Perceptually Coded Transmission of Arbitrary 3D Objects over Burst Packet Loss Channels enhanced with a Generic JND Formulation

Irene Cheng[1], Lihang Ying[2] and Anup Basu[1]

[1]Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada

[2]City of Edmonton, Edmonton, Alberta, Canada

Email:locheng@ualberta.ca, yinglihang@gmail.com, basu@ualberta.ca

Abstract—In this work we propose a new approach to account for burst packet loss during transmission of 3D objects represented by texture and mesh over unreliable networks. Our strategy includes applying stripification on the 3D mesh following the valence-driven algorithm and distributing nearby vertices into different packets, combined with an interleaving technique that does not need texture or mesh packets to be re-transmitted. The perceptually-driven technique is able to interpolate successfully lost mesh features even under severe packet loss. The reconstructed mesh is improved further by applying our curvature-driven probabilistic strategy to safeguard visually significant structures on the 3D surface. Experimental results show that smoothness on the object surface is preserved even at 50% packet loss. At 75% packet loss, smoothness on the object surface deteriorates but the overall shape of the objects is still preserved. We also define a Quality of Experience (QoE) metric to formulate the Just-Noticeable-Difference (JND) concept, to quantify the qualitative findings obtained from earlier subjective user studies, which provides flexibility to applications for reducing the transmission of visually redundant data.

Index Terms—QoE-aware error control, Just-Noticeable-Difference (JND), QoE metric, Burst packet loss, Unreliable network, 3D graphics, Mesh geometry-driven.

I. INTRODUCTION

3D graphics is commonly used in many online applications, such as games, virtual reality, augmented reality and other immersive environments including the latest 3DTV displays. In pace with the rapidly growing online entertainment industry and the supply of high definition display devices, increasing user demand for higher quality and richer multimedia content have driven application developers to search for breakthroughs in order to satisfy user expectations and enhance viewing experience. Furthermore, the launch of advanced wireless communication infrastructures has broadened user's viewing platforms from conventional large scale virtual CAVE and Reality Centre environments to a portable and mobile setting. One of the major challenges associated with these technological privileges is how to provide satisfactory Quality of Experience (QoE) so as to keep the consumers entertained and engaged; and sustain evolving commercial products and research.

Transmitting high definition data consumes significant bandwidth. For this reason, many state-of-the-art compression algorithms have been developed, associated with the introduction of various QoE metrics, which are essential for assessing the rendered quality of the decompressed data. These metrics can be subjective or objective. Subjective metrics require qualitative user studies. However, conducting user studies involves tedious ethics approval process, controlled experimental setup and recruiting a sufficiently large subject pool in order to achieve a statistically significant outcome. Objective metrics are based on quantitative analysis, e.g., signal-to-noise (SNR) and mean square error (MSE), hoping that the results can reflect the degree of satisfaction of human viewers. While QoE metrics for 2D media have been extensively studied, the research on incorporating human perceptual factors in the compression and transmission pipeline for 3D content still lags behind its 2D counterpart. In this paper, our contribution is three-folds: (1) enhancing the viewing experience by proposing a geometry-driven interleaving technique for transmitting 3D meshes over burst packet loss channels, (2) introducing a geometry-driven curvature histogram to assign higher priority to more noticeable surface features, and (3) formulating the Just-Noticeable-Difference (JND) concept in a mathematical model, which is consistent with qualitative user studies, and applying this QoE metric to minimize the transmission of redundant data so that constrained resources, e.g., time and bandwidth, can be allocated to other media resulting in better overall quality.

Transmitting texture alone, geometry alone or texture/geometry combined are challenging issues and worth individual discussions. A JND solution for transmitting high resolution texture taking human perception into consideration was published by the authors in separate papers. The trade-off between allocating limited bandwidth to texture vs. geometry was discussed and a perceptually optimized strategy was proposed in [5]. The focus of this manuscript is on geometry data transmitted over burst packet loss channels.

The rest of the paper is organized as follows: Section II reviews 3D mesh coding and introduces our perceptually-coded interleaving strategy for burst packet loss channels. Section III gives experimental results and analysis. Section IV formulates our QoE metric by defining a mathematical model to quantify the Just-Noticeable-Difference concept. Finally, conclusion is presented in Section V.

II. 3D CODING AND TRANSMISSION OVER BURST PACKET LOSS NETWORK

3D transmission over unreliable networks needs to take into account the possibility of packet loss, which often happens in burst. In order to minimize the impact of burst errors on the transmitted data, interleaving has been used by researchers as an effective tool [28][29][30]. However, interleaving techniques for 1D and 2D data such as image or video, which have fixed adjacent distance, are not suitable for arbitrarily connected 3D meshes, which do not follow the regular row and column traversal pattern. Our proposed interleaving technique, which considers both irregular traversal patterns and surface curvature, was not studied by previous interleaving methods.

