10-2009

Streaming 3D Meshes Using Spectral Geometry Images

Ying HE
Nanyang Technological University

Boon Seng CHEW
Nanyang Technological University

Dayong WANG
Nanyang Technological University

Steven C. H. HOI
Singapore Management University, CHHOI@smu.edu.sg

Lap Pui CHAU
Nanyang Technological University

DOI: https://doi.org/10.1145/1631272.1631332

Streaming 3D Meshes Using Spectral Geometry Images

Ying He
Nanyang Technological Univ.
Singapore
yhe@ntu.edu.sg

Boon-Seng Chew
Nanyang Technological Univ.
Singapore
bschew@pmail.ntu.edu.sg

Dayong Wang
Nanyang Technological Univ.
Singapore
dywang@ntu.edu.sg

Steven C.H. Hoi
Nanyang Technological Univ.
Singapore
chhoi@ntu.edu.sg

Lap-Pui Chau
Nanyang Technological Univ.
Singapore
elpchau@ntu.edu.sg

ABSTRACT

The transmission of 3D models in the form of Geometry Images (GI) is an emerging and appealing concept due to the reduction in complexity from $\mathbb{R}^3$ to image space and the wide availability of mature image processing tools and standards. However, geometry images often suffer from the artifacts and error during compression and transmission. Thus, there is a need to address the artifact reduction, error resilience and protection of such data information during the transmission across an error prone network. In this paper, we introduce a new concept, called *Spectral Geometry Images (SGI)*, which naturally combines the powerful spectral analysis with geometry images. We show that SGI is more effective than GI to generate visually pleasing shapes at high compression rates. Furthermore, by coupling SGI to the proposed error protection scheme, we are able to ensure the smooth delivery of 3D model across error networks for different packet loss rate simulated using the two-state Markov model.

Categories and Subject Descriptors

I.3.5 [**Computational Geometry and Object Modeling**]: Curve, surface, solid, and object representations; I.4 [**Image Processing and Computer Vision**]: Compression (coding)

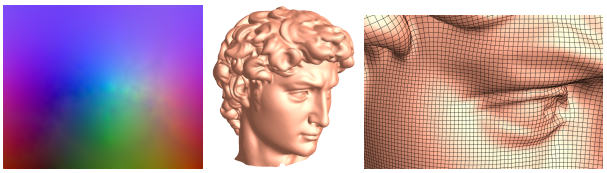
Keywords

Streaming 3D meshes, spectral analysis, geometry image, image compression, conformal parameterization, error resilience, transmission.

1. INTRODUCTION

The evolution of the Internet from a basic communication tool to a content driven community pushes the development of 3D virtual world interaction and exploration to a new height in recent years. Such a trend is significantly seen from the rapid growth in virtual world community and applications like 3D online gaming and content sharing. With the increasing demand for realism in the contents, avatar and environment of the virtual world, it has become an urgent need to ensure the effective representation and smooth delivery of such information across networks given the erratic nature of the Internet. Conventionally, most of this information is represented using 3D data. Although much research has been done to find effective forms to describe a 3D model, polygonal mesh is still the most commonly adopted data type for the representation and delivery of a 3D model/object in many real-world applications.

Despite great successes in computer graphics and virtual reality, the delivery of 3D contents using polygonal meshes have several limitations which are not yet to be thoroughly researched for the effective implementation in a practical system. These include the dependency of the connectivity information to ensure the correct decoding of a 3D model. Since the increase in complexity of model is proportional to the sizes of the geometrical and connectivity information of a mesh, it is difficult to ensure the lossless transmission of the connectivity information while meeting the critical time constraint of a real-time streaming application given a real-world scene that often comprises of multiple highly complex models.



(a) geometry image (b) mesh (c) close-up view

Figure 1: Geometry image of the David head. Geometry image encodes the geometry x, y, and z into a regular grid of the colors r, g and b, which in general is continuous over the entire image domain.

Geometry image, introduced by Gu et al. [11], provides an alternative way for shape representation and transmission. In sharp contrast to the irregular polygonal meshes, geometry image is a completely regular representation that all connectivity information is implicitly encoded in the image space. Furthermore, other geometry and appearance properties, such as normals, textures, materials, can also be stored using the same parameterization. Geometry im-

MM'09, October 19–24, 2009, Beijing, China.

age naturally bridges two research fields, image processing and geometry processing, and provides a way to borrow the well-studied image processing techniques to geometry processing. However, geometry image is fundamentally different from natural images, since geometry image in general is continuous and smooth over the entire image domain as shown in Fig. 1. To integrate geometry image with the existing image processing framework, there is a need for analyzing the geometric properties in geometry image space. To this end, this paper serves this purpose and introduces a new concept, called *Spectral Geometry Image (SGI)*, which naturally combines spectral analysis to geometry image. SGI treats the geometry as a signal and then transforms the signal into the frequency domain using manifold harmonics transform [30]. Then we separate the signal into low-frequency and high-frequency layers, where the low-frequency signals can be represented in a highly compact format and the high-frequency layers are represented as displacement within the user-specified range. Since all layers share the same parameterization in the image space, the original geometry can be easily reconstructed by adding all layers together. Spectral geometry image is more robust than the conventional geometry image in that the spectral analysis is performed on the 3D models and thus independent of the parameterization and re-sampling. As a summary, the contributions of this paper include

- We present a new concept of Spectral Geometry Image (SGI) and develop a framework of constructing spectral geometry images of real-world 3D models.
- We show that spectral geometry image is more powerful and flexible than the conventional geometry image for 3D shape representation and compression, and thus, facilitate the transmission in a lossy network.

The remaining of the paper is organized as follows: Section 2 briefly surveys the previous work in streaming polygonal meshes and spectral geometry processing. Then Section 3 details the algorithm to construct spectral geometry image. Next, Section 4 demonstrates the spectral geometry image in shape compression using image compression techniques. We state the problem of streaming spectral geometry image in Section 5 and packetization techniques in Section 6. The discussions and limitations are presented in Section 7. Finally, we conclude the paper and point out some future research directions in Section 8.

2. RELATED WORK

2.1 Streaming polygonal meshes

Several papers have explored 3D meshes as the basic form of transmission across lossy channel. In this series of work [2, 3, 4], Al-Regib considered the use of Compressed Progressive Mesh (CPM) [25] as the basic representation for progressive transmission. He considered both source and channel allocation for a given bandwidth and proposed the unequal allocation of error correction bits to each layer. The usage of the right transport protocol was also discussed in [1] to reduce the latency and improve the robustness of the 3D data. Bici et al. [7] have also considered the use of Joint Source and Channel Coding (JSCC) scheme for unequal Forward Error Correction (FEC) across packets using the Progressive Geometry Scheme (PGC)[17]. Multiple description coding (MDC)[5] [6] forms the other class of error transmission scheme used for error resilient transmission aside Unequal Error Protection (UEP). In the MDC scheme, descriptions are generated from the sub-meshes of a single model, each containing the full connectivity information to increase the decodability of the model. Aside from the conventional schemes, Mondet et al. proposed the progressive representations for streaming plant type models using retransmission scheme for lost packets [24] . In [8, 9], Cheng et al. proposed a progressive transmission scheme based on an analytical model to investigate the progressive reconstruction of meshes across lossy channel. Li et al. proposed a novel concept of generic middleware for handling different type of triangle based progressively compressed 3D models [19] suited for efficient network delivery of 3D progressive mesh models. They make use of a minimum cost set selector to determine the transmission protocol selection for the individual sub-layer based on their importance and perform streaming based on real-time network traffic characteristic. In another area of mesh based transmission, Yang et al. proposed an optimized scheme to jointly consider the texture effects and mesh representation based on rate-distortion surface to facilitate progressive transmission of 3D model with its respective texture information [32].

