Frequency-Selective Mesh-to-Mesh Resampling for Color Upsampling of Point Clouds

Viktoria Heimann, Andreas Spruck and André Kaup
Multimedia Communications and Signal Processing
Friedrich-Alexander University, Erlangen-Nuremberg, Germany
{viktoria.heimann, andreas.spruck, andre.kaup}@fau.de

Abstract—With the increased use of virtual and augmented reality applications, the importance of point cloud data rises. High-quality capturing of point clouds is still expensive and thus, the need for point cloud super-resolution or point cloud upsampling techniques emerges. In this paper, we propose an interpolation scheme for color upsampling of three-dimensional color point clouds. As a point cloud represents an object’s surface in three-dimensional space, we first conduct a local transform of the surface into a two-dimensional plane. Secondly, we propose to apply a novel Frequency-Selective Mesh-to-Mesh Resampling (FSMMR) technique for the interpolation of the points in 2D. FSMMR generates a model of weighted superpositions of basis functions on scattered points. This model is then evaluated for the final points in order to increase the resolution of the original point cloud. Evaluation shows that our approach outperforms common interpolation schemes. Visual comparisons of the jaguar point cloud underlines the quality of our upsampling results. The high performance of FSMMR holds for various sampling densities of the input point cloud.

Index Terms—point cloud, color upsampling, frequency-selective

I. INTRODUCTION

Point clouds are an emerging three-dimensional data type and are often used in automated driving, virtual, and augmented reality. A point cloud is a set of three-dimensional coordinates with eventually assigned texture or color. Point clouds mainly show object surfaces in three-dimensional space. The points are not restricted to lie on integer coordinates. Furthermore, the points are stored unordered, i.e., no connection to the points’ neighborhood can be established from the stored point cloud. An example of such a color point cloud is given in Figure 1.

As the acquisition of point clouds is expensive, mainly low resolution point clouds are acquired. This is in contrast to the rendering for high resolution screens. Thus, super-resolution of point clouds becomes necessary. Another term for point cloud super-resolution that is mainly driven from the computer science community is point cloud upsampling. We will use both interchangeable in this remainder of this paper. Comparing super-resolution of point clouds to super-resolution of images, new challenges arise. As the points are allowed to obtain any coordinate in the three-dimensional space, first, the problem of geometry upsampling has to be solved. Amenta et al. [2] developed a common algorithm for reconstructing three-dimensional surfaces. Therefore, they compute a Voronoi diagram. In [3], Alexa et al. used point sets to represent shape. They add new points at the vertices of Voronoi diagrams that are built on moving least square surfaces. Data-driven approaches were developed in recent years with the rise of neural networks. Yu et al. developed PU-Net [4], the first network that upsamples point clouds. Its feature extraction is based on PointNet++ [5], which classifies and segments point clouds. Furthermore, the joint loss function aims to insert new points uniformly. In order to increase the performance of PU-Net especially for large resolution factors, the upsampling unit is applied recursively several times in MPU net [6]. Li et al. [7] presented another improvement of PU-Net with PU-GAN, a PU-Net that is embedded in a generator-discriminator structure in order to further increase the performance. With a focus on edge handling, Yu et al. presented EC-Net [8]. It aims to better handle sharp edges in point cloud objects by inserting more points in edge-like areas. As these networks mainly work patch-based, Zhang et al. [9] presented a data-driven approach for the upsampling of point clouds by incorporating the entire point cloud and thus, focusing more on the overall shape of the object than on the local patch shape.

Super-resolution of color point clouds requires color upsampling in a second step. Therefore, common interpolation methods like linear, cubic, and natural neighbor interpolation...
are incorporated. Dinesh et al. [10] presented a graph-based approach that is applicable to both geometry and color upsampling. They create a k-nearest neighbor graph to estimate coordinates and RGB-values. These are then refined by minimizing a graph total variation.

In this paper, we will focus on the color super-resolution of color point clouds. In the upcoming section, we propose a 3D to 2D coordinate transform and show our framework for color upsampling of color point clouds. Then, we continue with the presentation of our novel frequency-selective mesh-to-mesh resampling algorithm. In Section IV, we show and interpret our results. Finally, the paper closes with a conclusion in Section V.