Although numerous transmission strategies have been introduced in the literature, most of them were designed for TCP/IP based networks and did not consider the possibility of packet loss over unreliable burst channels and do not incorporate perceptual factors to enhance QoE. In a wireless transmission environment, where packet loss may occur as a result of shadowing, fading and interference [21][25], a TCP strategy, which requires retransmission, leads to further delays and thus degradation in the quality of service. Multimedia transmission over wireless networks were discussed in [12] [26], but relatively little work was done addressing 3D transmission over lossy channels. In later research, approaches for robust transmission of mesh over lossy channels [1][4] have been outlined. Chen et al. [4] assumed that the successful transmission of the base layer was guaranteed. Progressive refinement was achieved by subsequent layers. However, for lossy channels, this implies retransmission of the base layer until the data is received. Retransmission adds an overhead on bandwidth limited connections, in particular for wireless and mobile networks, as well as multicast, real-time or time-constrained applications, such as interactive virtual or augmented, as well as gaming, environments. Al-Regib et al. [1] applied an adaptive scheme using error correction codes (ECC) to correct errors at different rates. ECC uses additional bits to correct errors, which might require significant overhead if the error rate is high. In contrast to the retransmission and ECC strategies, we simply assume a certain percentage of the packets may be lost and significant overhead is not feasible given the network constraints. In our previous work [18][7][5][10], we discussed how perceptual factors could help improve the quality of transmitted data; how constrained bandwidth could be allocated efficiently between texture and mesh data; and how arbitrary meshes were transmitted in situations where packet loss was possible. We compared how various types of 3D transmission strategies fared, and how to take perceptual quality into account in designing a better strategy. We first proposed our perceptually optimized strategy based on a multi-resolution representation of the texture and mesh. We then studied the transmission strategy of vertex geometry to minimize the risk of packet loss affecting a large neighborhood, followed by a discussion of the interpolation methods used to reconstruct the meshes.

In a communication network, packet loss is often bursty [12][16][2]. This can be caused by congestion on the Internet.

For wireless networks, burst packet loss can be caused by temporary link outage or fading-induced bit error. In order to minimize the adverse effect of burst errors, an interleaving technique aims to prevent errors from affecting a large neighborhood. Examples of interleaving techniques can be found in the literature [27] but their focus is on 1D sequences and 2D images. The strategy is to apply a predefined distance to separate neighboring elements in the transmitted stream, independent of which 1D sequence or 2D images are being transmitted. Due to the arbitrary distribution of vertices on a 3D mesh, a predefined distance is not effective because it may work on a specific geometry while fail on another. We propose a mesh geometry driven interleaving scheme for transmitting arbitrary 3D meshes over burst packet loss channels. The main idea is to automatically adapt the parameters to the geometric property of a 3D mesh based on the maximum burst length being considered. We conducted experiments to verify the efficiency of the proposed method.

A. Three-Dimensional Mesh Coding

A 3D mesh is represented by geometry and connectivity [20]. An uncompressed representation, such as the 16 ASCII format [23], is inefficient for transmission. To encode mesh connectivity efficiently, every polygon should be visited and recorded in an order that is efficient for compression.

Among the many 3D mesh connectivity encoding algorithms proposed since the early 1990s [19], the valence-driven approach [22] is considered to be one of the most effective techniques for 3D mesh connectivity compression. It has a compression ratio of 1.5 bits per vertex on the average. In our technique, the mesh connectivity is encoded using this technique and transmitted over the network.

B. Our Interleaving Technique

After receiving the data over a burst packet loss channel at the client, we first reconstruct the mesh partially based on the geometry and connectivity information successfully received. The vertices are then traversed and the mesh is reconstructed following the valence-driven decoding algorithm. When a vertex with lost geometry, L, is encountered, the adjacent reconstructed vertices, whose geometry is either not lost or is interpolated previously, are used to linearly interpolate the geometry of L.

Adjacent Distance and Motivation of Mesh Geometry-Driven Interleaving

Interleaving techniques for 1D and 2D data, which have fixed adjacent distance, are not suitable for 3D meshes because of their regular row and column traversal pattern. An example of arbitrary 3D connectivity is shown in Fig. 1. A vertex is drawn as a node in the graph. The traversal order of the vertices is indicated by the numbers in the nodes, which are assigned by the valence-driven algorithm. The relative adjacent distance between two connected vertices is written on the edge. Differing from 1D and 2D data, a 3D vertex is associated with unequal adjacent distances to its neighbors. In Fig. 1, the neighbors of Vertex 20 are Vertex 21, 22, 54, 60, and 99,
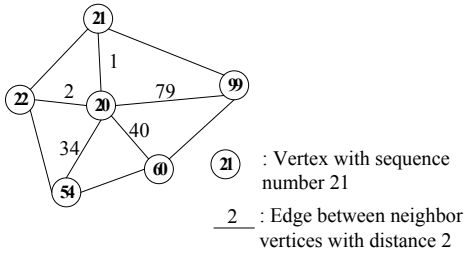
Fig. 1. An example of arbitrary neighboring vertex distance in a 3D mesh.