Geometry images are an effective representation for compressing shapes that are parameterized to the regular domain, like the rectangle or sphere [11, 15]. Peyré and Mallat presented geometric bandlets to compress geometrically regular objects (like geometry images and normal maps) [26]. They showed that bandeletization algorithm removes the geometric redundancy of orthogonal wavelet coefficients and thus is more effective than the wavelet based compression. Lin et al. considered the use of JPEG2K for compression and delivery of a geometry image [20] . They made use of the ROI (region of interest) characteristic in JPEG2K to achieve view dependent streaming. However, the impact of channel loss towards the decodable bit-stream of the 3D geometry model was not discussed in [20].

2.2 Spectral geometry processing

Spectral geometry processing relies on the eigenvalues and eigenvectors from mesh operators to carry out desired tasks. Motivated by the similarity of the eigenvectors of the graph Laplacian and the discrete Fourier transform, Taubin reduced the surface smoothing problem to low-pass filtering of the discrete surface signals [29]. Since then, there are a large amount of work in spectral geometry processing. We refer the readers to the State of The Art Report (STAR) by Zhang et al. [33] for the most recent survey.

Lévy pointed out the eigenfunctions of the Laplace-Beltrami differential operator capture the global properties of the surface, in some sense, "understand" the geometry [18]. Vallet and Lévy derived a symmetric discrete Laplacians using discrete exterior calculus which guarantees the eigenfunctions form an orthonormal basis, called manifold harmonics basis, with positive eigenvalues [30]. They also developed an efficient and numerically stable approach to compute the eigenfunctions of the Laplacian for meshes with up to a million vertices.

Manifold harmonics provide a natural way to analyze the signals defined on surfaces of arbitrary topology in a Fourier-transform like fashion, which thus has a wide range of ap-

plications in geometry processing. Rustamov presented the Global Point Signature (GPS), a deformation invariant shape signature using the eigenvalues and eigenfunctions of the Laplace-Beltrami operator [28]. Rong et al. [27] proposed spectral mesh deformation that compactly encodes the deformation functions in the frequency domain. Liu et al. presented a robust, blind, and imperceptible spectral watermarking approach for polygonal meshes using manifold harmonics transform [22]. They demonstrated that the spectral approach is very promising to be robust against noise-addition and simplification attacks.

Another related work is the spectral coding algorithm presented by Karni and Gotsman [16]. First, it partitions the 3D model into several submeshes, then computes the spectral of the adjacency matrix for each submesh, and finally, quantizes the spectral coefficients to finite precision. Our spectral geometry image approach is different from [16] in that we partition the frequency domain into hierarchical layers, i.e., the base layer contains the smoothest geometry and the top layer contains the high frequency geometric details, and compress each layer with different compression rates. Furthermore, by taking advantage of the regular structure of geometry image, we can apply more advanced compression techniques (like JPEG2K) than the simple quantization used in [16]. Thus, our approach is more flexible and can achieve better compression ratio with less artifacts.

3. SPECTRAL GEOMETRY IMAGE

This section presents the algorithmic details of constructing spectral geometry image, which includes the following three steps:

1. Apply manifold harmonics transformation to the given surface (Sec. 3.1)
2. Conformally parameterize the surface to a rectangular domain (Sec. 3.2)
3. Separate the low and high frequency layers and map them to the rectangular domain to form spectral geometry image (Sec. 3.3)

3.1 Spectral analysis

This subsection briefly reviews the algorithm to compute the spectrum of Laplacian, i.e., manifold harmonics basis. More details can be found in [30].

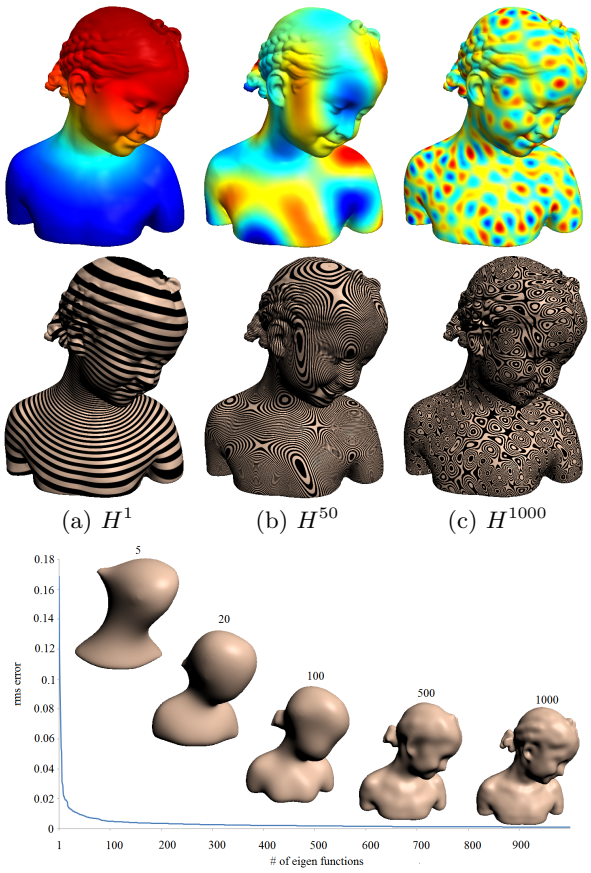
Given a surface M represented by a triangular mesh $M = (V, E, F)$ where V, E, and F are the vertex, edge and face sets, the symmetric Laplace-Beltrami operators is defined as [30]:

$$\begin{cases} \triangle_{ij} = 0 & \text{if} \{v_i, v_j\} \notin E \\ \triangle_{ij} = \frac{\cot\alpha + \cot\beta}{\sqrt{A_i A_j}} & \text{if} \{v_i, v_j\} \in E \\ \triangle_{ii} = -\sum_j \triangle_{ij} \end{cases}$$

where A_i and A_j are the areas of the two triangles that share the edge $\{v_i, v_j\}$ and α and β are the two angles opposite to that edge. The eigenfunctions and eigenvalues of the Laplace-Beltrami operator are all the pairs (H^k, λ_k) that satisfy:

$$\triangle H^k = \lambda_k H^k.$$

Since the Laplacian $\triangle$ is a symmetric matrix, its eigenvalues are real and eigenfunctions are orthogonal. We sort the eigenvalues in the increasing order, $0 = \lambda_0 \leq \lambda_1 \leq \lambda_2 \cdots \leq \lambda_n$. These eigenfunctions are the basis functions and any scalar function defined on M can be projected onto them.