II. COLOR UPSAMPLING

As described in Section I, color point cloud super-resolution requires a geometry and color upsampling step. In this section, we focus on the color upsampling part. We assume that the geometry upsampling is already conducted applying one of the presented algorithms [2]–[4], [6]–[10]. For color upsampling, we assume to have a set of originally known points carrying coordinate and color information. We denote the set of original points as $\mathcal{O}$. Furthermore, we assume the geometry upsampling algorithms to generate a set of points carrying solely coordinates. Color information of these points is not known yet. We denote the set of the to be reconstructed points as $\mathcal{R}$. The point cloud that we expect to input in our framework is the joint set $\mathcal{P} = \mathcal{O} \cup \mathcal{R}$. The point set $\mathcal{P}$ of a block of the Asterix point cloud is given in Figure 2. In the following, all computations are conducted locally on a block which is a $4 \times 4 \times 4$ cuboid from the three-dimensional point cloud. The colored and filled points are in $\mathcal{O}$ and show their respective RGB values. The blue circles denote the points in $\mathcal{R}$. Color information is not yet known for these points. Their coordinates are generated using a geometry upsampling technique before. We aim to reconstruct the color information of these points. On closer examination, it becomes obvious that a point cloud represents an object’s surface in three-dimensional space. As a surface is usually two-dimensional, we aim to transform the surface into a two-dimensional plane. The transform works block-based and consists of five steps. First, the Euclidean distance $w_c$ is determined between the points of the joint set of point cloud coordinates $\mathcal{P}$

$$w_c(p_i, p_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad \forall (p_i, p_j) \in \mathcal{P}.$$  

Knowing the euclidean distance between all points in the cloud, knowledge about neighborhood connections is established. Second, we further exploit this knowledge by creating a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertexes set $\mathcal{V}$ and edge set $\mathcal{E}$ with $(p_i, p_j) \in \mathcal{V}$ and $(p_i, p_j, w_c(p_i, p_j)) \in \mathcal{E}$. For the coordinate transform, we are interested in neighboring points. Thus, as a third step, a minimum spanning tree $\mathcal{M}$ is generated out of $\mathcal{G}$, i.e., $\mathcal{M} \subset \mathcal{G}$. Generating the minimum spanning tree enables us to calculate neighborhood relations. An extract of a minimum spanning tree is given in Figure 4. Fourth, we exploit the established neighborhood relations. As we want to incorporate the third dimension into the first two dimensions, we calculate the Euclidean distance in first and third, and in second and third dimension for neighboring vertexes of the minimum spanning tree which are representing point cloud points yielding

$$\Delta \tilde{r}_{ij} = \text{sgn}(x_j - x_i) \sqrt{(x_j - x_i)^2 + (z_j - z_i)^2},$$

Fig. 2: Block of the three-dimensional point cloud Asterix shown in Figure 1 before color upsampling. Filled points show their respective color values (here: varying from flesh-coloured to black and dark green). These colored points are the originally known points in point set $\mathcal{O}$. Color information has to be determined for the points denoted as blue circles belonging to the point set $\mathcal{R}$.

Fig. 3: The same extract of the point cloud Asterix as in Figure 2 before color upsampling and after transforming to 2D. The filled and colored points are the originally known points in point set $\mathcal{O}$. Color information has to be determined for the points denoted as blue circles belonging to the point set $\mathcal{R}$.
and

\[ \Delta \tilde{y}_{ij} = \text{sgn}(y_j - y_i) \sqrt{(y_j - y_i)^2 + (z_j - z_i)^2}. \] (3)

Using the sgn function, we keep the direction between the points in x- and y-direction, respectively. Fifth, we add (2) and (3) to the previous vertex in the graph starting at the randomly selected root node from the minimum spanning tree. Thus, the transformed coordinates \( \tilde{x}_j \) and \( \tilde{y}_j \) are determined

\[ \tilde{x}_j = \tilde{x}_i + \Delta \tilde{x}_{ij} \] (4)

\[ \tilde{y}_j = \tilde{y}_i + \Delta \tilde{y}_{ij}. \] (5)

Thus, we map the 3D coordinates into 2D space and incorporate the third dimension into the new coordinates. Thereby, we do not perform a projection that neglects the third dimension totally. The 3D to 2D transform is summarized in Figure 4. The exemplary extract of the Asterix point cloud in Figure 2 is shown after the transform in Figure 3. The transform keeps neighboring points from 3D next to each other in 2D as well. The additionally inserted points for upsampling are located within the set of originally known color values. Thus, a two-dimensional mesh-to-mesh resampling problem is formulated.

