIT[2], HYON-GON CHOO[1], REHAN HAFIZ[2], AND MOHSEN ALI[3]

[1]Immersive Media Research Laboratory, Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea
[2]Department of Computer Engineering, Information Technology University, Lahore 54600, Pakistan
[3]Department of Computer Sciences, Information Technology University, Lahore 54600, Pakistan
[4]Department of Electrical Engineering, Information Technology University, Lahore 54600, Pakistan

Corresponding author: Javed Iqbal (javed.iqbal@itu.edu.pk)

This work was supported by the Electronics and Telecommunications Research Institute (ETRI) funded by the Korean Government (The research of the fundamental media contents technologies for hyper-realistic media space) under Grant 23ZH1220.

ABSTRACT In recent years, there has been a noticeable increase in the inclination towards digitizing our surroundings, encompassing various domains such as virtual reality, cultural heritage conservation, and architectural representation. The computation of high-resolution three-dimensional (3D) colored point clouds and meshes holds significant importance for such applications. However, traditional structure-from-motion (SfM) techniques may produce sparse 3D point clouds when low-resolution input images are used, resulting in a low-quality mesh generation. Traditional point cloud upsampling techniques that improve the 3D point cloud resolution typically work on LiDAR-generated point clouds devoid of color information. Furthermore, most learned point cloud upsampling techniques compute graph features that capture local information by identifying a local neighborhood in a limited region around a point and hence may result in sub-optimal representation. To address these limitations, we propose *CloudUP*, a colored 3D point cloud upsampling approach that utilizes multi-scale spatial attention. Specifically, we design a novel Multi-Scale Point-Cloud Feature Extractor (MPFE) by employing attention across the scales to extract point cloud features and effectively capture 3D shape information of the points relative to its neighborhood. We further extract spatial neighborhood-guided color features used to predict the color for the upsampled points. The color prediction is trained with a content-preserving loss function that aims to maintain intricate details and vivid colors. Our color refinement pipeline is guided by a vibrant colored dataset (collected by us) to assist in preserving the 3D contents.

INDEX TERMS Point cloud upsampling, point cloud color upsampling, multi-scale features, local shape approximation, vibrant colors.

I. INTRODUCTION

The interest in augmented and virtual reality and the overwhelming response towards metaverse-like virtual immersive environments fuel the demand for high-resolution, realistic colored 3D models. With the much easier accessibility of 3D sensors like LiDAR, point clouds are increasingly getting popular for the collection of data, which is crucial in robotics [1] and autonomous vehicles [2]. However, these sensors produce sparse and noisy point clouds and can usually not capture small or far away objects [3] and fine details. On the contrary, structure from motion (SFM) methods, such as COLMAP [4] and MVS-methods [5], [6], [7] with pose estimation [8] and fusion [9] may also result in a sparse (low resolution) point cloud if only a small number of views/images are available and/or these images have low-resolution. 3D point cloud upsampling has been pitched as a viable scheme to add vertices, leading to better quality mesh generation for LiDAR-based point clouds [10]. However, since most of these techniques focus on LiDAR-based point clouds, they do not consider color and may not be directly applied to upsample low-resolution colored point clouds. One

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

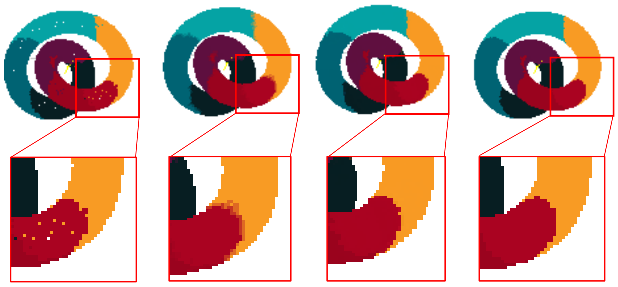

FIGURE 1. Top Row: (a) Input sparse colored point cloud, (b) colors upsampled using a KNN-based color estimation, (c) colors upsampled by our proposed scheme, (d) high-resolution Ground Truth (GT) point cloud. Bottom Row: Zoomed in patches to highlight differences. The KNN upsampled point clouds suffer from smoothness and blurriness effects, whereas our proposed strategy keeps the edges sharp.

possible way to upsample colors is to interpolate color values using existing approaches, following points upsampling. However, it is noted from the literature [11], [12], [13] that most of the color upsampling methods use K-Nearest Neighbors (KNN) based interpolation schemes for color upsampling, which results in blurry and poor-quality colors.

For this, we presented a color upsampling method that refines KNN colors and enhances the visual quality of the point cloud as shown in (Fig. 1). Given a sparse point cloud with colors in (a), (b) shows the output of color upsampling using KNN. It is quite evident from the figure that KNN smoothes and blurs the colors while the proposed method (c) caters to the precision of color in the local region.

Also for the task of point upsampling, the previously proposed methods extract dynamic graph representation. Although they produce impressive results, they are not expressive enough to represent the downstream task. The proposed dynamic feature extractors in these methods, rely on the K-nearest neighborhood (KNN) to construct a "local neighborhood" to compute features. This "K" is fixed for the whole point cloud. For small "K" the local neighborhood will resemble a planar region and fail to capture the surface pattern. Increasing "K" will result in sampling large neighborhoods that may be too complex to be represented by methods like DGCNN adequately. Any noise/artifact in the up-sampled vertices (points) shall be decremental to the overall visual quality of a colored point cloud. In this context, we observed that the state-of-the-art [3], [14], [15], [16] schemes for point cloud upsampling mostly rely on dynamic graph-based feature extractors [17] in a limited neighborhood and may fail to capture the global context of the surface geometry.

Contributions: To overcome the discussed limitations, we present CloudUP, a colored 3D point cloud upsampling scheme. Our contributions are fourfold. Secondly, a learned, content-preserving color predictor that produces sharp and vibrant upsampled colored point clouds. We use spatial neighborhood information to generate meaningful color features representing local color content. These color features and their corresponding spatial features predict refinement over KNN interpolation-based initial colors. Thirdly, a Vibrant Color Dataset (VCD) to guide the color prediction pipeline. Fourthly, we provide a succinct analysis of the plausibility of various quality matrices for color up-sampling. Finally, we test our scheme on non-colored (SkecthFab) and colored (LS-PCQA) point cloud datasets.

II. RELATED WORK

Due to significant applications in AR/VR technologies, robotics [1] and autonomous vehicles [2] as well as medical imaging and organs reconstruction [18], [19], point cloud upsampling research has seen a significant improvement over recent years. Specifically, deep learning-based methods have been more successful than the conventional optimization methods [20], [21], [22], [23], [24]. However, point cloud upsampling is a difficult task, because the point cloud data exists in unordered point sets. Thus, the optimization methods may fulfill the purpose of cloud upsampling, but they are not data-driven and may be prone to several limitations like complex geometries [14]. Furthermore, prior information, such as normals, is required as input in many optimization methods. With the introduction of deep neural networks such as PointNet++ [25] and DGCNN [17], it was possible to encode and learn spatial features from the point clouds.