(a) H^1 (b) H^{50} (c) H^{1000}

(d) inverse manifold harmonics transform

Figure 2: Spectral analysis on 3D surface. The eigenfunctions of Laplace-Beltrami operator are orthogonal and serve the manifold harmonics basis. Row 1: the color indicates the function value. Row 2: the texture mapping shows the isocurves of the basis functions. Row 3: reconstructing the 3D mesh from the frequency domain. The number above each model is the number of eigenfunctions used in surface reconstruction.

For each vertex v_i, define a hat function $\Phi_i : M \to \mathbb{R}$ such that $\Phi_i(v_i) = 1$ and $\Phi_i(v_j) = 0$ for all $j \neq i$. Then, the geometry of M is represented by functions $x = \sum_i x_i \Phi_i$ (resp. y, z), where x_i denotes the x-coordinate of v_i. The eigenfunctions H^k can be represented as $H^k = \sum_i H_i^k \Phi_i$. Projecting the function x to manifold harmonics basis, we have

$$\tilde{x}_k = < x, H^k > = \sum_i x_i H_i^k.$$

$\tilde{x}_k$ is the coefficient of k-th frequency of function x. Similarly, we can compute $\tilde{y}_k$ and $\tilde{z}_k$ for functions y and z, respectively.

To reconstruct the shape from the frequency domain, the

coordinates of the vertex v_i are given by

$$x_i = \sum_{k=1}^{m} \tilde{x}_k H_i^k, y_i = \sum_{k=1}^{m} \tilde{y}_k H_i^k, z_i = \sum_{k=1}^{m} \tilde{z}_k H_i^k,$$

where m is the user-specified number of eigenfunctions. With the increasing number of eigenfunctions, the shape can be faithfully reconstructed from the frequency domain. Figure 2 illustrates the manifold harmonics transformation on the Bimba model.

3.2 Conformal parameterization

A key step in constructing (spectral) geometry image is to parameterize the 3D model M to a rectangular domain $D \in \mathbb{R}^2$. Although there are many surface parameterization techniques, we prefer the conformal parameterization due to its shape preserving property and numerical stability [13]. In this paper, we focus on the genus-0 closed surface.

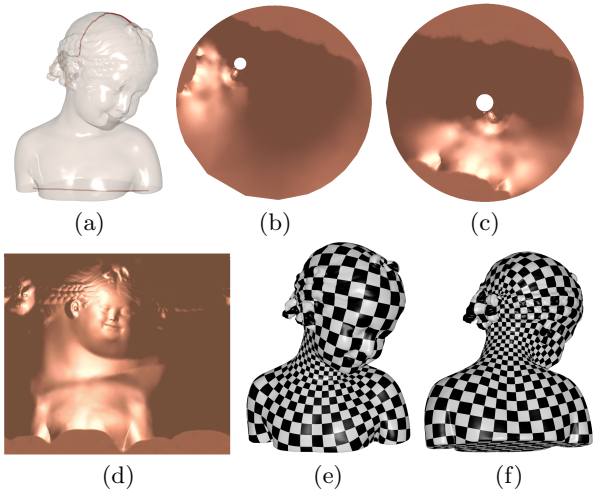


Figure 3: Conformal parameterization of the genus-0 Bimba model. (a) We first modify its topology by two cuts, one at the top and the other at the bottom of the model. The cut surface M' is a topological cylinder. (b) Then we compute the uniform flat metric by discrete Ricci flow and embed M' to a topological annulus. (c) Next, we map the topological annulus to a canonical annulus by a Möbius transformation. (d) We cut the canonical annulus by a line passing through the origin and conformally map it to a rectangle. (e)-(f) The checkerboard texture mapping illustrates the conformality of the parameterization.

Topological modification We first modify its topology by two cuts, i.e., one at the top and the other at the bottom of M. Let M' denote the resultant open surface. Note that M' has the same geometry of M, but is a topological cylinder.

Computing the uniform flat metric Let g denote the Riemannian metric of M'. We want to compute a metric that is conformal to g and flat everywhere inside M' and the geodesic curvature is constant on the boundary $\partial M'$. Such a metric is called uniform flat metric. It is proven that if the total geodesic curvature on each boundary is given, such a uniform flat metric exists and is unique.

In our implementation, we use discrete Ricci flow [12] to compute the uniform flat metric. We set the target Gaussian curvature of each interior point to zero, i.e., it is completely flat, $\bar{K} = 0$, $v \notin \partial M'$. M' has two boundaries, $\partial M' = C_0 \cup C_1$, where C_0 is the boundary with the longer length. Then we set the total geodesic curvature of the boundary C_0 and C_1 to be 2π and -2π respectively, i.e., $\int_{C_0} \bar{k} = 2\pi$ and $\int_{C_1} \bar{k} = -2\pi$. It can be easily verified that the total geodesic and Gaussian curvatures satisfy the Gauss-Bonnet theorem,

$$\int_{M'} K + \int_{\partial M'} k = \int_{M'} \bar{K} + \int_{\partial M'} \bar{k} = 2\pi\chi,$$

where $\chi = 0$ is the Euler number of M'.

It is proven that discrete Ricci flow converges exponentially fast [10] and the steady state is the desired uniform flat metric. With the uniform flat metric, the Gaussian curvature of interior vertices are zero, thus, the faces can be flattened one by one on the plane.

Conformal map to a canonical annulus Note that the embedded surface $\phi(M')$ may not be the canonical annulus. Let $\phi(C_1) : |z - c_1| = r_1$ and $\phi(C_2) : |z - c_2| = r_2$ be the outer and inner circles of the topological annulus. We want to find a Möbius transformation $w : \mathbb{C} \to \mathbb{C}$ to map $\phi(C_1)$ and $\phi(C_2)$ to concentric circles with center at the origin. A Möbius transformation is uniquely determined by three pairs of distinct vertices $z_i \in \mathbb{C}$ and $w_i \in \mathbb{C}$, $i = 1, 2, 3$, such that $w(z_i) = w_i$. Set $w_1 = 0$ and $w_2 = \infty$, i.e., w_1 and w_2 are symmetric w.r.t. the canonical annulus. Therefore, the pre-images z_1 and z_2 are symmetric w.r.t. $\phi(C_1)$ and $\phi(C_2)$, i.e., $|z_1 - c_1||z_2 - c_1| = r_1^2$ and $|z_1 - c_2||z_2 - c_2| = r_2^2$. We also set $z_3 = c_1 + (r_1, 0)$ and $|w_3| = 1$, i.e., the radius of the outer circle in the canonical annulus is one. Then, the Möbius transformation is given by

$$w(z) = \rho e^{i\theta} \frac{z - z_1}{z - z_2},$$

where $\rho = \frac{|z_3 - z_2|}{|z_3 - z_1|}$ and θ is an arbitrary angle.