The resampling problem can be solved by incorporating common interpolation schemes such as linear and cubic interpolation. In addition, we present a model-based approach called frequency-selective mesh-to-mesh resampling in the upcoming section. It aims to generate the missing color values \( \mathcal{R} \). Finally, these have to be assigned to the according point in three-dimensional space. Hence, this yields the upsampled color point cloud.

### III. FREQUENCY-SELECTIVE MESH-TO-MESH RESAMPLING

In this section, we introduce the novel Frequency-Selective Mesh-to-Mesh Resampling (FSMMR) for color upsampling of point clouds. As described in Section II color upsampling of point clouds requires an interpolation that can create new color values at arbitrary points, which we will refer to as mesh in the following, from original points at other mesh positions. We refer to the set of original mesh points as \( \mathcal{O} \) and to the set of to be reconstructed points as \( \mathcal{R} \). Both sets are shown exemplary in Figure 6.

Fig. 5: Summary of the proposed framework. 3D low-resolution point cloud \( \mathcal{O} \) and upsampled geometry points \( \mathcal{R} \)

- Calculate Euclidean distance
- Create graph \( \mathcal{G} = \{\mathcal{V}, \mathcal{E}\} \)
- Create minimum spanning tree \( \mathcal{M} \subset \mathcal{G} \)
- Calculate \( \Delta \tilde{x}_{ij} \) and \( \Delta \tilde{y}_{ij} \)
- Calculate \( \tilde{x}_j \) and \( \tilde{y}_j \)
- Calculate residual \( r^{(\nu)}[m,n] \)
- Calculate residual energy decrease \( \Delta \mathcal{E}^{(\nu)} \) for every basis function
- Selection of best fitting basis function
- Stopping criterium met?
- Yes
- Obtain signal in \( \mathcal{R} \) following (12)
- Assign reconstructed values to \( \{x,y,z\} \in \mathbb{R}^3 \)
- Generated model \( g^{(\nu)}[m,n] \)
- No

3D low-resolution point cloud \( \mathcal{O} \) and upsampled geometry points \( \mathcal{R} \)

Fig. 4: An extract of the minimum spanning tree is shown in the left image. The graph is transformed into 2D incorporating new coordinates \( \tilde{x} \) and \( \tilde{y} \) in the right graph.
Research has shown that image signals can be represented in terms of weighted superpositions of basis functions. This assumption is used for the extrapolation into unknown image areas in Frequency-Selective Extrapolation [11], to solve irregular sampling problems using Frequency-Selective Reconstruction [12], and for resampling from originally known mesh positions onto unknown grid points in Frequency-Selective Mesh-to-Grid Resampling [13], [14]. For color upsampling of point clouds, the surface of a three-dimensional object has to be upsampled. We assume, that the two-dimensional positions onto unknown grid points in Frequency-Selective Extrapolation [11], to solve image rejection into unknown image areas in Frequency-Selective Extrapolation [11], to solve irregular sampling problems using Frequency-Selective Reconstruction [12], and for resampling from originally known mesh positions onto unknown grid points in Frequency-Selective Mesh-to-Grid Resampling [13], [14]. For color upsampling of point clouds, the surface of a three-dimensional object has to be upsampled. We assume, that the two-dimensional areas in Frequency-Selective Extrapolation [11], to solve image rejection into unknown image areas in Frequency-Selective Extrapolation [11], to solve irregular sampling problems using Frequency-Selective Reconstruction [12], and for resampling from originally known mesh positions onto unknown grid points in Frequency-Selective Mesh-to-Grid Resampling [13], [14]. For color upsampling of point clouds, the surface of a three-dimensional object has to be upsampled. We assume, that the two-dimensional

\begin{equation}
\varphi(k,l) = \sum_{k,l \in \mathbb{K}} c_{k,l}\varphi_{k,l}(m,n),
\end{equation}

