https://doi.org/10.2352/EI.2022.34.17.3DIA-224

Feature-Driven 3D Range Geometry Compression via Spatially-Aware Depth Encoding

Broderick S. Schwartz, Matthew G. Finley, and Tyler Bell* [*]

Department of Electrical and Computer Engineering, University of Iowa; Iowa City, Iowa 52242, USA
[*] ***tyler-bell@uiowa.edu***

Abstract

The ever-growing variety of capture methods and applications for 3D range geometry continually increases the need for efficient methods of data storage and transmission. Compression techniques address this need by offering reduced file sizes while maintaining the precision needed for a particular application. Several such compression methods use phase-shifting principles to encode the 3D data into a 2D RGB image. In some applications, such as telepresence, high precision may only be required in a particular region within a scan. This paper proposes a feature-driven compression method that provides a way to encode regions of interest at higher precision while encoding the remaining data at lower precision to reduce file sizes. This method supports both lossless and lossy compression, enabling even greater file size savings and a wider range of applications. In the case of a depth scan of a bust, an extracted bounding box of the face was used to create an encoding distribution such that the facial region was encoded at higher precisions. When using JPEG 80, the global RMS reconstruction accuracy of this novel encoding was 99.72%; however, in the region of interest, the accuracy was 99.88%. This feature-driven encoding achieved a 26% reduction in compressed file size compared to a fixed, high precision encoding.

Introduction

Advances in optical and computing technologies have allowed many subsequent improvements to be made in the field of three-dimensional (3D) range scanning. Range-scanning systems are now capable of capturing high-precision 3D range data at speeds much faster than real-time (i.e., 30hz or 60hz) [1]. These increases in fidelity and acquisition speed bring with them an increase in the quantity of raw data being generated by such systems. To enable the more effective use of 3D range geometry for visualization and manipulation, this data must be compressed. One typical method of compressing 3D range data is to sinusoidally encode its depth information within the three color channels of a conventional 2D RGB image [2, 3, 4, 5]. This encoded depth information may then be additionally compressed using well-established image compression standards (i.e., PNG or JPEG, depending on the method), further reducing file sizes. Several image-based, depth compression algorithms, such as multiwavelength depth [5], employ a user-defined parameter, fringe width (P), that determines the frequency used to encode the depth information into the output image and is easy to transmit as overhead. In general, if the value of this fringe width is decreased, the encoding precision and file size increase; conversely, as the fringe width is increased, encoding precision and file size generally decrease. Conventionally, a constant fringe width parameter is defined for an entire encoding. However, as first described in the Variable Precision Depth (VPD) method [6, 7], this fringe width can be set on a pixel-by-pixel basis, determining the encoding precision for each pixel independently from the rest of the data. One crucial aspect of this pixel-wise fringe width map, $P(i,j)$, is that it must be precisely known both at the time of encoding and decoding for the depth to be reconstructed correctly. Precise determination of the map could be accomplished by sending an additional image alongside the depth encoding, but this would significantly increase the overall file size. The transmission of an entire additional image was avoided by creating the fringe width map, $P(i,j)$, from a simple transformation of the normalized depth map that is stored as the third channel of the three-channel encoding produced by this method. One proposed method to create this fringe map utilized a normal distribution fitted to this third channel. The normal distribution was used to determine the per-pixel fringe width according to the distribution of the pixels within the scan's depth range (depth ranges containing more points were encoded with greater precision and vice versa). As mentioned, precise knowledge is required to re-derive this distribution on the decoding side making the method strictly dependent upon the *lossless* transmission of the base encoding. If the encoding is compressed using lossy formats (i.e., JPEG or one of several video compression methods), then the synthesis of the fringe width map cannot be perfectly reproduced on the decoding side as values in the normalized depth map will have changed. Without a precise fringe width mapping the compressed 3D range geometry cannot be accurately decoded. The proposed method adapts and expands the approach set forth in VPD [6, 7] by enabling feature-driven compression of 3D range data, compatible with lossy file formats, for use in a variety of applications such as telecommunications, telepresence, and entertainment.