A. POINT UPSAMPLING

Deep learning based point upsampling methods can be divided into two subcategories on the basis of processing the input point clouds. The first set of methods takes the whole point cloud as input and upsamples it exploiting their global shape information [26]. The second set of approaches divides the point cloud into patches, and upsamples the point cloud based on local neighborhood information [3], [15], [27], [28]. The patch-based methods have recently shown great success in upsampling point clouds at multiple resolutions by extracting local point information using a graph-based neural network and then learning local approximation based on the point features to upsample points [15].

The authors in [14] showed superior performance compared to all conventional optimization methods by proposing the first deep learning based approach for point cloud upsampling, PU-Net. It uses PointNet++ to learn hierarchical features for each point and then upsample them using multi-branch multi-layer perceptrons (MLPs). Furthermore, PU-Net does not consider spatial relations between neighboring points, restricting the method from producing uniform points. Following PU-Net, the authors of MPU [29] proposed patch-based progressive 3D Point Set upsampling approach capable of upsampling the points by a factor of up to 16. Since it is a progression process, MPU requires great computation power, and training a network is relatively non-trivial requiring step-by-step guidance. Similarly, PU-GAN [30] aimed to learn the distribution of the upsampled points.

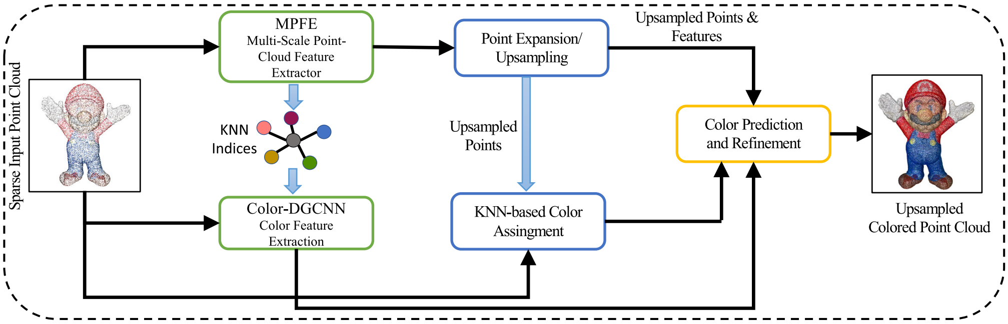

FIGURE 2. **CloudUP Pipeline: Illustration of the complete block diagram of the proposed approach. Given a sparse colored point cloud, our method upsamples the points and colors to generate a dense point cloud.**

Although a great emphasis was put on the performance gain of the discriminator, generator architecture did not contribute much to the final output.

PUGeo-Net [15] and MAFU [31] tried to preserve the points' uniformity and the underlying objects' geometry. PUGeo-Net specifically tried to learn the first order and second order fundamental forms of local geometry through the supervision of normals and spatial features of points.

Moreover, PU-GCN [3] proposed a novel model, NodeShuffle; which uses a graph convolutional network to encode local point information from point neighborhoods. More recently, Neural Points [27] has developed a novel representation of point clouds where each point represents a local geometric shape through neural fields. Ultimately, these local neural fields are combined to form a global surface. The aforementioned works highlight the significant influence of learning the correspondences between local point neighborhood in the sparse point cloud and its upsampled resolution on the quality of upsampling the point cloud based on local shape approximation. However, to the best of our knowledge, none of the existing approaches used the multi-scales information along with local neighborhoods to extract features for points and colors during upsampling. We thus propose a novel colored point cloud upsampling approach with an attention-based multi-scale point-cloud feature extraction (MPFE) module to capture the local neighborhood, information across the scales, and the correspondence at the input level to enhance the point upsampling performance.

B. COLOR UPSAMPLING

In literature, very few point cloud upsampling approaches [11], [12] consider color upsampling. The authors in YuZu [11] used K-nearest neighbors to upsample colors in a point cloud for volumetric video streaming. FGTV [12] created a KNN graph by linking 3D points in a point cloud, refining colors using weighted L1-Norm while keeping the locations of 3D points fixed. However, these approaches have high computational complexity and not viable for large point clouds. While most of the techniques focus on points upsampling [3], [14], [15], [30], only a couple of methods tried to predict and refine colors for dense point clouds [11], [12], [13]. To the best of our knowledge, no method provides a combined learned pipeline for points and colors upsampling for a given sparse point cloud. We propose CloudUP to simultaneously upsample points and their respective colors for a given sparse point cloud to create a high-quality colored 3D point cloud.

III. METHODOLOGY/APPROACH

We propose a two-stream architecture (Fig. 2) for the upsampling of colored point clouds. The first stream upsamples the points using a cross-attention-based multi-scale point-cloud feature extractor (MPFE), which computes features for each point in the input *sparse-point cloud*. The second stream consists of the color prediction module, which predicts and refines the colors for each point in the upsampled point cloud. We try to preserve the color variation by introducing a color variance conformity loss to avoid low-resolution color generation. Features computed at different stages of the point upsampling stream are used to compute color features by color prediction scheme.

A. PRELIMINARIES

Let $X = \{x_i \in \mathbb{R}^{6\times1}\}_{i=1}^{N}$ be a sparse point cloud having N points with $\mathbb{R}^{3\times1}$ spatial coordinates and $\mathbb{R}^{3\times1}$ RGB color values for each respective point. The aim is to generate a colored, dense point cloud $X_u = \{x_j \in \mathbb{R}^{6\times1}\}_{j=1}^{Q}$ with R times upsampled points and colors. Q represents the number of points in the upsampled point cloud, i.e., $Q = N \times R$, where R is the upsampling ratio. Similarly, for the training process, the corresponding upsampled ground truth point clouds are given by $Y_u = \{y_j \in \mathbb{R}^{6\times1}\}_{j=1}^{Q}$.

B. POINT CLOUD UPSAMPLING

Point upsampling refers to the technique used to generate a dense point cloud from a sparse, low-resolution point cloud while maintaining the original geometric shape of the point cloud. This paper introduces a patch-based upsampling method that leverages multi-resolution data features to effectively capture fine and coarse details within a sparse