TABLE I
HISTOGRAM OF ADJACENT DISTANCES IN TWO 3D MESH MODELS

Index	Cow			HammerHead		
	Adjacent Distance	Edge Count	Cum. Percent	Adjacent Distance	Edge Count	Cum. Percent
0	1	2,865	32.9	1	721	32.0
1	13	282	36.1	2	66	35.0
2	12	279	39.4	9	54	37.4
3	14	210	41.8	12	51	39.7
4	11	197	44.0	10	48	41.8
5	2	162	45.9	11	48	43.9
6	15	115	47.2	49	48	46.0
7	10	105	48.4	43	47	48.1
8	97	74	49.3	42	44	50.1
9	98	71	50.1	48	43	52.0
.	99	69	50.9	8	42	53.9
.	96	68	51.7	50	42	55.7
.	82	65	52.4	13	40	57.5
.	16	63	53.1	39	40	59.3
.	88	59	53.8	46	39	61.0
.	89	58	54.5	47	36	62.6
.	100	56	55.1	45	31	64.0
.	109	54	55.7	40	30	65.3
.	…	…		…	…	
m	…	…		…	…	
Total Edges		8,706	100		2,250	100

whose adjacent distances relative to Vertex 20 are 1, 2, 34, 50 and 79 respectively.

To illustrate how adjacent distances (magnitude) vary between connected vertices on arbitrary meshes, we compute the edge count (in descending order) associated with a particular adjacent distance for two 3D models: Cow and HammerHead (Table I). Observe that most edges have the shortest adjacent distance, 1 in this case, and different meshes have different adjacent distance frequencies as shown in the Edge Count column in Table I. Fig. 2 shows the histograms starting from the second highest edge count, excluding edges with distance 1. Motivated by this observation, we introduce a probabilistic interleaving algorithm, which adaptively adjusts a control parameter for different mesh geometries (different adjacent distance histograms), taking the maximum burst length into consideration.

Although we use the valence-driven algorithm as an example to illustrate the traversal pattern on irregular meshes, our mesh-dependent interleaving technique can work with any mesh encoding algorithm.

Proposed Probabilistic Mesh-Dependent Interleaving

In order to minimize the adverse effect of burst packet loss, neighboring vertices are interleaved into separate packets. An example of how our interleaving technique works is shown in

Fig. 3. We follow the valence driven algorithm and label the vertices in traversal order, starting from 0. The network packets are denoted by PKT0, PKT1, and so on. The interleaving pattern is controlled by two parameters: B and L. Parameter B is the maximum burst length considered and is used to define the number of packets in a block. The other parameter L is a control parameter computed using Algorithm 1 described below. L and the product $L \times B$ are used to compute the minimum distance required between elements in the same block. The interleaving pattern using $B = 2$ and $L = 3$ is given in Fig. 3. Blocks are separated by dotted lines (Block 0, Block 1 and Block 2 in this example). When the maximum burst length of 2 occurs, this interleaving pattern ensures that the neighbor of a lost vertex survives the burst.
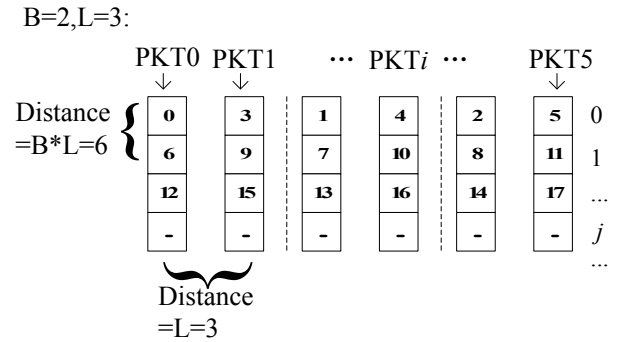

Fig. 3. Vertex interleaving for network packets using our probabilistic mesh-dependent interleaving strategy.

Let PKT_i be a network packet where $i(i = 0, 1, 2, \ldots)$ is the sequence number and $j(j = 0, 1, 2, \ldots)$ is the position number of a vertex in a network packet. The blocks of packets can be viewed as a matrix with j rows and i columns. We can associate a vertex label l (traversal order) with a (i, j) coordinate using Equation 1:

$$l = j \times B \times L + (i \bmod B) \times L + \left\lfloor \frac{i}{B} \right\rfloor \qquad (1)$$

For example, in Fig. 3, the label l of the 3rd vertex in PKT4, i.e., $i = 4, j = 2$, is calculated as follows:

$$2 \times B \times L + (4 \bmod B) \times L + \left\lfloor \frac{4}{B} \right\rfloor$$
$$= 2 \times 2 \times 3 + (4 \bmod 2) \times 3 + \left\lfloor \frac{4}{2} \right\rfloor$$
$$= 14$$

The $j \times i$ matrix has L blocks and each block has B packets. In Equation 1, the first term $j \times B \times L$ indicates that each row has $B \times L$ elements in the $j \times i$ matrix. In the second term, $(i \bmod B) \times L$, $(i \bmod B)$ denotes the packet sequence number in each block and L represents the adjacent distance of the vertices with the same j value in each block. The third term $\left\lfloor \frac{i}{B} \right\rfloor$ indicates the block number in the matrix. We have the following lemma.