where \( m \in \mathbb{R} \cup \mathbb{O} \) and \( n \in \mathbb{R} \cup \mathbb{O} \) denote the points’ arbitrary coordinates in horizontal and vertical direction, respectively. The indexes \( k \in \mathbb{N} \) and \( l \in \mathbb{N} \) denote the frequency indexes in horizontal and vertical direction, respectively, from the set of available basis functions \( \mathbb{K} \). The assigned expansion coefficient \( c \) can be interpreted as transform coefficient of the inverse transform as basis functions from the discrete cosine transform (DCT) are incorporated. The aim in FSMMR is to create a model \( g[m,n] \) that approaches (6). The model is generated block-wise and iteratively on the set of original points \( \mathbb{O} \)

\begin{equation}
g^{(\nu)}[m,n] = g^{(\nu-1)}[m,n] + \hat{c}_{u,v}\varphi_{u,v}[m,n],
\end{equation}

where \( \nu \) denotes the current iteration number and \( \hat{c}_{u,v} \) the estimated expansion coefficient of the selected frequency coefficients \( u \in \mathbb{N} \) and \( v \in \mathbb{N} \) in the current iteration. The model is initialized to zero, i.e. \( g^{(0)} = 0 \). The model’s aim is to meet the original signal \( f[m,n] \) as close as possible. Thus, the residual between both is determined

\begin{equation}
r^{(\nu)} = f[m,n] - g^{(\nu)}[m,n].
\end{equation}

As the residual has to be reduced, a weighted residual energy is defined

\begin{equation}
E^{(\nu)} = \sum_{(m,n)} w[m,n] \left( r^{(\nu)}[m,n] \right)^2.
\end{equation}

with a spatial weighting function \( w[m,n] \) that favors center points for the model generation. Furthermore, research has shown that natural images are mainly composed of low frequency basis functions [15]. We assume that this holds for objects’ surfaces as well. Hence, we incorporate the frequency weighting function

\begin{equation}
w_f[k,l] = \sigma^{\sqrt{k^2+l^2}},
\end{equation}

where \( \sigma \in [0,1] \) parametrizes the decay, into the selection of the best fitting basis function in iteration \( \nu \)

\begin{equation}
(u,v) = \arg \max_{(k,l)} \left( \Delta E^{(\nu)}_{k,l} w_f[k,l] \right).
\end{equation}

In every iteration, the basis function is selected that maximizes the residual energy reduction as it will close the gap between model and original signal the most. These steps are repeated until a stopping criterion such as a number of iterations or a minimal residual energy is met. Finally, the model is evaluated for the reconstructed points in \( \mathbb{R} \). Thus, the generated estimated expansion coefficients \( \hat{c}_{k,l} \) are multiplied by the basis functions \( \varphi_{k,l} \)

\begin{equation}
f[a,p] = \sum_{k,l \in \mathbb{K}} \hat{c}_{k,l}\varphi_{k,l}[a,p],
\end{equation}

at the newly inserted mesh points \( [a,p] \in \mathbb{R}^2 \in \mathbb{R} \).

A summary of the proposed color upsampling scheme incorporating FSMMR is shown as a flow graph in Figure 5.

IV. EXPERIMENTAL RESULTS

We conduct extensive experiments in order to show the performance of our proposed color upsampling scheme. We denote our proposed geometry transform from three-dimensional to two-dimensional surface as 2D approach described in Section II. For the upcoming interpolation, we compare linear
(LIN2), cubic (CUB2), natural neighbor interpolation (NAT2), and our proposed FSMMR. Furthermore, we evaluate the reconstruction quality without the proposed transformation and apply linear (LIN3) and natural neighbor (NAT3) interpolation directly to the three-dimensional point cloud. Thus, we refer to this approach as 3D.

In order to evaluate the quality of the reconstructed point clouds, we first downsample the original ones randomly. The selected points are taken as low-resolution point cloud and are handed over to the proposed algorithm as original point set \( O \).