The proposed approach allows regions of interest to be well-preserved within a scene while reducing the overall file size of the encoded output. Targeted precision can be achieved through feature detection (or selection) within a 3D range scan, and then generating a varying fringe width distribution map focused upon that feature, or features, of interest (e.g., face, eyes, etc.). When this feature-driven precision approach was used to compress a depth scan of a bust using JPEG 80, the resulting reconstruction had a global RMS accuracy of 99.72%; while, in the region of interest, the accuracy was 99.88%. As points outside of the region of interest were stored at lower precisions, file size savings were achieved. For the same depth scan of a bust, the proposed method gave greater than 26% file savings over an encoding where a fixed, high precision was used for the entire encoding. If some consideration is given to the manner in which this distribution is generated (i.e., smoothly varying between fringe widths), the method is compatible with both lossless and lossy compression standards, allowing high compression ratios to be achieved while maintaining fidelity in the targeted regions. The remainder of this paper will describe the proposed method, present experiments demonstrating its performance, and conclude with a summary.

Principle

Multiwavelength Depth Encoding

The proposed method is based on a phase-encoding technique called multiwavelength depth encoding (MWD) [5]. The MWD method encodes a floating-point depth map into the three color channels of a conventional RGB image by applying three different transformations to the depth map. Two of these operations are sinusoidal modulations and the third is a scaling of the depth map. The transformations can be described as

$$I_1(i,j) = \frac{1}{2} + \frac{1}{2}\sin\left(2\pi \times \frac{Z(i,j)}{P}\right), \tag{1}$$

$$I_2(i,j) = \frac{1}{2} + \frac{1}{2}\cos\left(2\pi \times \frac{Z(i,j)}{P}\right), \tag{2}$$

$$I_3(i,j) = \frac{Z(i,j) - \min(Z)}{\text{Range}(Z)}, \tag{3}$$

where Z is the depth information being encoded, P is the user-defined fringe width that specifies the encoding frequency, and I_1, I_2, and I_3 are the encodings of the depth information. Together I_1, I_2, and I_3 comprise the encoded image and are typically stored as the red, green, and blue channels of the encoded image, respectively. Often, P is defined by the user with a more convenient value that describes the number of encoding periods (n_{str}) by $P = \text{Range}(Z)/n_{str}$. The MWD method makes use of a constant value for P (equivalently a constant n_{str}), encoding with the same precision throughout the entire scan. If additional detail is required in a particular region, the entire scan must be encoded at that increased precision, and higher precision comes at the cost of increased file size. This trade-off can be particularly disadvantageous when portions of the scan outside of the region of interest are unimportant and do not need to be encoded with high precision. The proposed method addresses this type of situation by introducing a spatially-aware, variable precision encoding approach that is able to encode different portions of a scan with different levels of precision using an encoding period map $N_{str}(i,j)$ which varies pixel-by-pixel.

Spatially-Aware Depth Encoding

To encode 3D range data in a feature-driven manner using the proposed method, the desired feature(s) of interest must first be identified within the data that is to be compressed. The selection of such features is application dependent and may be decided at the time of compression. Any approach may be used for selecting a feature of interest, from arbitrary selection to automated feature detection. A hemisphere is used as an example and is shown in Fig. 1(a). An example feature of interest is chosen arbitrarily as a region of high gradient, as shown in Fig. 1(b). After the pixel-space bounding box of the feature (or the pixel coordinate itself) has been determined, a distribution map centered on the region of interest, $D(i,j)$, must be calculated. This distribution drives the number of periods, and thus the precision, with which each pixel is encoded. It should be noted that the distribution map, $D(i,j)$, can represent any arbitrary distribution and may be freely determined by the user at the time of compression based on the regions of interest to be preserved by the target application.

In the example, the center point of the region of interest was used to first generate a global distance image $E(i,j)$ (calculated using the Euclidean distance). This distance image was then manipulated to create a distribution map, $D(i,j)$, with the desired properties. Here, that was a plateau of high precision (where the distribution is valued at or near one) that quickly falls off away from the feature of interest (to a value near zero). This was accomplished by first applying a simple two-parameter function

$$E'(i,j) = \left(\alpha - \frac{E(i,j)}{max(E)}\right)^{\beta}, \tag{4}$$

where $\alpha = 1.1$ and $\beta = 7$. A threshold operation was then performed to produce a distribution image valued between zero and one:

$$D(i,j) = \begin{cases} E'(i,j) & \text{if } E'(i,j) \leq 1 \\ 1 & \text{if } E'(i,j) > 1. \end{cases} \tag{5}$$