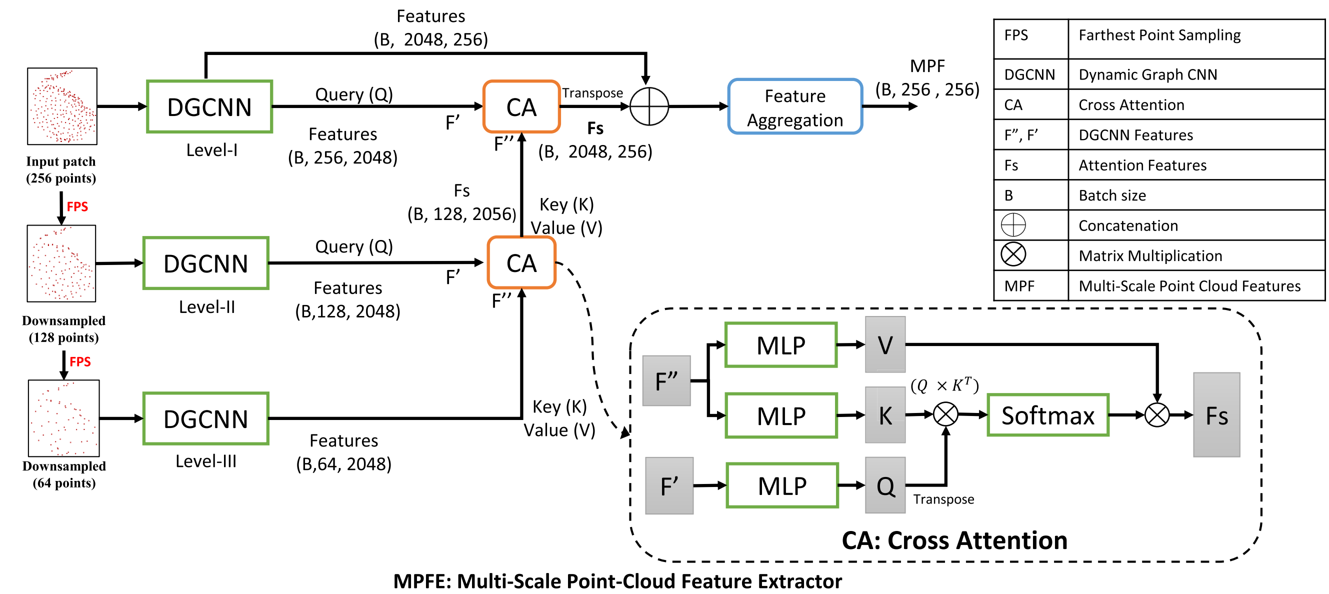

FIGURE 3. **Proposed Multi-scale point cloud feature extraction network with cross attention.**

input patch. By doing so, we successfully mitigate the adverse effects of noise on spatial features.

1) MULTI-SCALE POINT-CLOUD FEATURE EXTRACTOR (MPFE)

Point upsampling methods have recently employed a Dynamic graph network-like structure to extract local neighborhood information for each point effectively. This approach enables capturing similar structural information among points within a patch. The feature extractors in this approach update the feature correspondences based on local context information during each iteration. This limitation prevents the network from effectively capturing both local and global structures simultaneously. To address this limitation, we propose a cross-attention-based multi-scale point cloud feature extractor (MPFE). This extractor allows us to learn the structure of the input patch at multiple resolutions by utilizing a feature pyramid. Further, cross-attention (CA) is used to learn the association between features at different resolutions by refining features to preserve the geometric structure of the point patch after upsampling.

Spatial Attention-Based Multi-Scale Features: To extract multi-scale feature input, each sparse point patch (P_r) is downsampled into two sub-patches by x2 (P_r2) and x4 (P_r4) using Farthest Point Sampling (FPS) [32]. To extract the locality of each patch, they are processed with DGCNN independently. The features for each patch are of size (B, 2048, M) where M defines the number of points in a patch and B is the batch size. Once features for all patches are extracted we apply cross-attention going from lower resolution to high resolution. Embeddings for features representing sub-patches are created initially. Then, these embeddings are used to apply cross-attention to the features that represent the original input patch. We then apply a function F_a, aggregating the original features and final embedding, to learn the important and salient aspects of cloud structure. The cross-attention is defined by (1),

$$CA(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

K represents the Key vector, Q represents the Query, and V represents the Value. Key (K) and Value (V) are generated at low-resolution features. While Query is generated from high-resolution features. The attention weights are calculated by taking the dot product between Q and K^T from different resolutions, normalizing, and applying softmax, which is used to weight the Value(V) vector. It is to be noted that F_a is a non-linear function defined using MLP.

2) POINTS EXPANSION

To upsample input points, we adopt a local shape approximation technique similar to [15]. For this, we first parameterize point x_i using its neighborhood feature extracted from MPFE to the 2D domain by applying the function F_m to learn an affine transformation (a_t). Given the upsampling ratio, we predict the offset for new points in the parametric domain. The learned affine transformation (a_t) is applied to the point offset to map them onto a 2D plane that is tangent to the sparse point. We add these transformed points to the initial sparse point to map them back onto the point space from the parametric 2D space. This learned affine transformation (a_t) also gives the sparse normal since the outer product of the first two columns gives the tangent to the plane. To further approximate the local shape of these newly transformed points, we define a mapping that learns the second-order approximation of the local shape. For this, we first replicate

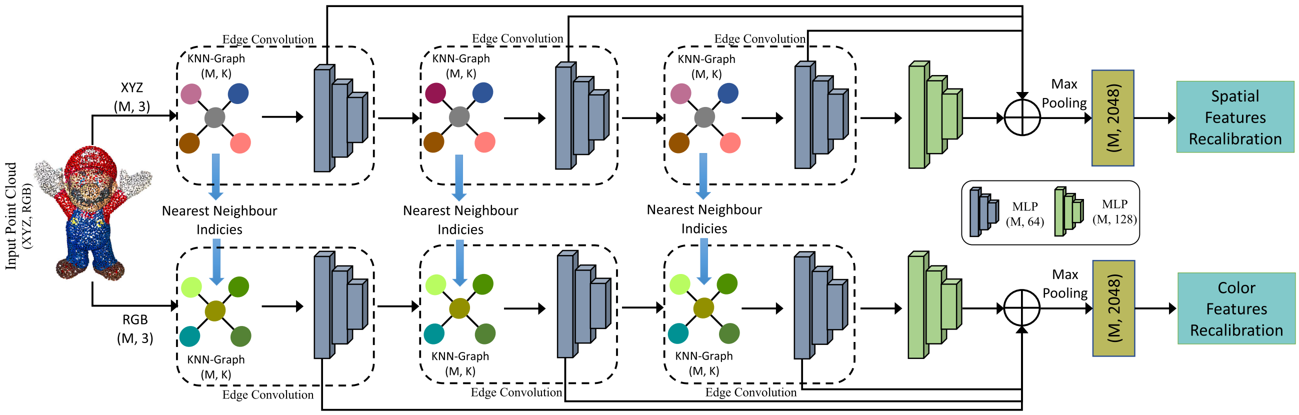

FIGURE 4. Color Feature Extraction: The top pipeline is level 1 of the spatial features extraction module (MPFE) using graph convolutions and cross attention. For color, a similar architecture like level-1 of MPFE is used; Color-DGCNN where the KNN-Indices from the level-1 of the MPFE are used to compute Spatial Neighbourhood Guided Color Features (SNCF).