Lemma 1: The adjacent distance between two vertices in the same block is a multiple of L and the adjacent distance between two vertices in each packet is a multiple of $B \times L$.
Proof: Suppose two vertices in a block are (i_1, j_1) and (i_2, j_2).
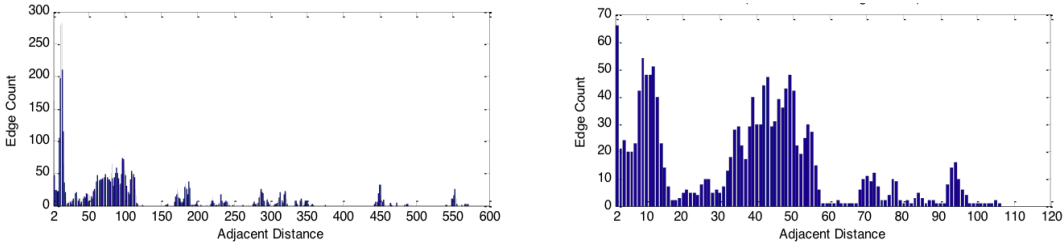
Fig. 2. The histograms of two 3D mesh models: Cow (Left) and HammerHead (Right), starting from the second highest edge count, show that adjacent distance frequencies are mesh geometry-dependent.

Since vertices in the same block have the same block number, we obtain $\left\lfloor \frac{i_1}{B} \right\rfloor = \left\lfloor \frac{i_2}{B} \right\rfloor$. From Equation 1, the adjacent distance between two vertices in the same block is:

$$
\begin{aligned}
& |j_1 \times B \times L + (i_1 \bmod B) \times L + \left\lfloor \frac{i_1}{B} \right\rfloor \\
- \; & (j_2 \times B \times L + (i_2 \bmod B) \times L + \left\lfloor \frac{i_2}{B} \right\rfloor| \\
= \; & |(j_1 - j_2) \times B + (i_1 \bmod B) - (i_2 \bmod B)| \times L,
\end{aligned}
$$

which is a multiple of L.

The vertices in each packet have the same i. Suppose that two corresponding vertices are $(i, j1)$ and $(i, j2)$. Therefore, the adjacent distance between the vertices in each packet is:

$$
\begin{aligned}
& |j_1 \times B \times L + (i \bmod B) \times L + \left\lfloor \frac{i}{B} \right\rfloor \\
- \; & (j_2 \times B \times L + (i \bmod B) \times L + \left\lfloor \frac{i}{B} \right\rfloor)| \\
= \; & |j_1 - j_2| \times B \times L
\end{aligned}
$$

which is a multiple of $B \times L$.

Note that the above interleaving pattern illustrates the minimum number of packets that satisfies the adjacent distance restriction. A higher number of packets can be transmitted by partitioning each block horizontally and by dividing each packet (PKT_i) into smaller packets.

Computation of Control Parameter L

We put B packets in a block, inside which the adjacent distance between corresponding elements at the same j position in consecutive packets is L. For example, in Fig. 3, PKT0 and PKT1 belong to a block, while PKT2 and PKT3 belong to another block. When a burst error occurs, the maximum loss is one block or two consecutive packets. The parameter L is computed so that vertices in the same neighborhood are distributed into different blocks. Suppose a burst error occurs and PKT0 and PKT1 are lost, then so is Vertex 3; but, Vertex 3 can be reconstructed based on the geometry information in its neighbors which are not lost. However, if its neighbors are transmitted in the same block, then a larger piece of the surface geometry is lost and reconstruction based on interpolation is less accurate. We choose a proper value for L based on Theorem 1, to avoid adjacent vertices being assigned to the same block.

Theorem 1: If the adjacent distance between two neighboring vertices in a mesh is not a multiple of L, the vertices in the same block are not adjacent in the mesh.
Proof: According to Lemma 1, the adjacent distance between two vertices in the same block is a multiple of L. If the adjacent distance between any neighboring vertices in the mesh is not a multiple of L, then the vertices in the same block cannot be adjacent in the mesh.

To avoid adjacent vertices being lost in a bursty channel, we define a confidence level Γ to guarantee that at least this percentage of adjacent elements will not appear in the same block. This is achieved by choosing an appropriate L value based on the mesh geometry using Algorithm 1 (given in Appendix A), where we find the minimal L, which is not divisible by the adjacent distances covered by the cumulative percentage Γ, in order to use as few packets as possible for network efficiency. If there are too many vertices for transmission in $L \times B$ packets, the number of packets can be increased to a multiple of $L \times B$. For example, if the number of vertices is V and the maximum number of vertices that could be transmitted in a network packet is v, the number of packets used is $\left\lceil \frac{V}{L \times B \times v} \right\rceil \times L \times B$. In this case, to map the vertices into network packets, we just replace B by $\left\lceil \frac{V}{L \times B \times v} \right\rceil \times B$. Once a proper L value is chosen, vertices can be assigned into network packets and retrieved from network packets in a proper order. (Algorithms 2 & 3 are given in Appendix A).