Lastly, a Gaussian filter was applied using MATLAB's $imgaussfilt()$ function with $\sigma = 50$. Regardless of how it is generated, this distribution (which lies between zero and one), must then be scaled to lie between two user-defined values. These values, n_{min} and n_{max}, represent the range of encoding periods—and thus the range of encoding precisions—that is used to compress the depth data. This scaling is performed as

$$N_{str}(i,j) = D(i,j) \times (n_{max} - n_{min}) + n_{min}. \tag{6}$$

An example of this encoding period map, $N_{str}(i,j)$, is shown as Fig. 1(c). This $N_{str}(i,j)$ may then be used to encode the depth information, pixel-by-pixel, into the color channels of a standard 2D RGB image. Encoding is accomplished by employing Eqs. (1)-(3) after some small re-configuring to make use of $N_{str}(i,j)$ in place of P as

$$I_1(i,j) = \frac{1}{2} + \frac{1}{2}\sin\left(2\pi \times \frac{Z(i,j)}{\text{Range}(Z)} \times N_{str}(i,j)\right), \tag{7}$$

$$I_2(i,j) = \frac{1}{2} + \frac{1}{2}\cos\left(2\pi \times \frac{Z(i,j)}{\text{Range}(Z)} \times N_{str}(i,j)\right), \tag{8}$$

$$I_3(i,j) = \frac{Z(i,j) - \min(Z)}{\text{Range}(Z)}. \tag{9}$$

Above, Z is the depth map being encoded. The three signals I_1, I_2, and I_3 are stored in the three channels of a color image (typically the red, green, and blue channels, respectively), as shown in Fig. 1(d). This resulting image encoding may then be compressed using either lossless (i.e., PNG) or lossy (i.e., JPEG) file formats and either stored or transmitted.

Along with the image, it is necessary to store or send a few extra parameters that are used to perfectly reconstruct $N_{str}(i,j)$ at the time of decoding. In the example, where a single feature

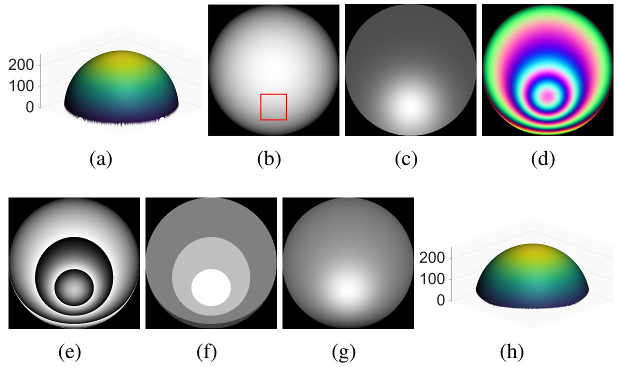

Figure 1. The proposed method of feature-driven 3D range geometry compression applied to a 512 × 512 pixel depth map of a hemisphere with a 256 mm depth range. (a) 3D rendering of the hemisphere to be compressed; (b) depth map of (a) with the selected feature highlighted; (c) $N_{str}(i,j)$, which lies between $n_{min} = 1$ and $n_{max} = 6$; (d) the encoded 2D image produced by this method which can be stored with lossless or lossy compression; (e) the wrapped phase ϕ_{HF} extracted from (d); (f) the stair map K used to unwrap ϕ_{HF}; (g) the unwrapped absolute phase Φ; (h) 3D reconstruction from (d) when stored with PNG.

of interest is used, these parameters are: the pixel coordinate of the feature of interest, the two-parameters (α and β) of the function used to generate the region of focus, the size of the smoothing filter to generate the distribution $D(i,j)$ (σ), and the minimum (n_{min}) and maximum (n_{max}) encoding periods. Altogether this requires six additional parameters beyond those required by the constant precision encoding. Even if these parameters were stored at floating-point precision, this would add only 24 bytes of additional overhead.