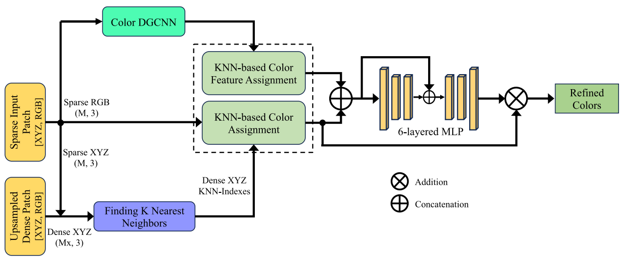

FIGURE 5. Color Refinement Pipeline: Patch from input sparse point cloud and patch from dense upsampled point cloud are passed to ***KNN-based Color Assignment*** module, which predicts initial colors. Initial colors along with upsampled and spatial neighborhood-guided color features are concatenated and passed to an MLP refining initial colors.

the input sparse point features, mapping them to their corresponding upsampled points from the parametric domain. We then input these upsampled points and their respective spatial attention-based multi-scale features to a parametric function to learn the second-order shape approximation and predict the offsets added back into the upsampled points to obtain the final dense points. Similarly, dense normals offsets are also predicted, given the sparse normals and multi-scale point features using a multi-layer neural network. The loss functions to train the point cloud upsampling network for both points and normals are given by Eq. (2) and Eq. (3).

$$L_{CD} = \frac{1}{RM}\left(\sum_{\mathbf{x_i} \in X_u} \|\mathbf{x_i} - \hat{\mathbf{x}}_i\|_2 + \sum_{\mathbf{y_j} \in Y_u} \|\mathbf{y_j} - \hat{\mathbf{y}}_j\|_2\right) \tag{2}$$

where $\hat{\mathbf{x}}_i$ is the nearest point from ground truth to the upsampled point $\mathbf{x_i}$. Similarly, for any ground truth point $\mathbf{y_j}$, the $\hat{\mathbf{y}}_j$ represents the nearest predicted point in X_u.

$$L_{\mathbf{n}} = \frac{1}{M}\sum_{i=0}^{M} min(\|\mathbf{n} - \hat{\mathbf{n}}\|_2, \|\hat{\mathbf{n}} + \mathbf{n}\|_2) \tag{3}$$

where $\hat{\mathbf{n}}$ are the corresponding normals from the ground truth point cloud.

C. COLOR UPSAMPLING

Color Upsampling is a process to predict high-resolution colors for upsampled point clouds using input sparse colored point clouds. Mapping between input sparse point clouds and upsampled point clouds can be challenging due to the unordered structure of the point cloud data. Furthermore, the upsampled point cloud may contain noise or errors. This makes the underlying task more difficult. We designed a three-phase pipeline for this purpose. Initial colors are assigned to dense, upsampled point clouds in the first phase. To accomplish this, we employ the nearest neighbor algorithm. Based on Euclidean distance, the nearest neighbor of each 3D point in a dense point cloud is calculated in a sparse point cloud. Each point in the dense point cloud is given the color information of its nearest neighbor. In the second phase, to maintain the geometric structure of the point cloud, we introduce Color-DGCNN shown in Fig. 4. Color-DGCNN extracts the color features by incorporating spatial neighborhood information from MPFE level-1. In the third stage, these color features are utilized to refine the initially assigned colors by predicting and adding change δ to them.

1) SPATIAL NEIGHBORHOOD GUIDED COLOR FEATURES (SNCF)

Each input sparse patch comprises unstructured data points, consisting of location and color information. To extract meaningful information from the input patch, it is necessary to establish the structure of the patch. This study introduces Color-DGCNN, a modified version of the dynamic graph convolutional neural network (DGCNN). The Color-DGCNN model consists of two parallel streams. One stream is responsible for spatial feature extraction from MPFE Level-I, while the other stream focuses on color feature extraction (see Fig. 4). The spatial feature extractor used in this study is a classical Dynamic Graph Convolutional Neural Network (DGCNN) [17]. It is applied at the MPFE-Level I for processing input patches. Integrating the color extraction pipeline with spatial features ensures the preservation of the point patch structure during the extraction of color information. Similar to the spatial feature extractor, the color features extractor undergoes three stages of EdgeConvolution on the color information obtained from the previous stage. The computation of a graph in EdgeConvolution is not possible using color information, as it will lead to a loss of structure and edge information. In our method, the spatial feature extractor graph is thus used during each EdgeConvolution stage, effectively preserving point patch structural and boundary information.

2) COLOR REFINEMENT

The initial colors assigned using nearest neighbors are refined using SNCF. For this, we need to refine SNCF features for each point in an upsampled point patch. In Literature [14], [15], [16] for Point Upsampling, the features were duplicated by upsample ratio and refined for upsampled points. However, duplicating color features can add anomalies to upsampled colors as color information on a point can vary irrespective of point structure. Hence, we use the KNN algorithm to replicate the color features to dense points shown in Fig 5. Each point in the dense patch is assigned a feature based on its spatial nearest neighbor in the sparse patch. The color features are then refined using a non-linear function F_r predicting deltas δ in initial colors, refining colors without blurring the color edges. Where F_r is a non-linear function represented by an MLP.

3) COLOR UPSAMPLING LOSS

In this section, we introduce color loss components for color upsampling. We have combined three different loss functions to estimate correct color predictions, avoiding excessive smoothness and blurriness.

$$L_{rgb} = \frac{1}{n}\sum_{i=0}^{n} \| C_{GT}^{\phi(i)} - C_{GEN}^{i} \|_2 \tag{4}$$

Equation (4) minimizes the mean loss over all the points in a given batch, thus introducing the smoothness effect in the predicted colors. To overcome this, we define the local neighborhood-based variance loss (5). Specifically, for all points in the upsampled point cloud and the ground truth point cloud, we compute RGB colors variance V_{GEN}^i and V_{GT}^i with their 8-nearest neighbors, respectively. Then, we minimize the difference between the GT points variance V_{GT}^i and the upsampled points variance V_{GEN}^i using (5).

$$L_{var} = \frac{1}{n}\sum_{i=0}^{n} \| V_{GT}^{\phi(i)} - V_{GEN}^{i} \|_2 \tag{5}$$

To further minimize the blurriness introduced by averaging used in KNN-based color upsampling, we transform the RGB colors to HSV space and compute MSE loss (6) for upsampled and GT points.

$$L_{hsv} = \frac{1}{n}\sum_{i=0}^{n} \| HSV_{GT}^{\phi(i)} - HSV_{GEN}^{i} \|_2 \tag{6}$$

The total loss for color refinement is the summation of L_{rgb}, L_{var}, and L_{hsv} and is given by (7),

$$L_{Color} = \alpha * L_{rgb} + \beta * L_{var} + \gamma * L_{hsv} \tag{7}$$

where, α, β and γ are the scaling factors for each loss component. The $\phi(i)$ is the correspondence function computed by calculating the spatial locations based on nearest neighbors between the up-sampled and ground-truth patches.

IV. EXPERIMENTS

We thoroughly evaluate the CloudUP scheme over multiple datasets for upsampling the 3D point cloud and color. Below, we provide the details of datasets, model architecture, training details, & evaluation strategy.