III. EXPERIMENTAL RESULTS AND ANALYSIS

Quantization, prediction and statistical coding are used to compress the geometry data. However, a common characteristic for all transmission strategies, for lossy channels without error correction or re-transmission, is that the lost data can only be reconstructed based on information available within the same packet. The reason is because of the assumption that other packets can be lost. Since the vertices within the same packet in our technique are not adjacent neighbors, the compression rate cannot be as efficient as the valence-driven approach [22], which has a compression ratio of 1.5 bits per vertex on the average for encoding mesh connectivity. Nevertheless, the valence-driven approach is designed for reliable networks. In general, the compression rate of strategies for lossy channels is less efficient than those for reliable channels. Also, note that the encoding of connectivity costs only about 10% of what is needed for encoding geometry [31]. The overhead of using our technique to overcome the
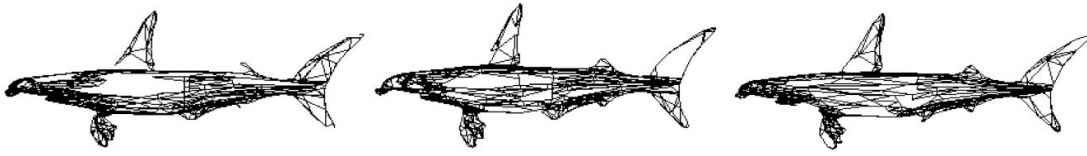
Fig. 4. Comparison of results: (Left) no interleaving; (Middle) fixed-parameter interleaving; (Right) our mesh geometry-dependent interleaving.