To decode the depth information, the encoding period map, $N_{str}(i,j)$, must be regenerated using the additional parameters. Next, the wrapped high frequency phase ϕ_{HF} (Fig. 1(e)) must be recovered from the encoded image's channels by

$$\phi_{HF}(i,j) = \tan^{-1}\left(\frac{I_1(i,j) - 0.5}{I_2(i,j) - 0.5}\right). \tag{10}$$

The arctangent function is range limited, thus, the high frequency phase that has been calculated contains periodic discontinuities which must be corrected. This correction (or phase unwrapping) process requires that the low frequency phase ϕ_{LF} first be retrieved from the blue channel of the encoded image by

$$\phi_{LF} = I_3(i,j) \times 2\pi - \pi. \tag{11}$$

The low frequency phase is then used to determine the location and magnitude of the discontinuities that must be corrected in ϕ_{HF} to unwrap it. This mapping is called the stair map, K, (Fig. 1(f)) and is computed by

$$K(i,j) = \text{Round}\left(\frac{\phi_{LF}(i,j) \times N_{str}(i,j) - \phi_{HF}(i,j)}{2\pi}\right). \tag{12}$$

The unwrapped absolute phase may then be calculated by

$$\Phi(i,j) = \phi_{HF}(i,j) + 2\pi \times K(i,j) \tag{13}$$

The depth information can then be recovered from Φ (Fig. 1(g)) via

$$Z(i,j) = \frac{\Phi(i,j) \times \text{Range}(Z)}{2\pi \times N_{str}(i,j)}. \tag{14}$$

This recovered Z (Fig. 1(h)) is the reconstructed depth information in the original depth range where the accuracy of each pixel is determined by $N_{str}(i,j)$.

Experiments

The performance of the proposed feature-driven encoding method is first demonstrated by applying it to an ideal hemisphere (with a size of 512 × 512 pixels, a depth range of 256 mm, and a raw file size of 1.05 MB). The feature of interest was the same arbitrary point that was selected in Fig. 1, and $N_{str}(i,j)$ was created using the same procedures as in Eqs. (4)-(6) with $\alpha = 1.1$ and $\beta = 7$. The proposed method, encoding the scan with a varying encoding period, is compared against results produced by two, constant encoding period (and thus encoding precision) cases. In this experiment, all encodings were compressed using the JPEG 80 file format. The 3D renderings of the reconstructed hemisphere from each encoding are shown in Fig. 2. For each encoding approach, the chosen feature of interest is highlighted and shown at increased scale for easier comparison of the surface renderings. The rendering of the reconstruction produced by a low constant encoding period of $n_{str} = n_{min} = 1$ is shown in Fig. 2(a). The rendering of the reconstruction produced by the proposed method with a varying encoding period ($n_{str} = [1,6]$) is shown in Fig. 2(b). The rendering of the reconstruction produced by a high constant encoding period of $n_{str} = n_{max} = 6$ is shown in Fig. 2(c). The highlighted region of interest in Figs. 2(a)-2(c) are enlarged and shown as Figs. 2(d)-2(f), respectively.

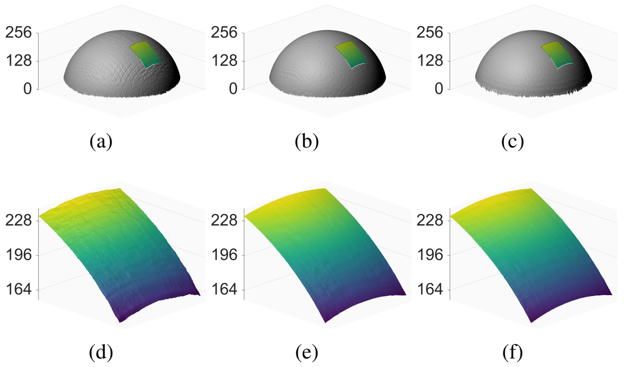

Figure 2. Surface renderings of an ideal hemisphere, of size 512 × 512 and original depth range 256 mm, compressed using the proposed feature-driven encoding and two constant precision encodings with the region of interest highlighted. All encodings were stored using JPEG 80. (a) Surface rendering of the low, constant precision encoding when $n_{str} = 1$; (b) surface rendering of the proposed method when the encoding precision varies from $n_{str} = [1,6]$ following a distribution centered on the highlighted region of interest; (c) surface rendering of the high, constant precision encoding when $n_{str} = 6$; (d)-(f) the regions of interest within surface renderings (a)-(c), respectively.