A. EXPERIMENTAL SETUP

1) DATASETS

To train and evaluate the proposed CloudUP, we used two publically available 3D points datasets, **LS-PCQA Dataset** (used by [33]) and **SketchFab Dataset** (used by [16], [34]). The **LS-PCQA Dataset** [35] is a synthetic dataset with 104 colored surface meshes constructed for different objects. We use 78 point clouds for training & 15 for testing in this work. Note that the point clouds are sampled over available mesh surfaces uniformly. We also evaluated our point upsampling on the **SketchFab Dataset** [15], a non-colored synthetic LIDAR-based 3D point cloud dataset predominantly used to evaluate point cloud upsampling. It contains 103 meshes, with 90 for training and 13 for testing having variable geometric topologies.

2) MODEL ARCHITECTURE AND TRAINING DETAILS

The architecture of CloudUP is illustrated in Figure 2. The input point cloud is partitioned into patches of size 256. The patches are fed to MPFE and Color-DGCNN, which compute spatial and color features for each patch separately. The point expansion technique increases the resolution of the input patches by incorporating spatial features. The color refinement module utilizes up-sampled points and color features to colorize patches during the process of upsampling. We employed a 6-layered MLP (Multi-Layer Perceptron) with a skip connection in the refinement network shown in Fig. 5 with LeakyReLU used as an activation function.

Our proposed approach is trained in two stages. In the first stage, we train the MPFE and points expansion modules for 800 epochs with a batch size of 28, following the approach used in [15]. Once trained, we freeze these modules for back-propagating the gradients. In the next stage, we train the Color-DGCNN & Color Refinement network for 100 epochs with a batch size of 32. We use the farthest point sampling algorithm to generate overlapping patches of 256 points from the input point cloud. During training, we use the Adam optimizer with a learning rate of 0.001 and the same loss scaling factors as defined in [15], except for color loss, where α, β, and γ are all set to 10. All experiments

TABLE 1. Quantitative comparison of the proposed CloudUP with SOTA methods on SketchFab and LS-PCQA datasets.

Approach	Points: on SketchFab dataset			Points: on LS-PCQA dataset		
	CD(10^{-3})	HD (10^{-2})	EMD (10^{-3})	CD(10^{-2})	HD (10^{-2})	EMD (10^{-3})
PU-Net [14]	7.55	2.51	2.55	2.26	6.42	5.54
MPU [29]	6.58	2.32	1.83	1.96	5.11	5.17
PU-GAN [30]	6.74	2.33	1.98	2.16	5.20	6.16
PUGeo-Net [15]	6.90	2.20	8.00	2.40	5.10	5.00
PU-GCN [3]	**6.60**	2.24	1.98	**1.91**	5.11	4.94
PUGAC-Net [16]	6.84	2.33	1.78	1.99	6.71	5.30
CloudUP (Ours)	7.80	**1.33**	**1.50**	2.54	**3.82**	**3.30**

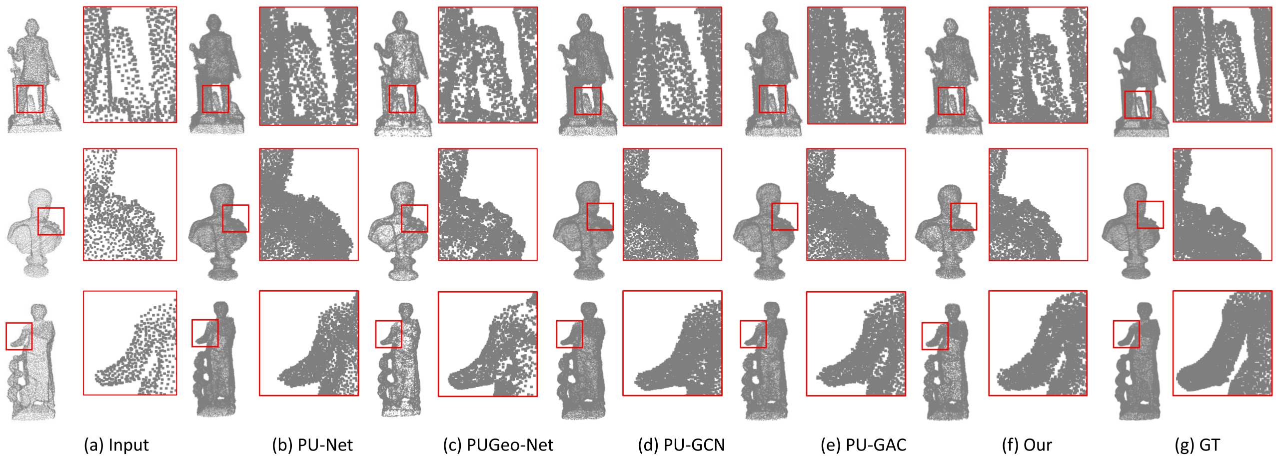

FIGURE 6. Qualitative results for CloudUP points upsampling on SketchFab dataset. CloudUP upsamples the points precisely leading to accurate point clouds with preserved structure.

TABLE 2. Quantitative comparison of the proposed CloudUp with KNN-based methods on LS-PCQA dataset, in terms of predicted colors. For evaluation correspondences are obtained from Points CD, and then used to compute MSE, PSNR, and MDV.

Approach	CD-MSE	CD-PSNR	MDV
KNN color (K=3)	0.16	11.13	0.0071
CloudUp (Without VCD)	0.15	11.15	0.072
CloudUp (With VCD)	**0.13**	**11.15**	**0.0041**

use a Quadro-RTX with 48GB of memory in the PyTorch framework.

3) EVALUATION METRICS

We conducted comprehensive qualitative and quantitative evaluations on two datasets. To quantitatively evaluate our approach, we employ commonly used metrics, including Chamfer distance (CD), Hausdorff distance (HD), Earth Mover Distance (EMD), and mean square error (MSE). The metrics used are inspired by the following papers [10], [22], [36]. We employ a two-step approach to evaluate the qualitative and quantitative performance of color upsampling. Firstly, we establish spatial correspondences between two point clouds using (CD) Chamfer Distance. Subsequently, we compare the color values at these corresponding points using mean squared error (MSE) and peak signal-to-noise ratio (PSNR). We also evaluated our results using the mean difference variance (MDV), described further in the next section.

MDV: The Mean Difference Variance (MDV) metric is used to assess the performance of color upsampling by evaluating how effectively color variations are handled. For this, we compute the mean absolute difference between the variances of the respective patches of the upsampled and GT point clouds. To calculate the MDV, the upsampled and GT point clouds are divided into smaller patches, each containing 256 points. To ensure similarity between compared patches, the centroids of corresponding patches in the upsampled and ground truth (GT) point clouds are maintained at the same position. The color variance of a patch is computed by calculating the variance for each color channel (RGB) and then taking the mean. The MDV value, as represented in (8), is calculated as the mean of the absolute difference between the variance of upsampled point cloud patches and the variance of the ground truth (GT) point cloud patches. A lower Mean Delta Value (MDV) indicates a higher degree of accuracy in predicting colors that closely match the ground truth colors.

$$\mathrm{MDV} = \frac{1}{n}\sum_{i=0}^{n} \left| V_{rgb}(P_{gt_i}) - V_{rgb}(P_{up_i}) \right| \tag{8}$$

$V_{rgb}(P_{gt_i})$ and $V_{rgb}(P_{up_i})$ are the variances of the i^{th} patch in the GT point cloud and the upsampled point clouds, respectively.