TABLE II
RECONSTRUCTION ERROR AFTER TRANSMITTING THROUGH BURST PACKET LOSS CHANNELS USING A CONFIDENCE LEVEL OF 80%

Model (Vertex Number)	Average burst length $\bar{B}$ and B	Reconstruction Error (Hausdorff distance of metro tool [30])								
		Overall loss probability =0.25			Overall loss probability =0.50			Overall loss probability =0.75		
		No Inter-leaving	Fixed Inter-leaving	Our Inter-leaving	No Inter-leaving	Fixed Inter-leaving	Our Inter-leaving	No Inter-leaving	Fixed Inter-leaving	Our Inter-leaving
Armadillo (1,752)	$\bar{B}=2,B=4$	6.414548	5.072178	3.653101	6.414548	5.07217*	6.512835	9.990111	9.814941	8.795731
	$\bar{B}=3,B=6$	7.919993	5.754521	3.694067	8.227816	7.501959	5.749495	23.776726	11.85642	11.16828
	$\bar{B}=5,B=10$	8.642299	3.817449	3.374231	9.784201	6.51283*	10.227290	12.583591	18.94281	7.836614
	$\bar{B}=8,B=16$	7.314794	7.430123	3.969433	10.111954	8.704135	7.041734	9.040297	11.12948	9.590316
	$\bar{B}=16,B=32$	9.772218	2.925999	2.613735	9.074914	4.70990*	7.430123	14.351581	12.97940	10.65304
Bunny (2,503)	$\bar{B}=2,B=4$	0.003803	0.003299*	0.004887	0.005294	0.00324*	0.004177	0.007525	0.00642*	0.009103
	$\bar{B}=3,B=6$	0.005332	0.002247*	0.002675	0.005333	0.004212	0.003051	0.010152	0.00691*	0.009133
	$\bar{B}=5,B=10$	0.005958	0.002724	0.002321	0.004219	0.004020	0.003440	0.008195*	0.008510	0.008642
	$\bar{B}=8,B=16$	0.006558	0.002724	0.002015	0.011808	0.006049	0.003902	0.006706*	0.010426	0.007836
	$\bar{B}=16,B=32$	0.007189	0.002719	0.002430	0.009112	0.004455	0.003273	0.012534	0.005786	0.013182
Cow (2,904)	$\bar{B}=2,B=4$	0.028558	0.028422	0.015234	0.025717*	0.028361	0.028361	0.028803*	0.033525	0.037176
	$\bar{B}=3,B=6$	0.028361	0.014078	0.011498	0.052100	0.02857*	0.028361	0.037924*	0.039583	0.061668
	$\bar{B}=5,B=10$	0.028543	0.028352	0.012177	0.035720	0.022341	0.021634	0.036498	0.036604	0.036384
	$\bar{B}=8,B=16$	0.030541	0.016767*	0.017503	0.039978	0.028422	0.023404	0.069975	0.035720	0.038703
	$\bar{B}=16,B=32$	0.049300	0.018703	0.010858	0.074655	0.029358	0.028422	0.080797	0.048883	0.036678
Dinosaur (5,000)	$\bar{B}=2,B=4$	1.725558	1.003255*	1.057358	2.239904	1.744817	1.519358	2.275198	2.286047*	2.580115
	$\bar{B}=3,B=6$	2.488605	0.797432*	1.032733	2.488605	1.48978*	1.609889	14.732584	3.136629	2.604890
	$\bar{B}=5,B=10$	1.871356	0.941308	0.892378	2.688417	1.64225*	1.677934	3.775949	2.708860	2.219183
	$\bar{B}=8,B=16$	2.498059	1.144875	0.869984	2.956688	2.046298	1.466011	4.056268	3.990355	2.776798
	$\bar{B}=16,B=32$	4.052977	0.846244	0.746002	2.243943	1.919845	1.617296	3.825114	3.33772*	4.612769
Hammer Head (752)	$\bar{B}=2,B=4$	0.042211	0.030476	0.027487	0.061547	0.046718	0.045592	0.058907*	0.068149	0.080285
	$\bar{B}=3,B=6$	0.059997	0.019541*	0.025401	0.063380	0.063042	0.033602	1.114240	0.04851*	0.069737
	$\bar{B}=5,B=10$	0.032809	0.029024	0.023440	0.032253	0.062119	0.036285	0.080349	0.075982	0.052423
	$\bar{B}=8,B=16$	0.080349	0.025024*	0.027417	0.117185	0.03770*	0.047700	0.050160	0.065599	0.037467
	$\bar{B}=16,B=32$	0.080349	0.019309	0.015704	0.027887	0.068105	0.029026	0.039274	0.070234	0.068480
Venus (8,268)	$\bar{B}=2,B=4$	0.014420	0.008777*	0.009574	0.018623	0.01166*	0.012738	0.020197	0.019288	0.016522
	$\bar{B}=3,B=6$	0.019706	0.010910	0.007925	0.019981	0.01033*	0.013955	0.023202	0.01970*	0.023027
	$\bar{B}=5,B=10$	0.020908	0.007976	0.008459	0.014658	0.01046*	0.014187	0.021176	0.01944*	0.019928
	$\bar{B}=8,B=16$	0.017097	0.008362	0.006730	0.022764	0.00970*	0.009855	0.020809	0.027700	0.026078
	$\bar{B}=16,B=32$	0.017991	0.011013	0.008412	1.041932	0.021760	0.016782	0.040121	0.022040	0.021407
Armadillo (34,594)	$\bar{B}=2,B=4$	0.005483	0.003963*	0.004415	0.006853	0.005660	0.005237	0.008888*	0.010021	0.009696
	$\bar{B}=3,B=6$	0.005279	0.003519*	0.003996	0.008026	0.005164*	0.005467	0.012097	0.011502*	0.012722
	$\bar{B}=5,B=10$	0.006018	0.003619	0.003260	0.012782	0.005086*	0.005867	0.012901	0.008806*	0.009424
	$\bar{B}=8,B=16$	0.006173	0.002623*	0.003832	0.023105	0.005736*	0.005920	0.020554	0.009275	0.006168
	$\bar{B}=16,B=32$	0.007424	0.002558	0.002385	0.013594	0.003980	0.003886	0.021538	0.006087*	0.011003
Horse (48,485)	$\bar{B}=2,B=4$	0.001561	0.001336	0.001059	0.001319	0.001510	0.001177	0.001855*	0.002389	0.002328
	$\bar{B}=3,B=6$	0.000911*	0.001320	0.001343	0.001718	0.001550	0.001191	0.002490	0.002269	0.002352
	$\bar{B}=5,B=10$	0.001311	0.001039	0.000904	0.002421	0.001181*	0.002482	0.002272	0.001920*	0.002134
	$\bar{B}=8,B=16$	0.001100	0.000741*	0.001060	0.002880	0.001379	0.001378	0.002906	0.001322*	0.002341
	$\bar{B}=16,B=32$	0.001192	0.000904	0.000793	0.001852	0.001238	0.001156	0.004214	0.001156*	0.002923

lossy transmission without error correction and retransmission is that it needs $B+1$ (the maximal burst length) reference vertices instead of one reference vertex for compression in the lossless transmission. For 3D models with thousands of vertices, this overhead is relative small. Our comparisons with no-interleaving and fixed-interleaving methods were based on the same available bandwidth and no additional overhead was incurred in our technique, which demonstrates a novel interleaving pattern robust to burst packet loss that outperformes other methods in general. Compared to re-transmission, our technique requires less overhead.

In order to verify our probabilistic mesh geometry-dependent interleaving technique, we conducted experiments using different arbitrary 3D meshes. A confidence level of 80% was used throughout our experiments. Similar to [2][24], we simulated the burst packet loss channel by a two-state Markov model known as the Gilbert model [14]. The overall loss probability of the channel was set to 25%, 50%, and 75%. The average burst length $\bar{B}$ of packet loss was set to 2, 3, 5, 8, and 16 packets.

Fig. 4 shows the reconstructed HammerHead fish mesh after transmitting (Left) with no interleaving, (Middle) with

fixed-parameter interleaving, and (Right) with our mesh geometry-dependent interleaving using a burst packet loss channel with overall loss probability equal to 25% and burst length equal to 5. Note that when no interleaving or fixed-parameter interleaving was applied, a larger area of the geometry was lost. When our mesh-dependent interleaving was applied, the lost vertices were distributed more evenly.