Conformal map to a rectangular domain We cut the canonical annulus by a line passing through $(0, 0)$ and $(1, 0)$. Finally, we conformally map the cut annulus to a rectangular domain by

$$\psi(z) = |z| + i \arg z, \quad z \in \mathbb{C}$$

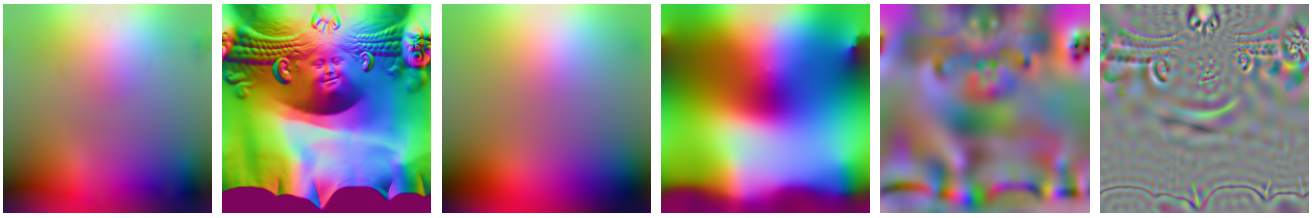
Putting them all together, the conformal parameterization $f : M' \to D \in \mathbb{R}^2$ is given by the composite map,

$$f = \psi \circ w \circ \phi,$$

which is guaranteed to be conformal (angle-preserving) and diffeomorphism. The checkerboard texture mapping illustrates the conformality of the parameterization.

3.3 Construction of spectral geometry image

Spectral geometry image is more flexible than the conventional geometry image due to its capability to separate geometry image into low- and high-frequency layers. Note that the low-frequency layers represent the rough shape and the high-frequency layers represent the detailed geometry. In our framework, the user specifies the number of desired layers l and the reconstruction tolerance for each layer ϵ_i, $i = 1, \cdots, l-1$. We denote M_i the reconstructed mesh with m_i eigenfunctions where m_i is determined by finding the



(a) GI (b) normal map of (a) (c) SGI_1 $\epsilon_1 = 0.04$ (d) normal map of (c) (e) SGI_2 $\epsilon_2 = 0.01$ (f) SGI_3

Figure 4: Spectral geometry image. (a)-(b) show the geometry image of the Bimba model. (c)-(f) show the 3-layer spectral geometry image. The normal maps in (b) and (d) highlight the difference between SGI_1 and GI. To better view the high frequency layers SGI_2 and SGI_3, the pixel values are normalized to $[0, 255]$.

smallest integer such that

$$\begin{aligned} & \|M_i - M\|_\infty \\ = & \max_j \sqrt{\sum_{k=m_i+1}^{\infty} (\tilde{x}_k H_j^k)^2 + (\tilde{y}_k H_j^k)^2 + (\tilde{z}_k H_j^k)^2} \leq \epsilon_i. \end{aligned}$$

The spectral geometry images are defined as follows:

$$\begin{aligned} & SGI_1 : M_1 \rightarrow D \\ & SGI_2 : M_2 - M_1 \rightarrow D \\ & \cdots \\ & SGI_i : M_i - M_{i-1} \rightarrow D \\ & \cdots \\ & SGI_l : M - M_{l-1} \rightarrow D \end{aligned}$$

Intuitively speaking, M_1 represents the coarsest reconstruction of the 3D model. The remaining layers M_i encode the displacement between the following two consecutive layers M_i and M_{i-1} and present the model with increasing quality until the original model is decoded in the top layer M_l. Thus, to reconstruct the geometry with the user-specified tolerance ϵ_i, we simply add the layers up to i. Figure 4 shows the 3-layer spectral geometry image of the Bimba model with the tolerances $\epsilon_1 = 0.04$ and $\epsilon_2 = 0.01$. The model is normalized to a unit cube.

4. SPECTRAL GEOMETRY IMAGE COMPRESSION

In the previous section, we have defined the different layers of spectral geometry image from the 3D model. Now, with low and high frequency geometry represented in images, we can make use of existing image processing techniques to achieve compression and delivery of the model.

The JPEG2K and JPEG-XR are widely adopted standards due to their robustness and efficacy in compressing natural images and they both have good encoders that support 48-bit images. In Fig. 5, we compare JPEG2K and the conventional JPEG-XR on a set of 10 natural images and 10 geometry images. It can seen that JPEG2K is more effective than JPEG-XR to compress both kinds of images for moderate quality results, while JPEG-XR is more preferred for lossy compression of high quality results. Since JPEG2K adopted the discrete wavelet transform for energy packing, it results in better PSNR for a given bit size compared to JPEG-XR. In this paper, JPEG2K is adopted as

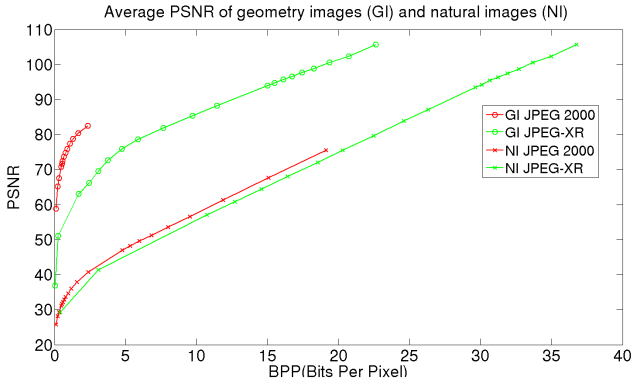


Figure 5: JPEG2K vs JPEG-XR. One can clearly see that JPEG2K is more effective than JPEG-XR at low *bpp*, while JPEG-XR is more preferred for lossy compression of high quality results. Due to the smooth nature, geometry images have better PSNR than the natural images.

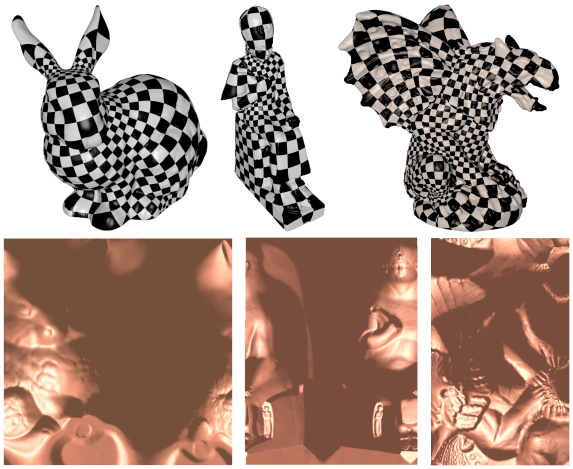
the compression algorithm for both spectral geometry image and geometry image.

To measure the quality of the compressed surface M', we use the following root-mean-square error d_r and mean curvature error d_H,

$$\begin{aligned} d_r(M, M') &= \sqrt{\frac{1}{|V|} \sum_{i=1}^{|V|} \|v_i - v_i'\|^2}, \\ d_H(M, M') &= \sqrt{\frac{1}{|V|} \sum_{i=1}^{|V|} \| \triangle v_i - \triangle v_i'\|^2}, \end{aligned}$$

where $|V|$ is the number of vertices in M and M'. Note that $\triangle v = H\mathbf{n}$ where $\mathbf{n}$ is the normal and H the mean curvature, thus, d_H encodes the surface details and measures the smoothness of the compressed mesh and reflects the visual quality better than d_r.

Figure 6 shows the test models and their parameterizations. Figure 7 compares the performance of GI and SGI of the Gargoyle model. We use d_H to measure the smoothness of the compressed models. Note that both GI and SGI discard high frequency details at high compression rate. The low-frequency layer of SGI is more smooth than that of GI and thus lead to less artifacts.