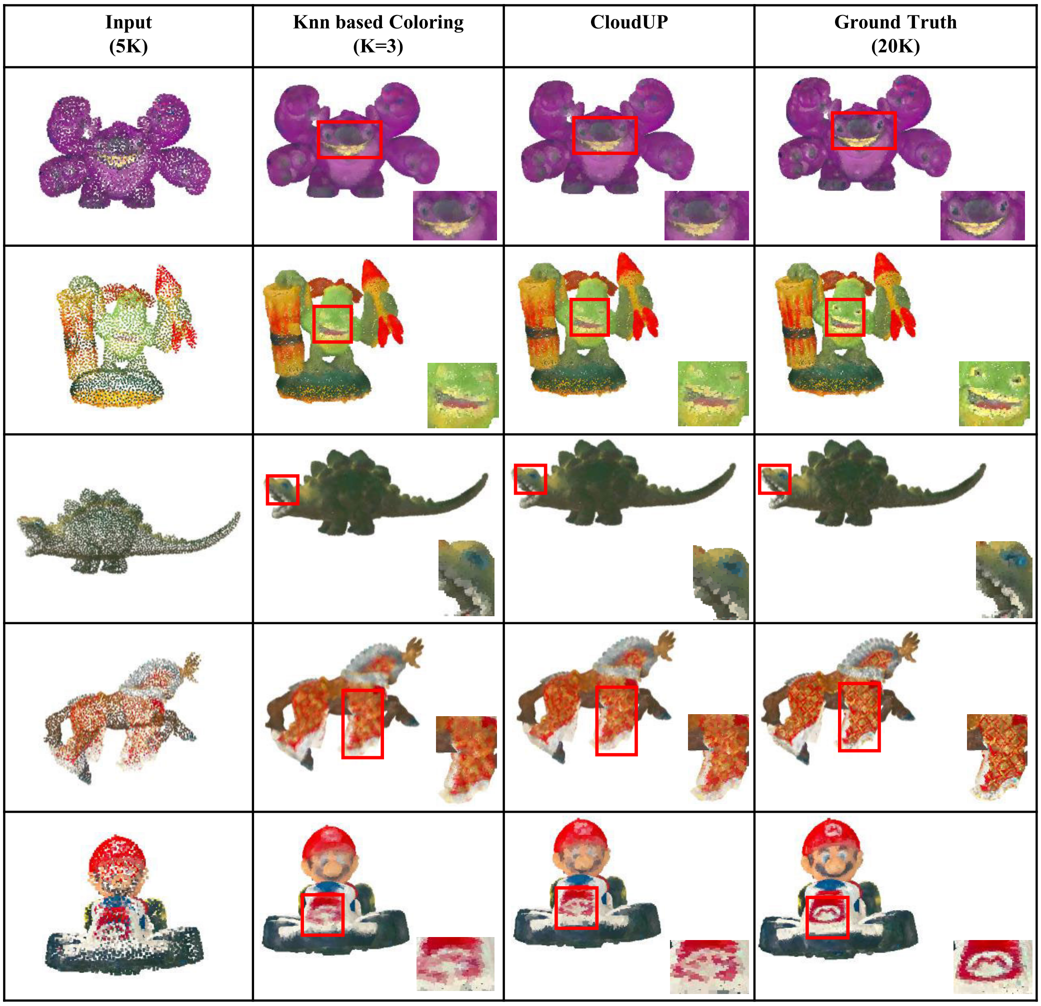

FIGURE 7. Qualitative results for CloudUP on LS-PCQA dataset. In general, CloudUP provides up-sampled point clouds with vibrant colors & details.

TABLE 3. Average Color Entropy for LS-PCQA and VCD datasets (10^{-2}).

Dataset	Entropy
LS-PCQA [35]	2.03
VCD	**9.01**

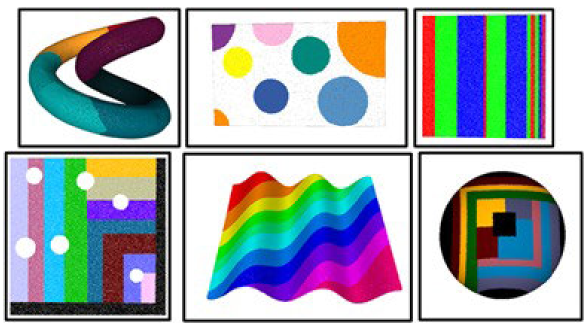

FIGURE 8. A few sample 3D point clouds from our Vibrant Color Dataset (VCD).

B. EXPERIMENTAL RESULTS

This section details the quantitative and qualitative experimental results on synthetic and real 3D scan datasets.

1) QUANTITATIVE EVALUATION

Table 1 and Tabel 2 provide the quantitative results of our proposed points upsampling and color refinement schemes.

Specifically, Table 1 compares the error metrics (CD, HD, and EMD) evaluated over the test scans of the respective datasets. For SOTA methods, the codes from the official repositories were used to train on the respective datasets as per the suggested training details provided by the authors. The table shows that the proposed approach consistently outperforms all the SOTA methods in HD and EMD for all datasets while keeping comparable performance for CD metrics. Table 2 provides the quantitative result for our color upsampling and refinement strategy for the LS-PCQA dataset. It can be observed that CloudUP can consistently refine the color as compared initial KNN estimate. As shown in Table 2, the quantitative results of the proposed CloudUP

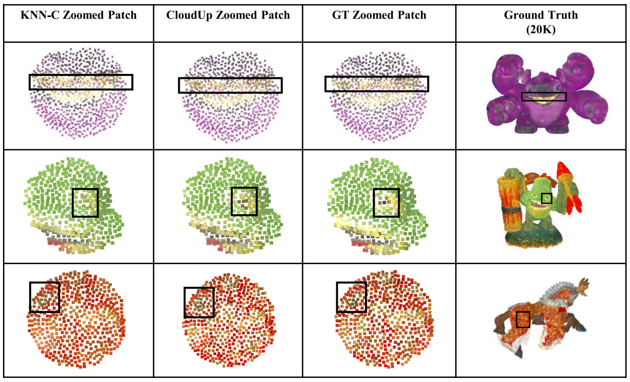

FIGURE 9. **Qualitative results at Patch Level of CloudUP for point cloud color refinement. KNN-based color estimates were used to compare the color values. It can be seen that KNN interpolation smooths out the colors at the edges. Zoom for the best visibility.**

show better performance for Color refinement and point upsampling. Better points upsampling combined with refined colors leads to accurate color point-cloud upsampling results, as shown in Figure. 7.