More examples are shown in Fig. 7. The meshes were transmitted with 50% and 75% packet loss respectively and with a burst length equal to 16. The results show that smoothness on the object surface was preserved even at 50% packet loss. At 75% packet loss, smoothness on the object surface deteriorated but the overall shape of the objects was still preserved. It is important to point out that the visual quality of the reconstructed mesh is affected by the original density of the mesh [9]. Less dense meshes have higher possibility of displaying artifacts due to burst packet loss because of the larger relative change (relative change and Just-Noticeable-Difference will be discussed in the next Section). The Dinosaur mesh had a high density of 5000 vertices, and the visual quality of the reconstructed mesh at 75% packet loss was satisfactory. The other meshes with lower density: Armadillo, Bunny and Cow, showed obvious degradation in visual quality after reconstruction at 75% packet loss.

We use the metro tool [11] to measure errors between original models and reconstructed models based on Hausdorff distance. The metro tool applies surface sampling and *point-to-surface* distance computation. It samples vertices, edges and faces by taking a number of samples that is approximately 10 times the face number. The reconstruction errors of three transmission strategies: without interleaving, with fixed-parameter interleaving ($L = 64$), and with our mesh geometry-driven interleaving using a burst packet loss channel, are shown in Table II using 6 models. The lost geometry was interpolated as described in Section II.B above. Among all the 120 cases, the *no interleaving* approach had the smallest reconstruction error in 9 cases (7.50%), the *fixed-parameter interleaving* method had the smallest reconstruction error in 41 cases (34.17%), and our mesh geometry-driven interleaving technique had the smallest reconstruction error in the remaining 70 cases (58.33%). The comparison results verify that in general the interleaving methods perform better than a strategy with no interleaving, and our probabilistic geometry-driven interleaving technique performs better than the fixed-parameter interleaving technique.

While some applications are able to tolerate higher loss, others may need to restrict the loss below a specified level (a psychophysical interpretation of this threshold is given in Section IV). We improve the reconstructed mesh further by applying our curvature-driven probabilistic strategy [32] to safeguard visually significant structures on the 3D surface. Critical mesh features, with high curvature, like sharp edges and corners are assigned higher priority to improve the reconstruction results. A curvature index of a normalized mesh, describing the relative elevation of a visually significant surface feature from the average plane formed by its adjacent vertices, is added to the transmission pipeline and stored in a predefined adjacent vertex when the probability of packet loss

becomes unacceptable. For model reconstruction purpose, a lookup table recording a number of elevation levels is used. Based on the 3D models tested, sixteen levels are generally sufficient and each index requires 4 bits to encode. Thus, the actual bandwidth taken up by the curvature indices is dependent on the probability of packet loss, the tolerance of an application, and is mesh geometry-driven. For relatively smooth surfaces, curvature index is not necessary.

We assume that there is a *significant curvature level* or Just-Noticeable-Difference (JND) threshold (Fig. 5) beyond which (to the left of the dotted vertical line in Fig. 5) linear interpolation can be applied without generating noticeable visual artifacts to the HVS, and thus no extra protection on those curvature is necessary. The assumption is based on our user study observation of JND on 3D objects [8][6].
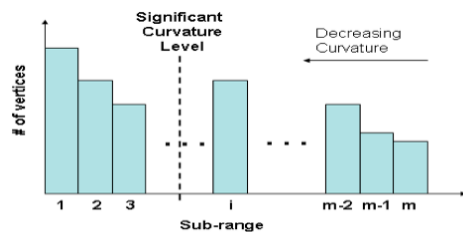

Fig. 5. An example of curvature histogram: surface feature points are prioritized (with more significant ones on the right) based on their visual impacts.

IV. THE PROPOSED JND VISUAL DISCRIMINATION METRIC

Although earlier study [8][6] observed that a JND threshold is associated with 3D objects visualization, an average threshold may not be sufficient for some applications, which need to adjust the percentage of target users, or select a minimum or maximum threshold so that the target users will be satisfied with the quality delivered over the transmission pipeline. We thus propose in this section a visualization discrimination model, based on which an application can alter appropriate parameters to estimate the percentage of user population meeting certain criteria.

Given a 3D model as stimulus, the perceptual impact varies from one viewer to another. The probability that a viewer i can correctly discriminate the visual quality of the stimulus $P(\theta_i)$ is dictated by the chance of guessing correctly (c) and the relative change ($\tau(d_x, d_z)$) generated on the model surface [8][6]. This psychophysical concept follows the same spirit as how ability or other hypothesized traits are described in Item Response Theory (IRT) [17]. We model the probability of correctly discriminating a stimulus by a viewer i as:

$$P(\theta_i) = c + \frac{1 - c}{1 + e^{d(\theta_i - \tau(d_x, d_z))}} \qquad (2)$$

where d is a constant and $\tau(d_x, d_z)$ is an implicit function.