As we focus on color upsampling in this paper, the coordinates of the skipped points are kept in order to reconstruct the missing color values at the right positions. These points form the point set \( R \). This approach enables us to evaluate the final result in terms of peak signal-to-noise ratio (PSNR). We measure color PSNR conducting two steps. First, the PSNR is determined for the three color channels R, G, and B separately. Second, the average of the independently calculated PSNR values on each color channel is taken as color PSNR. We conducted every experiment for three runs in order to exclude effects that might occur due to the random selection of low-resolution points. All results that we show in the following are averaged values for the three runs. In one run, every interpolation scheme receives the same downsampled point cloud. Thus, the presented results are fully comparable.

The reconstruction quality of the point clouds of the 3D Color Mesh dataset [1] for a sampling ratio of 50% is shown in Table I in terms of reconstruction PSNR in dB. The reconstruction PSNR measures the PSNR solely for the reconstructed points. Original points are not incorporated in the quality measurement. Best performances are highlighted in bold font. Table I demonstrates that our proposed approach incorporating FSMMR performs best for most point clouds. If FSMMR is not the best performing method such as for the CableCar point cloud, the three-dimensional approaches, LIN3 and NAT3, yield the highest reconstruction quality. The two-dimensional approaches using common interpolation schemes never yield best result. Thus, the transform from 3D to 2D can be neglected as reason for the high quality of our proposed FSMMR scheme.

Furthermore, we evaluated the performance of the 3D Color Mesh dataset for various sampling densities. A sampling density of 10% means that only 10% of the original points are kept and used for the reconstruction of the point cloud. Thereby, we simulate various enlargement factors of the point cloud. Thus, it is justified to expect smaller PSNR values for smaller sampling densities as more points have to be reconstructed based on less data. The averaged reconstruction PSNR for sampling densities from 10% to 80% is given in Figure 7. The PSNR is averaged for the entire 3D Color Mesh dataset. As expected, the higher the sampling density, the better the reconstruction result for the point clouds. Moreover, the figure demonstrates that FSMMR is the best performing technique for all sampling densities.

A visual example for the reconstruction quality is given for the Jaguar point cloud in Figure 8. In Figure 8a, the original point cloud is presented. In Figure 8b, the low-resolution point cloud which is the starting point for the upsampling algorithms is shown. In Figure 8c and 8d, the final point cloud is shown for LIN3 and NAT3 methods, respectively. Both point clouds show missing values in the region of the head of the jaguar, especially the left ear, the nose and the region above the eyes are affected by that. This effect originates in the definition of linear and natural neighbor interpolation. In the border regions pixel values have to be extrapolated. A classical interpolation scheme cannot fulfill this requirement. The 2D approaches LIN2, CUB2, and NAT2 in Figure 8e, 8f, and 8g, respectively, show missing values along the block borders that are used for the proposed 3D to 2D transform. The number of missing pixels increases due to the growing number of borders as a boarder occurs for every block. The result for our proposed FSMMR approach is shown in Figure 8h. No missing pixels can be detected, nor at the head of the jaguar nor at block borders within the point cloud although the same block structures are used for FSMMR as well as for LIN2, CUB2, and NAT2. This demonstrates that FSMMR can conduct interpolation and extrapolation in one step. A slight color shift into green can be detected at the mouth of the jaguar for FSMMR.

| Point Cloud | 3D | 2D |
|-------------|----|----|
| NineTails   | 23.7 | 24.6 |
| Asterix     | 27.2 | 28.1 |
| CableCar    | 25.3 | 26.8 |
| Dragon      | 26.7 | 27.3 |
| Duck        | 13.6 | 14.5 |
| GreenDinosaur | 25.3 | 26.8 |
| GreenMonstre | 25.5 | 26.6 |
| Horse       | 23.6 | 24.5 |
| Jaguar      | 21.6 | 22.9 |
| LongDinosaur | 21.9 | 23.4 |
| Man         | 30.3 | 31.2 |
| Mario       | 23.4 | 24.5 |
| MarioCar    | 23.1 | 24.2 |
| PokemonBall | 8.9  | 9.6  |
| Rabbit      | 21.0 | 21.9 |
| RedHorse    | 22.9 | 24.0 |
| Statue      | 24.3 | 25.9 |

**TABLE I: Results for all point clouds from the 3D Color Mesh dataset in terms of reconstruction PSNR in dB. Best qualities are given in bold.**