Vibrant Color Dataset: The LS-PCQA dataset lacks point clouds with rich color information, which limits its suitability for training models in color refinement. We created a new dataset called the Vibrant Color Dataset (VCD) to address this limitation. The VCD consists of 10 basic shapes, such as rectangles, triangles, planes, and prisms, each with different colors. Figure 8 provides a sample of our VCD Dataset. As evidence for our hypothesis, we compared the mean entropy of color histograms, a measure of texture content and sharpness, for the VCD dataset and the LS-PCQA dataset. Table 3 presents the entropy of the LS-PCQA dataset and our colorful dataset, which demonstrates the rich color content of our dataset. As shown in table2, training CloudUP with the combined LS-PCQA and VCD datasets improves the color-refinement network's performance.

2) QUALITATIVE EVALUATION

Finally, we provide the qualitative results of the color refinement in Figure 7 and in Figure 9. Compared to KNN-based color upsampling, CloudUP refines the colors precisely, leading to accurate and sharp boundaries as shown in Figure 7. Figure provides qualitative results for five scans (4Arms-Monster, Green-Monster, Dinosaur, Horse, and Mario-car). GT shows each case's high-resolution ground truth 3D point cloud at 20K point resolution. GT is downsized to 5K points and provided as an input for 4X upsampling to both KNN-based color upsampling and CloudUP. The resulting upsampled point clouds are visualized in figure 7 using Open3D.

Figure 9 compares a zoomed-in view of upsampled patches. Overall, CloudUP's enhanced SNCF-guided color refinement produces 3D point clouds with enhanced visual quality. In particular, the KNN-based color estimate completely distorts the Green Monster's eye, whereas CloudUP's

TABLE 4. **Ablation experiments for color refinement network, KNN-C: KNN-based Color, L_{rgb}: RGB-MSE loss, L_{var}: Variance MSE loss, L_{hsv}: HSV-MSE loss.**

Method	SNCF	KNN-C	L_{rgb}	L_{hsv}	L_{var}	CD-MSE	CD-PSNR
CloudUP		✓				0.182	10.800
	✓					0.177	11.095
	✓	✓				0.179	11.083
	✓	✓	✓			0.157	**12.270**
	✓	✓	✓	✓		0.120	11.210
	✓	✓	✓	✓	✓	**0.118**	11.2

TABLE 5. **Effect of multi-scale feature extraction & aggregations on LS-PCQA dataset. Values are on a scale of 10^{-2}.**

Approach	CD	HD	EMD
CloudUP (no-MPFE)	2.8	5.0	0.80
CloudUP (MPFE-KNN) [29]	2.57	4.22	0.35
CloudUP (MPFE-Cross Attention)	**2.54**	**3.82**	**0.33**

color refinement pipeline offers precise upsampling with preserved features. Thus, in comparison to SOTA methods, the proposed method significantly refines the color by preserving the geometry and color information.

C. ABLATION EXPERIMENTS

A comprehensive ablation study was conducted to analyze the distinct contributions made by each component within the point upsampling and color prediction/refinement model. The outcomes of various loss functions and additional components assessed for the color refinement network are presented in Table 4. In the context of point cloud upsampling, the utilization of cross-attention-based feature extraction and multi-scale point-cloud feature extraction (MPFE) has demonstrated more favorable outcomes in comparison to single-level feature extraction and a basic nearest neighbor-based multi-scale feature assignment, as evidenced by the data presented in table 5.

V. CONCLUSION

A novel approach has been proposed for addressing the two components involved in 3D color point cloud upsampling, namely the generation of new points (point upsampling) and the assignment of colors to these points (color upsampling). For 3D point upsampling, we designed a Multi-Scale Point-Cloud Feature Extractor (MPFE) that uses cross attention considering spatial information to capture the 3D shape information of a point's neighborhood. For color upsampling, we have designed a mechanism to compute the point cloud's Spatial Neighbourhood-based Color Features (SNCF). These features are then utilized to refine a KNN-based color estimation. We provide a Vibrant Color Dataset (VCD) to aid in the color refinement pipeline. Analyses conducted on two standard data sets, both qualitative and quantitative, demonstrate that our method yields better results with superior visual quality.

REFERENCES

[1] A. Geiger, J. Ziegler, and C. Stiller, "StereoScan: Dense 3D reconstruction in real-time," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 963–968.

[2] X. Ma, Z. Wang, H. Li, P. Zhang, W. Ouyang, and X. Fan, "Accurate monocular 3D object detection via color-embedded 3D reconstruction for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6850–6859.

[3] G. Qian, A. Abualshour, G. Li, A. Thabet, and B. Ghanem, "PU-GCN: Point cloud upsampling using graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11683–11692.

[4] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4104–4113.

[5] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "MVSNet: Depth inference for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 767–783.

[6] X. Gu, Z. Fan, S. Zhu, Z. Dai, F. Tan, and P. Tan, "Cascade cost volume for high-resolution multi-view stereo and stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2495–2504.

[7] Y. Cai, M. Cao, L. Li, and X. Liu, "An end-to-end approach to reconstructing 3D model from image set," *IEEE Access*, vol. 8, pp. 193268–193284, 2020.

[8] M.-J. Shin, W. Park, J. H. Kim, J. Kim, K.-J. Yun, and S.-J. Kang, "Camera pose estimation framework for array-structured images," *ETRI J.*, vol. 44, no. 1, pp. 10–23, Feb. 2022.

[9] S. Galliani, K. Lasinger, and K. Schindler, "Massively parallel multiview stereopsis by surface normal diffusion," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 873–881.

[10] C.-H. Yeh and W.-C. Lin, "Enhanced point cloud upsampling using multi-branch network and attention fusion," in *Proc. Int. Conf. Comput. Syst., Inf. Technol., Electr. Eng. (COSITE)*, Oct. 2021, pp. 51–56.

[11] A. Zhang, C. Wang, B. Han, and F. Qian, "YuZu: Neural-enhanced volumetric video streaming," in *Proc. 19th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*, Renton, WA, USA: USENIX Association, 2022, pp. 137–154.

[12] C. Dinesh, G. Cheung, and I. V. Bajić, "Super-resolution of 3D color point clouds via fast graph total variation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 1983–1987.

[13] T. M. Borges, D. C. Garcia, and R. L. de Queiroz, "Fractional super-resolution of voxelized point clouds," *IEEE Trans. Image Process.*, vol. 31, pp. 1380–1390, 2022.

[14] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "PU-Net: Point cloud upsampling network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2790–2799.

[15] Y. Qian, J. Hou, S. Kwong, and Y. He, "PUGeo-Net: A geometry-centric network for 3D point cloud upsampling," in *Proc. Eur. Conf. Comput. Vis.* Tel Aviv, Israel: Springer, 2020, pp. 752–769.

[16] B. Han, X. Zhang, and S. Ren, "PU-GACNet: Graph attention convolution network for point cloud upsampling," *Image Vis. Comput.*, vol. 118, Feb. 2022, Art. no. 104371.