Absolute error images are presented for the same three encodings in Fig. 3. These error images were created by taking

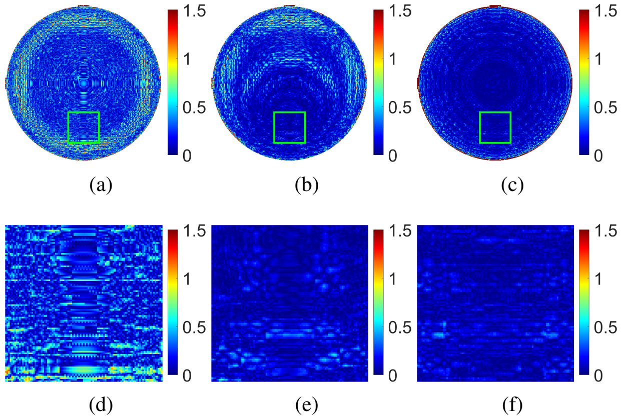

Figure 3. Absolute error images (in mm) of the proposed feature-driven encoding and two constant precision encodings when all approaches were applied to a 512 × 512 hemisphere with a depth range of 256 mm and stored using JPEG 80. (a) Absolute error image of the low constant precision encoding when $n_{str} = 1$; (b) absolute error image of the proposed method when the encoding precision varies from $n_{str} = [1, 6]$ following a distribution centered on the region of interest; (c) absolute error image of the high constant precision encoding when $n_{str} = 6$; (d)-(f) the region of interest within error images (a)-(c), respectively.

the absolute difference between each reconstruction and the original ideal hemisphere. Figure 3(a) shows the absolute error image produced by a constant encoding period of $n_{str} = n_{min} = 1$. Figure 3(b) shows the absolute error image produced by the proposed method with $n_{str} = [1, 6]$ according to the generated $N_{str}(i, j)$. Figure 3(c) shows the absolute error image produced by a constant encoding period of $n_{str} = n_{max} = 6$. Figures 3(d)-3(f) show enlarged versions of the highlighted portions of Figs. 3(a)-3(c), respectively. A cursory inspection shows that the region of interest is recovered with practically identical precision in both the high, constant precision approach and the proposed variable precision approach, while the low, constant precision approach exhibits much greater errors in the same region.

The low, constant encoding period resulted in a global RMS reconstruction accuracy of 99.51% (1.25 mm of RMS error in the original depth range of 256 mm) and an accuracy of 99.87% (0.32 mm of error) within the region of interest. The proposed, varying encoding period approach had a global RMS accuracy of 99.58% (1.09 mm of error) with an accuracy of 99.95% in the region of interest (0.13 mm). The high, constant encoding period approach had a global RMS accuracy of 99.84% (0.40 mm) with an accuracy of 99.96% in the region of interest (0.10 mm). It should be noted that for each reconstruction, large outliers (defined as spikes of greater than 25 mm) have been identified and excluded from these error calculations. In practice, these erroneous pixels are generally caused by phase unwrapping errors in regions of high gradient (e.g., sharp edges), and can often be easily identified and discarded or filtered.

Table 1. Results comparing reconstructions of a 512 × 512 pixel hemisphere (with a depth range of 256 mm and raw file size of 1048.58 KB) from low and high constant period and varying period encodings compressed with JPEG 80. The first column shows the global RMS error and accuracy values. The second column shows the RMS error and accuracy values for the region of interest. The third column shows the file sizes for each approach.

JPEG 80 Results	**Global RMS Error / Accuracy**	**ROI RMS Error / Accuracy**	**File Size (KB)**
Low Constant ($n_{str} = 1$)	1.25 mm / 99.51%	0.32 mm / 99.87%	18.42
Proposed ($n_{str} = [1, 6]$)	1.09 mm / 99.58%	0.13 mm / 99.95%	27.48
High Constant ($n_{str} = 6$)	0.40 mm / 99.84%	0.10 mm / 99.96%	44.22

The visual similarity (as seen in Fig. 2) between the high, constant encoding period (Fig. 2(f)) and the proposed, varying encoding period method (Fig. 2(e)) in the region of interest gives qualitative support to these quantitative results. By design, the error and accuracy metrics are most similar between the high, constant encoding period results and the proposed, varying encoding period results within the region of interest. Outside of the region of interest, the results of the low, constant encoding period approach and the proposed method more closely align. The file sizes of the encoded images follow a similar trend, with the high fixed precision encoding requiring a file size of 44.22 KB; the proposed, feature-driven, variable precision encoding requiring 27.48 KB; and the low fixed precision encoding yielding a file size of 18.42 KB. The proposed method delivers accuracy comparable to the high precision approach within the region of interest, while globally reducing overall encoded file sizes. These numeric results are assembled in Table 1.