(a) Bunny (b) Ramesses (c) Gargoyle

Figure 6: Conformal parameterization of the test models.

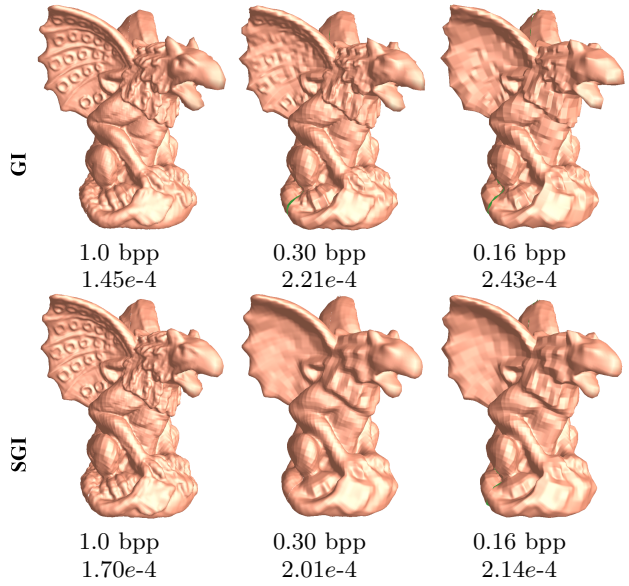


Figure 7: Mean curvature error d_H measures the visual quality. Row 1: the GI of resolution 366×600; Row 2: the 2-layer SGI of resolution 183×300 and 366×600 with $\epsilon_1 = 0.008$. The numbers below each figure are the *bits per pixel* (*bpp*) and the mean curvature error d_H. SGI has smaller d_H at low *bpp* because the high frequency layer is discarded and the low-frequency layer has less distortion than GI.

Figure 8 compares the performance of GI and SGI of the Bimba model. When reconstructing the geometry from multi-layer SGI of different resolutions, we up-sample the bottom layers to the resolution of the top-most layers. This upsampling usually smoothes the low-frequency geometry, but it does not change the top-most layer that contains the high-frequency details within the user-specified range. Since the mean curvature vector encodes the differential coordinates [21] representing the local details, it is insensitive towards the small-scale deformation of the bottom layers. As a result, the SGI-256/512 outperforms the SGI-512/512 in terms of mean curvature error and the visual quality. However, the root-mean-square error d_r, in contrast to d_H, is highly sensitive to the bottom layers, and the PSNR of SGI-512/512 is better than SGI-256/512.

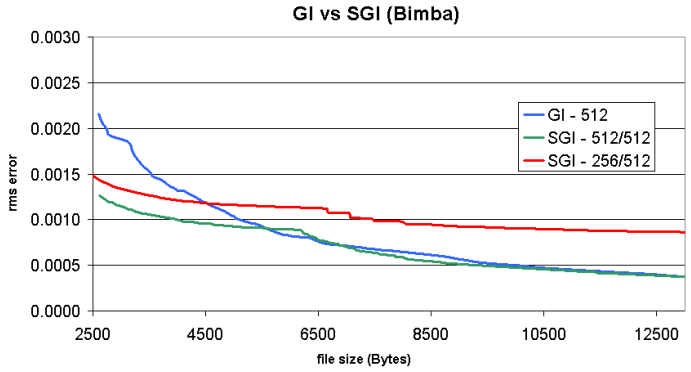
5. STREAMING SPECTRAL GEOMETRY IMAGES

In the design of our coding scheme for the SGI, we consider the dynamic nature of the channel bandwidth and propose the use of multi-resolution reconstruction suited for meeting the demand of different clients/server terminal. For the purpose of progressive streaming, the input spectral geometry image is decomposed into l image layers to support the reconstruction of partial bit-stream from coarse to high quality 3D model at the decoder end. However, due to the dependency between the layers of such multi-resolution models, the effects of channel errors on the decoded 3D model could be extremely significant when the compressed data is transmitted across erroneous channel. Therefore, there is a need to exert some form of error control to ensure a measure of reliability is maintained in the presence of network error. The forward error code aims to protect data against such channel errors through the introduction of parity codes. It is well known in its ability for error detection and error correction for data communication system. Here, the popular Reed Solomon code is used in this paper to generate the necessary (FEC) code for efficient protection during the progressive transmission.

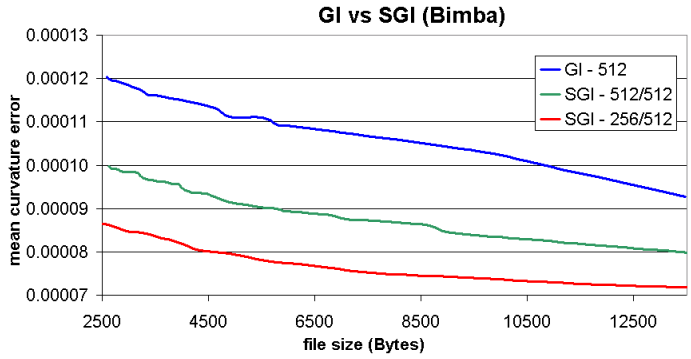
Next, we need to address the dependency between each image layer of the SGI since the higher layers are coded based upon the lower layers. Therefore, in order to minimize the impact of transmission error, an appropriate amount of FEC allocation is necessary to ensure the uneven protection to the different image layers of the model. The next section solves the problem for the channel bits allocation to each image layer that will enable us to retrieve the best 3D model quality at the decoder end, given a constraint in the total bit budget.

6. PACKETIZATION

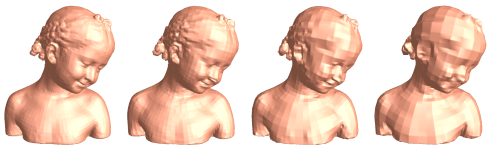
The bit-stream generated from the earlier section consists of image layers having different amount of importance. We will look into the aspect of error resilience partitioning and error correction coding (FEC allocation) for effective delivery of the model across error prone network. In streaming application, a simple yet efficient error resilient method is considered for $SGI_1, SGI_2, \cdots, SGI_l$. Each layer of image differences is first partitioned into slices containing group of macroblocks (GOM) and packetize independently. Since there exists correlation between the macroblocks of each group, a damaged macroblock due to channel error can be concealed using the surrounding undamaged blocks. Next, out of consideration for the packetization of different layers, we denote the width and height of a bitstream as w_b and h_b respectively, where $b = 1, 2, \cdots, l$ represents the layer number. The total packet number and packet size is denoted by N and M respectively. Since the bitstream consists of layers having unequal importance, thus an uneven amount of channel allocation should be assigned to the more important lower layers than its subsequent higher ones. We want to find the best channel allocation to the individual