[17] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Oct. 2019.

[18] J. Zhang, L.-R. Gong, K. Yu, X. Qi, Z. Wen, Q. Hua, and S. H. Myint, "3D reconstruction for super-resolution CT images in the Internet of Health Things using deep learning," *IEEE Access*, vol. 8, pp. 121513–121525, 2020.

[19] N. Chong, "3D reconstruction of laparoscope images with contrastive learning methods," *IEEE Access*, vol. 10, pp. 4456–4470, 2022.

[20] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, "Computing and rendering point set surfaces," *IEEE Trans. Vis. Comput. Graphics*, vol. 9, no. 1, pp. 3–15, Jan. 2003.

[21] H. Huang, S. Wu, M. Gong, D. Cohen-Or, U. Ascher, and H. Zhang, "Edge-aware point set resampling," *ACM Trans. Graph.*, vol. 32, no. 1, pp. 1–12, Jan. 2013.

[22] Y. Lipman, D. Cohen-Or, D. Levin, and H. Tal-Ezer, "Parameterization-free projection for geometry reconstruction," *ACM Trans. Graph.*, vol. 26, no. 3, p. 22, Jul. 2007.

[23] S. Wu, H. Huang, M. Gong, M. Zwicker, and D. Cohen-Or, "Deep points consolidation," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–13, Nov. 2015.

[24] C. Dinesh, G. Cheung, and I. V. Bajić, "3D point cloud super-resolution via graph total variation on surface normals," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 4390–4394.

[25] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[26] S. Qiu, S. Anwar, and N. Barnes, "PU-Transformer: Point cloud upsampling transformer," in *Proc. Asian Conf. Comput. Vis.*, 2022, pp. 2475–2493.

[27] W. Feng, J. Li, H. Cai, X. Luo, and J. Zhang, "Neural points: Point cloud representation with neural fields for arbitrary upsampling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 18633–18642.

[28] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "EC-Net: An edge-aware point set consolidation network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 386–402.

[29] W. Yifan, S. Wu, H. Huang, D. Cohen-Or, and O. Sorkine-Hornung, "Patch-based progressive 3D point set upsampling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5958–5967.

[30] R. Li, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "PU-GAN: A point cloud upsampling adversarial network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7203–7212.

[31] Y. Qian, J. Hou, S. Kwong, and Y. He, "Deep magnification-flexible upsampling over 3D point clouds," *IEEE Trans. Image Process.*, vol. 30, pp. 8354–8367, 2021.

[32] Y. Eldar, M. Lindenbaum, M. Porat, and Y. Y. Zeevi, "The farthest point strategy for progressive image sampling," *IEEE Trans. Image Process.*, vol. 6, no. 9, pp. 1305–1315, Sep. 1997.

[33] Y. Liu, Q. Yang, Y. Xu, and L. Yang, "Point cloud quality assessment: Dataset construction and learning-based no-reference metric," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 19, no. 2s, pp. 1–26, Jun. 2023.

[34] W. Feng, J. Li, H. Cai, X. Luo, and J. Zhang, "Neural points: Point cloud representation with neural fields for arbitrary upsampling," 2021, *arXiv:2112.04148*.

[35] Y. Liu, Q. Yang, Y. Xu, and L. Yang, "Point cloud quality assessment: Dataset construction and learning-based no-reference metric," 2020, *arXiv:2012.11895*.

[36] A. Javaheri, C. Brites, F. Pereira, and J. Ascenso, "Improving PSNR-based quality metrics performance for point cloud geometry," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 3438–3442.

YONGJU CHO received the B.S. and M.S. degrees in electrical and computer engineering from Iowa State University, in 1997 and 1999, respectively, and the Ph.D. degree from Michigan State University, in 2009. He joined the Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea, in 2001. He has been involved in developing a data broadcasting system, multimedia networking, and immersive media. His research interests include digital signal processing, adaptive wireless video communications, and computer vision.

RIMSHA TARIQ received the B.S. degree in computer engineering from COMSATS University Islamabad, Lahore, Pakistan, in 2019, and the M.S. degree in computer engineering from the College of Electrical and Mechanical Engineering, NUST, in 2022. She is currently a Research Associate with the VISPRO Laboratories, Information Technology University. Her research interests include image processing, approximate computing, machine learning, and 3D reconstruction.

USAMA HASSAN received the B.S. degree from the University of Management and Technology, Lahore, Pakistan. He is currently a Research Assistant with the Intelligent Machines Laboratory, Information Technology University, Lahore. His research interests include scene understanding and 3D reconstruction. He is also working on point cloud upsampling and 3D reconstruction.

JAVED IQBAL received the B.S. and M.S. degrees from COMSATS University Islamabad, Pakistan. He is currently pursuing the Ph.D. degree with the Intelligent Machines Laboratory, Information Technology University, Lahore, Pakistan. His research interests include domain adaptation, self-supervised learning, 3D reconstruction, exploring domain adaptation for semantic segmentation, object detection varying from normal to adverse weather conditions, and ground imagery to satellite imagery. He is also working on point cloud upsampling and 3D reconstruction.

ABDUL BASIT received the B.S. degree in computer science from Information Technology University, Lahore, in 2022. He is currently pursuing the M.S. degree in computer science with the Lahore University of Management Sciences. He is also a Research Assistant with the Intelligent Machines Laboratory, Information Technology University. His research interests include multi-view geometry, 3D reconstruction, and scene understanding.

HYON-GON CHOO received the B.S. and M.S. degrees in electronic engineering and the Ph.D. degree in electronic communication engineering from Hanyang University, South Korea, in 1998, 2000, and 2005, respectively. He is currently a Principal Researcher with the Electronics and Telecommunications Research Institute, Daejeon, South Korea. He is also the Director of the Immersive Media Research Section. His research interests include video coding for machines, holography, multimedia protection, and 3D broadcasting technologies.

REHAN HAFIZ received the Ph.D. degree in electrical engineering from The University of Manchester, Manchester, U.K., in 2008.

He is currently with Information Technology University, Lahore, Pakistan, as a Professor of the Faculty of Engineering, where he founded and directed the Vision Processing Laboratory, which focuses on areas, such as vision system design, approximate computing, design of application-specific hardware accelerators, deep learning, FPGA-based design, and applied image and video processing. Apart from several publications in the above areas, he holds multiple patents in the U.S., South Korea, and Pakistan patent office.

MOHSEN ALI received the Ph.D. degree from the University of Florida. He is currently an Assistant Professor with Information Technology University (ITU) and the Co-Founder of the Intelligent Machines Laboratory. His current research interests include domain adaptation and self-supervised learning. He has been pursuing a multi-disciplinary problem of understanding urbanization and economic conditions by combining information from satellite imagery analysis and geospatial datasets. His work has been published in reputable computer vision and machine learning venues. He is a Fulbright alumnus and is also advising the Computer Vision and Machine Learning Group, ITU. He has been awarded the Google Research Scholar Award.