An additional experiment is presented where a more complex data set was compressed using the two constant encoding periods and the proposed varying encoding period approach. The more complex data that was selected is a scan of a bust of former U.S. President Hayes [8]. The scan has a resolution of 892 × 718 pixels, a depth range of 477.4 mm, and a raw file size of 2.56 MB. The feature of interest was selected as the face of the bust. MATLAB's face detection algorithm was used to locate the center point of the face which was then used to create the $N_{str}(i, j)$ as described in Eqs. (4)-(6), where in this case $\alpha = 1.2$ and $\beta = 7$. Each of the encoding approaches were stored using the JPEG 80 file format.

Only the absolute difference images—calculated by taking the absolute difference between each reconstruction and the original data—are shown for this case. The left column of Fig. 4 (Figs. 4(a) and 4(d)) shows the absolute difference images when a low constant period encoding with $n_{str} = n_{min} = 1$ was used. The middle column of Fig. 4 (Figs. 4(b) and 4(e)) shows the absolute difference images when the proposed method's encoding period was varied between $n_{min} = 1$ and $n_{max} = 6$ following the custom distribution, $N_{str}(i, j)$, generated using the extracted face as described above. The outcome after reconstructing an encoding that used a high constant encoding period, $n_{str} = n_{max} = 6$, is shown in the right column of Fig. 4 (Figs. 4(c) and 4(f)). These results help to visually confirm the previously demonstrated relationship between the number of encoding periods used and the encoding precision of the output. Where the proposed method (middle column) has been encoded with a greater number of encoding periods (the region surrounding the face), the error image more closely resembles the higher, constant precision result (right column). Outside of the region of interest, the proposed method's

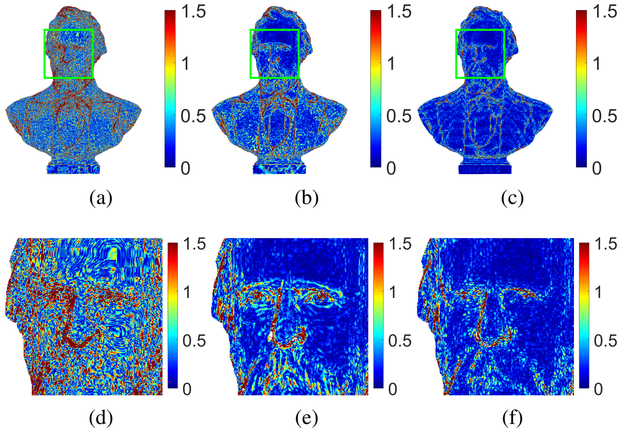

Figure 4. Absolute error images (in mm) produced when an 892 × 718 scan of former U.S. President Hayes, with a depth range of 477.4 mm and a raw file size of 2.56 MB, is compressed using the proposed feature-driven variable encoding period approach and two constant encoding period approaches. All encodings were stored in the JPEG 80 format. (a) Absolute error image of the low constant precision encoding when $n_{str} = 1$; (b) absolute error image of the proposed method when the encoding precision varies from $n_{str} = [1,6]$ following a distribution centered on the region of interest (the face); (c) absolute error image of the high constant precision encoding when $n_{str} = 6$; (d)-(f) enlarged regions of interest within error images (a)-(c), respectively.

results more closely resemble the lower, constant precision encoding result.

The trends that are visually evident in the qualitative error images in Fig. 4 are also demonstrated in the numeric results. The global RMS reconstruction accuracy for the low, constant precision encoding with $n_{str} = n_{min} = 1$ was 99.56% (1.94 mm of RMS error in the original depth range of 477.4 mm); the accuracy of the higher constant precision encoding with $n_{str} = n_{max} = 6$ was 99.87% (0.64 mm of error); and the proposed method nearly splits the difference at 99.72% accuracy (1.35 mm of error). However, within the region of interest, the proposed method's accuracy of 99.88% (0.56 mm) is significantly closer to the higher precision encoding's accuracy of 99.89% (0.54 mm) than it is to the lower precision encoding's accuracy of 99.66% (1.61 mm). These results were achieved by the proposed method with a JPEG 80 encoding of 54.20 KB, compared to the high precision encoding's file size of 73.47 KB and the low precision encoding's file size of 39.32 KB. These results are summarized in Table 2.

Table 2. Results comparing low and high constant encoding period reconstructions to a varying encoding period reconstruction when an 892 × 718 pixel scan of a bust of former U.S. President Hayes (with a depth range of 477.4 mm and raw file size of 2561.8 KB) is compressed with the JPEG 80 file format. The first column shows the global RMS error and accuracy values. The second column shows the RMS error and accuracy values for the region of interest. The third column shows the file sizes for each approach.