(a) root-mean square error d_r

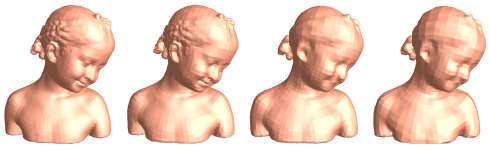


(b) mean curvature error d_H



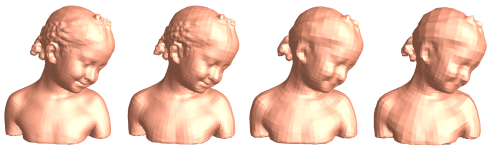
13,475 B | 8,574 B | 4,010 B | 2,595 B
68.8 dB | 64.3 dB | 57.5 dB | 53.3 dB
9.27e-5 | 1.08e-4 | 1.17e-4 | 1.20e-4

(c) GI of resolution 512×512



13,378 B | 8,572 B | 3,928 B | 2,618 B
61.3 dB | 60.5 dB | 58.3 dB | 56.9 dB
7.70e-5 | 8.21e-5 | 8.28e-5 | 8.61e-5

(d) 2-layer SGI of resolution 256×256 and 512×512



13,625 B | 8,566 B | 4,022 B | 2,622 B
68.9 dB | 65.4 dB | 60.1 dB | 58.0 dB
8.65e-5 | 9.47e-5 | 9.65e-5 | 9.98e-5

(e) 2-layer SGI of resolution 512×512 and 512×512

Figure 8: Shape compression using geometry image (GI) and spectral geometry image (SGI). GI is of resolution 512×512. We constructed two SGIs with 2 layers and $\epsilon_1 = 0.0075$. The first SGI is of resolution 256×256 and 512×512. The second SGI is of resolution 512×512 and 512×512. As shown in (a) and (b), SGI's performance is better than GI at low compression rates. (c)-(e) show the compressed GI and SGI. The numbers below each figure are the file size ($Bytes$), the PSNR ($20 \log_{10} 1/d_r$ dB) and the mean curvature error d_H. When reconstructing the geometry from multi-layer SGI of different resolutions, we up-sample the bottom layers to the resolution of the top-most layer. This up-sampling usually smoothes the bottom layers. The mean curvature vector d_H encodes the differential coordinates and is insensitive towards the small deformation of the bottom layers. As a result, SGI-256/512 outperforms the SGI-512/512 in terms of mean curvature error and the visual quality. However, the root-mean-square error d_r, in contrast to d_H, is highly sensitive to the bottom layers. Thus, SGI-512/512 has smaller d_r than SGI-256/512 at high compression rates.

layers to maximize the quality of the decoded model, which is measured by the mean curvature error d_H.

6.1 Channel Allocation

We now address the allocation problem for the different layers and denote by Q and C respectively at the total bit budget from the network bandwidth, and the total channel bits available for FEC allocation. We thus have the following inequality:

$$C \leq Q - \sum_{i=1}^{l-1} SGI_i$$

The channel allocation for a bitstream is represented as C_i where $C_1, C_2, \cdots, C_{l-1}$ denote the channel bits allocated for the individual layers of the model and SGI_i as the source bits in the i layers. Fig. 9 shows the packetization scheme used on each sub-bitstream. The proposed scheme takes into consideration the importance of the individual layers and allocates more channel information in the form of Reed Solomon(RS) code to higher priority lower layers as compared to the higher one. For this reason, we made the assumption that the channel rates for individual layers of a sub-bitstream to be always non-decreasing in nature where $r_1 \geq r_2 \geq \cdots \geq r_{l-1}$. Here, r_i denote the channel rate of the layer i determine from $N - w_i$. This was proven to be a necessary condition for type-II packet with convex R-D function. We define the distortion reduction of the layer i calculated from the error measurement d_H as $R_i = D_{i+1} - D_i$ where $i = 1, 2, ..., l-1$. Now, we want to determine the best channel rate allocation to the different layers and achieve a minimum d_H. The overall contribution to the improvement in model quality is represent as

$$D = \sum_{i=1}^{l-1} R_i \times P(i)$$

In this paper, we adopted the two-state Markov model as the channel estimator to approximate the wireless channel's

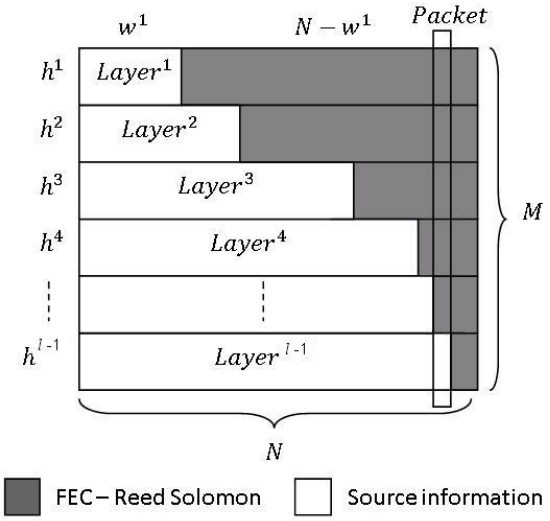


Figure 9: Channel allocation scheme for the individual layers and sub-bitstream

packet loss behavior. $P(i)$ represents the probability that the layer i of bitstream is decodable. For each bitstream,

$$P(i) = \sum_{m=0}^{N-w_i} p(m,n)$$

We illustrate the used of RS codes for layers protection to ensure the decodability of a single layer when $N - w_i$ is received. $p(m,n)$ denotes the probability of losing m packets while transmitting n packets. The objective so far is to find the channel rate for each layer and determine the bitstream to obtain the best reconstructed quality. To reduce the complexity of the optimization, we define the non-decreasing channel rate allocation earlier as

$$constraint\ 1: \qquad r_1 \geq r_2 \geq \cdots \geq r_{l-1}$$

This ensures that the lower and more important layers to always receive a greater amount of channel allocation than the higher layers for efficient channel allocation. Next, to ensure the constraint in channel bandwidth is kept, we define

$$constraint\ 2: \qquad \sum_{i=1}^{l-1} SGI_i + C \leq Q.$$

This constraint ensures the total source bits and available channel protection information to never exceed the total bandwidth during transmission. Now, with all the constraints and distortion metrics defined, the final optimization problem is formulated as:

$$\min D, \text{subject to constraint 1 and 2}$$

Table 1 shows our simulation results for the transmission of the Bimba model. In the experiment, we perform the channel simulation for the SGI and GI datasets using varying packet lost rate of 2%, 5%, 12% and 20%. The total number of pixels used in SGI and GI is 65,536. Due to the efficient compression of the lower frequency geometries of the SGI, the 3D model can be encoded using a minimum bit budget of 0.4-0.55 bits per pixel (bpp) for the Spectral Geometry Image and conventional GI. The *bpp* define the total bit budget Q used for both source and channel encoding. Since in the case of SGI where the amount of data reduction is significant from the lower layer, this enable more data bits to be available for channel allocation of the other layers, thus improving the overall quality of the decoded 3D model. The proposed UEP scheme has also taken into consideration the importance of each layer and allocate channel bits accordingly. From the experimental result, it shows the superior performance of Spectral Geometry Image technique over the conventional GI. By coupling the SGI with the proposed UEP scheme, we are able to ensure the smooth degradation of the model over varying packet loss rate comparable to the GI encoding method, while maintaining a better error measurement during the transmission.