**V. Conclusion**

In conclusion, we propose to use the frequency-selective mesh-to-mesh resampling technique for color upsampling of 3D color point clouds. As we assume the point cloud to represent a surface, we transform it into two-dimensional space. The transform enables us to apply FSMMR for color upsampling of 3D point clouds properly. The extensive evaluation shows that our proposed method works best for all sampling densities, i.e., enlargement factors. The visual comparison shows that only slight color shifts can occur using FSMMR, but all desired points are reconstructed and no missing pixels occur in border regions of blocks.
ACKNOWLEDGMENT

The authors gratefully acknowledge that this work has been supported by the Deutsche Forschungsgemeinschaft (DFG) under contract number KA 926/8-1.

REFERENCES

[1] Anass Nouri, C. Charrier, and O. Lézoray, “Technical report : Grey 3D colored mesh database,” Technical report, Normandie Université, Unicaen, EnsiCaen, CNRS, GREYC UMR 6072, Jan. 2017.

[2] Nina Amenta, M. Bern, and M. Kamvysselis, “A New Voronoi-Based Surface Reconstruction Algorithm,” in Proceedings of the 25th annual conference on Computer graphics and interactive techniques, July 1998, SIGGRAPH, pp. 415–421.

[3] Marc Alexa, J. Behr, D. Cohen-Or, S. Fleishmann, D. Levin, and C. T. Silva, “Point Set Surfaces,” in IEEE Visualization, 2001.

[4] Lequan Yu, X. Li, C. Fu, D. Cohen-Or, and P. Heng, “PU-Net: Point Cloud Upsampling Network,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2790–2799.

[5] Charles Ruizhongtai Qi, L. Yi, H. Su, and L. J. Guibas, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” in Advances in Neural Information Processing Systems, 2017, vol. 30.

[6] Wang Yifan, S. Wu, H. Huang, D. Cohen-Or, and O. Sorkine-Hornung, “Patch-based Progressive 3D Point Set Upsampling,” in IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5951–5960.

[7] Ruihui Li, X. Li, C. Fu, D. Cohen-Or, and P. Heng, “PU-GAN: A Point Cloud Upsampling Adversarial Network,” in Proceedings of the IEEE International Conference on Computer Vision, October 2019.

[8] Lequan Yu, X. Li, C. Fu, D. Cohen-Or, and P. Heng, “EC-Net: An Edge-Aware Point Set Consolidation Network,” in Computer Vision – ECCV 2018, 2018, pp. 398–414.

[9] Wentai Zhang, H. Jiang, and Zhang, “Data-driven Upsampling of Point Clouds,” Computer-Aided Design, vol. 112, pp. 1–13, 2019.

[10] Chinthaka Dinesh, G. Cheung, and I. V. Bajić, “Super-Resolution of 3D Color Point Clouds Via Fast Graph Total Variation,” in IEEE International Conference on Acoustics, Speech and Signal Processing, 2020, pp. 1983–1987.

[11] André Kaup, K. Meisinger, and T. Aach, “Frequency selective signal extrapolation with applications to error concealment in image communication,” in International Journal of Electronics and Communications, June 2005, vol. 59, pp. 147–156.

[12] Michael Schoberl, J. Seiler, S. Foessel, and A. Kaup, “Increasing Imaging Resolution by Covering Your Sensor,” in 18th IEEE International Conference on Image Processing, 2011, pp. 1897–1900.

[13] Ján Koloda, J. Seiler, and A. Kaup, “Frequency-Selective Mesh-to-Grid Resampling for Image Communication,” in IEEE Transactions on Multimedia, August 2017, vol. 19, pp. 1689–1701.

[14] Viktoria Heimann, Nils Genser, and André Kaup, “Key Point Agnostic Frequency-Selective Mesh-to-Grid Resampling using Spectral Weighting,” in Proceedings of the IEEE 22nd Workshop on Multimedia Signal Processing, 2020.

[15] Jürgen Seiler, M. Jonscher, M. Schöberl, and A. Kaup, “Resampling Images to a Regular Grid From a Non-Regular Subset of Pixel Positions Using Frequency Selective Reconstruction,” in IEEE Transactions on Image Processing, November 2015, vol. 24, pp. 4540–4555.