JPEG 80 Results	**Global RMS Error / Accuracy**	**ROI RMS Error / Accuracy**	**File Size (KB)**
Low Constant ($n_{str} = 1$)	1.94 mm / 99.56%	1.61 mm / 99.66%	39.32
Proposed ($n_{str} = [1,6]$)	1.35 mm / 99.72%	0.56 mm / 99.88%	54.20
High Constant ($n_{str} = 6$)	0.64 mm / 99.87%	0.54 mm / 99.89%	73.47

Summary

This paper has presented a novel feature-driven 3D range geometry compression method which makes use of a spatially-aware, variable precision depth encoding that is compatible with both lossless and lossy 2D image formats. Any arbitrary approach can be used to select regions of interest and generate spatially-aware encoding distributions. The proposed method offers the ability to achieve smaller file sizes by compressing unimportant information much more aggressively while preserving the selected regions of interest with higher precision. For example, when encoding a depth scan of a bust using the proposed method and JPEG 80 image compression, the global RMS accuracy was 99.72% (1.35 mm of error) compared to a high precision encoding which achieved 99.87% (0.64 mm) globally. However, the RMS accuracy of the proposed method in the region of interest was 99.88% (0.56 mm) compared to the high resolution encoding's result of 99.89% (0.54 mm). The high precision encoding required 73.47 KB to achieve these results where the proposed method used only 54.20 KB, a savings of more than 26%. The proposed feature-driven spatially-aware 3D range geometry compression method offers pixel-by-pixel customization in how data is compressed, flexible choice of file format, and increased file savings, all while maintaining the targeted precision required for applications with specific regions of interest.

References

[1] Song Zhang. High-speed 3d shape measurement with structured light methods: A review. *Optics and Lasers in Engineering*, 106:119–131, 2018.

[2] Nikolaus Karpinsky and Song Zhang. Composite phase-shifting algorithm for three-dimensional shape compression. *Optical Engineering*, 49(6):1 – 6, 2010.

[3] Song Zhang. Three-dimensional range data compression using computer graphics rendering pipeline. *Appl. Opt.*, 51(18):4058–4064, Jun 2012.

[4] Pan Ou and Song Zhang. Natural method for three-dimensional range data compression. *Appl. Opt.*, 52(9):1857–1863, Mar 2013.

[5] Tyler Bell and Song Zhang. Multiwavelength depth encoding method for 3d range geometry compression. *Appl. Opt.*, 54(36):10684–10691, 2015.

[6] Matthew G. Finley and Tyler Bell. Variable precision depth encoding for 3d range geometry compression. In *3D Measurement and Data Processing 2020*, pages 34–1–34–7. Electronics Imaging, Society for Imaging Science and Technology, Jan 2020.

[7] Matthew G. Finley, Jacob Y. Nishimura, and Tyler Bell. Variable precision depth encoding for 3d range geometry compression. *Appl. Opt.*, 59(17):5290–5299, Jun 2020.

[8] Smithsonian Institution. Rutherford B. Hayes. Accessed and modified under Creative Commons (CC0) from `https://3d.si.edu/object/3d/rutherford-b-hayes%3A1f3700e8-6d01-4488-8ab9-ea14031ef641`, 2021.

Author Biography

Broderick S. Schwartz is a Ph.D. student in Electrical and Computer Engineering at the University of Iowa. Broderick received his B.S. in Mechanical Engineering from Purdue University in 2019 and his M.S. in Electrical and Computer Engineering from the University of Iowa in 2021. His current research interests include 3D capture and compression.

Matthew G. Finley is a recent Ph.D. graduate from The University of Iowa's Electrical and Computer Engineering Department. Finley's research focuses on applied image and signal processing, which he is continuing during his time in the Miles Research Lab (also at The University of Iowa) as a Postdoctoral Research Scholar.

Prof. Tyler Bell is an Assistant Professor of Electrical and Computer Engineering at the University of Iowa. He leads the Holo Reality Lab and is a faculty member of the Public Digital Arts (PDA) cluster. Tyler received his Ph.D. from Purdue University in 2018. His current research interests include high-quality 3D video communications; high-speed, high-resolution 3D imaging; virtual reality, augmented reality; human computer interaction; and multimedia on mobile devices.