Table 1: Comparison of the Equal Error Protection(EEP) and Unequal Error Protection(UEP) schemes for the Bimba model. The GI is of resolution 256×256 and the 3-layer SGI is of resolution 128×128, 128×128 and 256×256. The packet loss rate are 2%, 5%, 12% and 20%, and the *bpp* is in the range of 0.40-0.55 bits. The error measurement is presented in $\mathbf{10}\log_{10} 1/d_H$

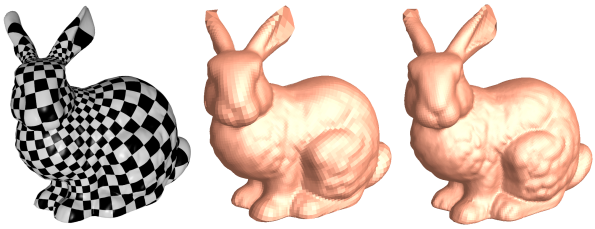
EEP *bpp*	2%	5%	12%	20%
0.4 (SGI)	35.84577	35.81972	35.74718	35.72328
0.4 (GI)	34.92499	34.89077	34.78266	34.74139
0.45 (SGI)	35.85196	35.85086	35.82787	35.76416
0.45 (GI)	34.93229	34.93089	34.89907	34.80539
0.5 (SGI)	35.94645	35.92651	35.85149	35.81652
0.5 (GI)	35.2009	35.10979	34.80532	34.68106
0.55 (SGI)	35.95014	35.94829	35.95013	35.84936
0.55 (GI)	35.21916	35.21019	35.07541	34.79984
UEP *bpp*	2%	5%	12%	20%
0.4 (SGI)	35.85197	35.85196	35.85194	35.75175
0.4 (GI)	34.93232	34.9323	34.93227	34.93151
0.45 (SGI)	35.85197	35.85196	35.85195	35.85189
0.45 (GI)	34.93232	34.9323	34.93228	34.93218
0.5 (SGI)	35.95016	35.95014	35.95013	35.95006
0.5 (GI)	35.21924	35.21918	35.21911	35.21759
0.55 (SGI)	35.95016	35.95014	35.95013	35.95007
0.55 (GI)	35.21924	35.21918	35.21912	35.21884

7. DISCUSSIONS

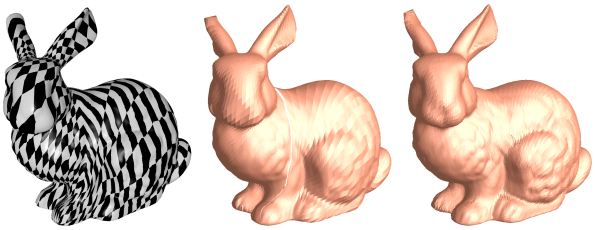
In this section, we discuss several issues of spectral geometry image and show its limitations.

The major difference between the proposed spectral geometry image and geometry image is that the SGI performs the manifold harmonics transform (MHT) on the 3D model and then partitions the frequency domain with the user-specified tolerance. Finally, the partitioned layers are re-sampled to the 2D domain by the parameterization. Thus, the spectral analysis of SGI is independent of the parameterization. The GI re-samples the 3D geometry to the rectangle domain, and then performs the discrete wavelet transform (DWT) on the 2D rectangular grid. Thus, compared to SGI, GI's spectral analysis highly depends on the parameterization and re-sampling. As shown in Fig 10, a parameterization of poor quality may result in the jaggedness artifacts due to the high anisotropy. However, SGI with the same parameterization shows more robust results.

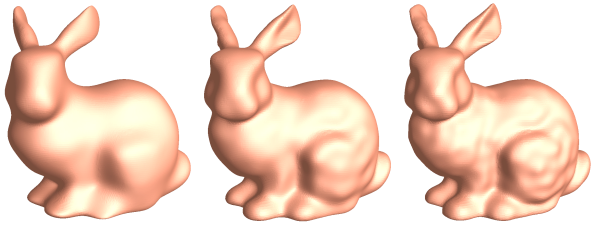
Bandlets extend the wavelets to capture the anisotropic



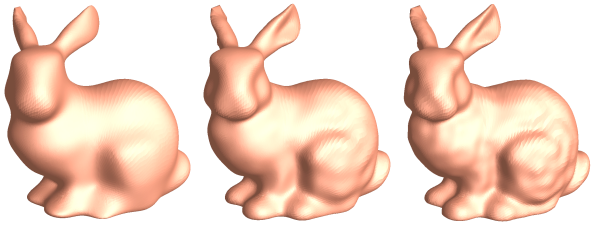
(a) DWT applied to 2D regular grid with parameterization of good quality



(b) DWT applied to 2D regular grid with parameterization of poor quality



(c) MHT applied to 3D mesh



(d) re-sampling of MHT results by the parameterization of poor quality

Figure 10: Discrete wavelet transform is highly dependent of the parameterization and re-sampling. (a) and (b) show DWT applied to 2D regular grid with different parameterizations. The middle and right figures show the reconstructed meshes with LL subband and LL+HL+LH subbands respectively. Clearly, the parameterization with poor quality results in the jaggedness artifacts due to high anisotropy. (c) Manifold harmonics transform is performed on the 3D meshes directly, thus, independent of the parameterization. From left to right, the reconstructed meshes with the number of eigenfunctions, 160, 500 and 1000 respectively. (d) shows the re-sampling of MHT results by the parameterization of poor quality. It has much less artifacts than that of (b).

regularity of edge structures, thus, they are very promising in processing images with rich geometric structures [23]. Peyré and Mallat applied bandlet to compress geometrically regular objects, like geometry images and normal maps, and showed that bandlets improve the wavelets on complex 3D models [26]. Our approach makes use of JPEG2K (which in turn is based on wavelet) to compress spectral geometry images. Bandlets are more effective than wavelets to approximate smooth edges and sharp features. However, there is no gain to apply bandlets to the highly smooth geometry. Within spectral geometry image framework, only the high-frequency layer encodes the edges and sharp features. Thus, it would be promising to combine spectral geometry images with bandlets such that the low- and high-frequency layers are compressed by wavelets (JPEG2K) and bandlets separately.

Progressive mesh (PM) is an effective solution to deliver 3D models across error prone network since its hierarchical structure facilitates reducing latency time for previewing large-scale, complex models. However, the quality of PM based model is highly dependent on the correct decoding of the connectivity information, and previous LOD layers of a 3D mesh is often necessary for progressive reconstruction of the model. In the context of SGI, the delivery order of the SGI layers is additive in nature thus making it suitable for transmission across unreliable channel. In addition, the connectivity information of the 3D model is also implicitly encoded in the image, thus additional protection for such information is not required for SGI.

There are several limitations in our framework. First, the user specifies the number of layers of SGI and the reconstruction tolerance for each layer, then our algorithm computes the desired number of eigenfunctions used to reconstruct the surface. The tolerance must be chosen with care. If the tolerance is too big, the low-frequency layer only captures very rough geometry and does not encode enough information. On the other hand, if the tolerance is too small, the low-frequency layer encodes too many details and may not be helpful to reduce the file size. Furthermore, a small tolerance usually results in large number of eigenfunctions. Unfortunately, solving the eigen problems on surfaces is expensive and time consuming even with the state-of-the-art method [30].

Another limitation lies in the surface parameterization. In this paper, we parameterize the genus-0 model to a rectangular domain using conformal parameterization [13]. The conformal parameterization minimizes the angle distortion, however, usually results in large area distortions. For example, the Bunny ears and Gargoyle wings and head have large area distortion as shown in Fig. 6. For such models, a polycube [31, 14] is an ideal parametric domain for the geometry image, but it is out of the scope of this paper.

8. CONCLUSIONS AND FUTURE WORK

This paper presents *spectral geometry image*, a novel 3D model representation suitable for streaming application across lossy channel. We show that spectral geometry image is a more powerful and flexible encoding scheme compared to the conventional geometry image techniques in terms of shape representation and data compression. In addition, the SGI couples well with the proposed channel allocation scheme and outperforms the conventional geometry image techniques in term of error resilience towards channel loss.

There are several interesting topics that are worthy of further investigation. First, SGI being a newly developed technique, we will like to further investigate its performance with other existing state of art 3D model representation in

term of encoding sizes and quality. Another exciting area involves the transmission of SGI across wireless channels that poses a challenging problem due to the lossy nature of wireless channels. Thus there is a need for developing better error resilient, correction and concealment tools to ensure the smooth delivery of 3D model across the error prone network. Last, conformal parameterization to a rectangular domain may introduce very large area distortion if the 3D model has complicated topology and geometry. In the future, we are going to generalize the spectral geometry image to the polycube domain [31, 14] and apply it to real-world models of complicated geometry and topology.

9. ACKNOWLEDGEMENT

This work was supported by the Singapore National Research Foundation Interactive Digital Media R&D Program, under research Grant NRF2008IDM-IDM004-006. We would like to thank the reviewers for their careful reviews and constructive comments. The 3D models are courtesy of Stanford University and Aim@Shape Shape Repository.

10. REFERENCES

[1] G. Al-Regib and Y. Altunbasak. 3TP: an application-layer protocol for streaming 3-D models. *Multimedia, IEEE Transactions on*, 7(6):1149–1156, 2005.

[2] G. Al-Regib, Y. Altunbasak, and R. Mersereau. Bit allocation for joint source and channel coding of progressively compressed 3-D models. *TCSVT*, 15(2):256–268, 2005.

[3] G. Al-Regib, Y. Altunbasak, and J. Rossignac. Error-resilient transmission of 3D models. *ACM Trans. Graph.*, 24(2):182–208, 2005.

[4] G. Al-Regib, Y. Altunbasak, and J. Rossignac. An unequal error protection method for progressively transmitted 3D models. *Multimedia, IEEE Transactions on*, 7(4):766–776, 2005.

[5] M. O. Bici and G. B. Akar. Multiple description scalar quantization based 3d mesh coding. In *ICIP*, pages 553–556, 2006.

[6] M. O. Bici, A. Norkin, G. Akar, A. Gotchev, and J. Astola. Multiple description coding of 3d geometry with forward error correction codes. In *3DTV07*, pages 1–4, 2007.

[7] M. O. Bici, A. Norkin, and G. B. Akar. Packet loss resilient transmission of 3d models. In *ICIP (5)*, pages 121–124, 2007.

[8] W. Cheng. Streaming of 3D progressive meshes. In *MULTIMEDIA '08*, pages 1047–1050, 2008.

[9] W. Cheng, W. T. Ooi, S. Mondet, R. Grigoras, and G. Morin. An analytical model for progressive mesh streaming. In *MULTIMEDIA '07*, pages 737–746, 2007.

[10] B. Chow and F. Luo. Combinatorial ricci flows on surfaces. *J. Differential Geom.*, 63(1):97–129, 2003.

[11] X. Gu, S. J. Gortler, and H. Hoppe. Geometry images. In *SIGGRAPH*, pages 355–361, 2002.

[12] X. Gu, S. Wang, J. Kim, Y. Zeng, Y. Wang, H. Qin, and D. Samaras. Ricci flow for 3D shape analysis. In *ICCV*, pages 1–8, 2007.

[13] X. Gu and S.-T. Yau. Global conformal parameterization. In *SGP*, pages 127–137, 2003.

[14] Y. He, H. Wang, C.-W. Fu, and H. Qin. A divide-and-conquer approach for automatic polycube map construction. *Computers and Graphics*, 33(3):369–380, 2009.

[15] H. Hoppe and E. Praun. Shape compression using spherical geometry images. In *Advances in Multiresolution for Geometric Modelling*, pages 27–46, 2003.

[16] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *SIGGRAPH*, pages 279–286, 2000.

[17] A. Khodakovsky, P. Schröder, and W. Sweldens. Progressive geometry compression. In *SIGGRAPH*, pages 271–278, 2000.

[18] B. Lévy. Laplace-beltrami eigenfunctions towards an algorithm that "understands" geometry. In *SMI*, page 13, 2006.

[19] H. Li, M. Li, and B. Prabhakaran. Middleware for streaming 3d progressive meshes over lossy networks. *TOMCCA*, 2(4):282–317, 2006.

[20] N.-H. Lin, T.-H. Huang, and B.-Y. Chen. 3d model streaming based on jpeg 2000. *Consumer Electronics, IEEE Transactions on*, 53(1):182–190, 2007.

[21] Y. Lipman, O. Sorkine, D. Cohen-Or, D. Levin, C. Rössl, and H.-P. Seidel. Differential coordinates for interactive mesh editing. In *SMI '04*, pages 181–190, 2004.

[22] Y. Liu, B. Prabhakaran, and X. Guo. A robust spectral approach for blind watermarking of manifold surfaces. In *MM&Sec*, pages 43–52, 2008.

[23] S. Mallat and G. Peyré. A review of bandlet methods for geometrical image representation. *Numerical Algorithms*, 44(3):205–234, 2007.

[24] S. Mondet, W. Cheng, G. Morin, R. Grigoras, F. Boudon, and W. T. Ooi. Streaming of plants in distributed virtual environments. In *MULTIMEDIA '08*, pages 1–10, 2008.

[25] R. Pajarola and J. Rossignac. Compressed progressive meshes. *TVCG*, 6(1):79–93, 2000.

[26] G. Peyré and S. Mallat. Surface compression with geometric bandelets. *TOG*, 24(3):601–608, 2005.

[27] G. Rong, Y. Cao, and X. Guo. Spectral mesh deformation. *TVC*, 24(7-9):787–796, 2008.

[28] R. M. Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *SGP*, pages 225–233, 2007.

[29] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH*, pages 351–358, 1995.

[30] B. Vallet and B. Lévy. Spectral geometry processing with manifold harmonics. *Comput. Graph. Forum*, 27(2):251–260, 2008.

[31] H. Wang, Y. He, X. Li, X. Gu, and H. Qin. Polycube splines. *Computer-Aided Design*, 40(6):721–733, 2008.

[32] S. Yang, C.-H. Lee, and C.-C. J. Kuo. Optimized mesh and texture multiplexing for progressive textured model transmission. In *MULTIMEDIA '04*, pages 676–683, 2004.

[33] H. Zhang, O. van Kaick, and R. Dyer. Spectral methods for mesh processing and analysis. In *Proc. of Eurographics State-of-the-art Report*, pages 1–22, 2007.