Given an original surface S_i, our approach first computes the medial axis M_i of S_i. The new simplified surface S_k can be generated through edge collapse [15][13] replacing vertices $Vx, Vy \in S_i$ with a new vertex $V_z \in S_k$, or through vertex
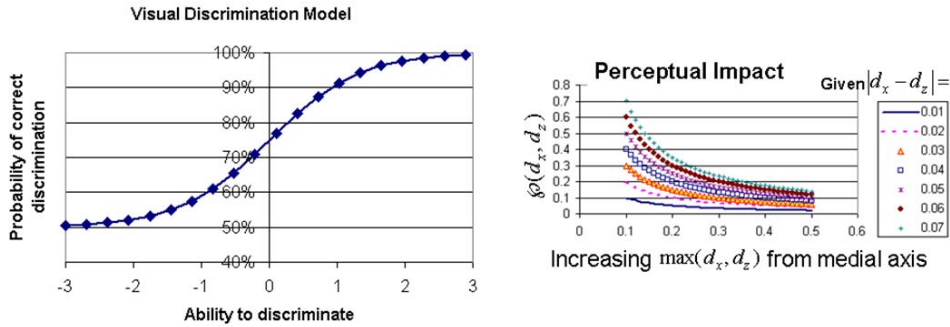
Fig. 6.   6 (Left) given a range of discrimination ability, the probability of correct discrimination follows a S-shape curve; and (Right) given a relative change (stimulus), the perceptual impact function defines a diminishing impact when the change occurs farther away from the medial axis on the 2D projection - $\max(d_x, d_z)$ gets bigger.

removal followed by hole-filling [3]. In the latter case, the center of the average plane formed by the neighbors of the vertex removed is denoted as V_z. If multiple changes occur, removing connected neighbors, the final V_z is chosen. Given a surface change starting at V_x and terminating at V_z, the implicit function is defined by:

$$\tau(d_x, d_z) = \frac{|d_x - d_z|}{\max(d_x, d_z)} \qquad (3)$$

where d_x and d_z are the shortest distances of V_x and V_z from M_i respectively.

The S-shape function (Eq. 2) is often called the sigmoid curve or psychometric curve (Fig. 6(Left). The bottom left convergence of the curve represents the highest difficulty to discriminate relative to a viewer's ability. In this case, a viewer with low discrimination ability has to guess and the probability of guessing correctly based on the two-alternative-forced-choice (2AFC) [6] setting is 50% after a sufficiently large number of discrimination tests have been conducted. The top right convergence indicates that a viewer with high ability can correctly discriminate 100% of the time. The point of inflection (75% correctness) of the psychometric curve provides the average threshold. Above the threshold and moving towards the right, the probability of a viewer discriminating correctly increases, while moving towards the left below the threshold, the probability of a viewer discriminating correctly decreases ending up with guessing.

We define this model to explain the perceptual impact caused by deleting or adding a vertex on a mesh surface, filling the void by interpolation similar to the reconstruction process after packet loss. The left convergence corresponds to guessing by the viewer when the change is not noticeable. Moving along the curve from left to right indicates that an increasingly higher percentage of viewers can correctly discriminate as the change becomes more noticeable, and ultimately when the change is obvious 100% of the viewers can discriminate correctly. The point of inflection is defined as the JND threshold of relative change. The HVS responds to relative change instead of absolute change when discriminating curvature artifacts on 3D surfaces [8][6]. $|d_x, d_z| / \max(d_x, d_z)$ represents the relative change on a 3D surface.

We are interested in the threshold where the relative change

of a stimulus is just noticeable. Note that in the JND Visual Discrimination Model the perceptual impact generated by the relative change $\tau(d_x, d_z)$ diminishes when the change occurs farther away from the medial axis of the mesh (Fig. 6(Right) on the 2D projection, when $\max(d_x, d_z)$ gets bigger. This characteristic of diminishing visual impact is invariant to changing distance from the medial axis (illustrated by the various curves in Fig. 6(Right). Our JND quality estimation formulation captures the observations obtained in subjective user studies [6].

A detailed discussion of the JND threshold and comparisons with other psychophysical findings in non-3D cases were given in [8] [6], where the advantage of using JND in visualizing 3D models was validated by conducting subjective user studies. However, no mathematical formulation was published before. The formulation contains the important parameters used in the subjective user studies [6], namely the probability of guessing and relative change. The customization of these parameters is application dependent. By adjusting the parameters, the visual discrimination curve will become steeper or flatter, and thus increase or decrease the percentage of target users, as well as the minimum and maximum thresholds required in an application. The JND formulation helps to suppress visually redundant data from the transmission pipeline, which will not improve QoE.

V. CONCLUSION

In this paper we introduced a mesh geometry-driven interleaving and surface curvature evaluation strategy for 3D meshes transmitted over unreliable networks taking burst packet loss into account. We also formulated a QoE metric on Just-Noticeable-Difference (JND) of relative change, to preserve visually significant and suppress visually redundant data, based on our earlier qualitative observations obtained from subjective user studies. Experimental results were shown with arbitrary meshes to demonstrate that the approach works well even when a high percentage of packets are lost. We distributed neighboring vertex information into different packets to minimize the risk of lost data affecting a large neighborhood, and assigned higher priority to surface features with higher curvatures, which generates bigger visual impacts if lost during transmission. Experiments on models with different
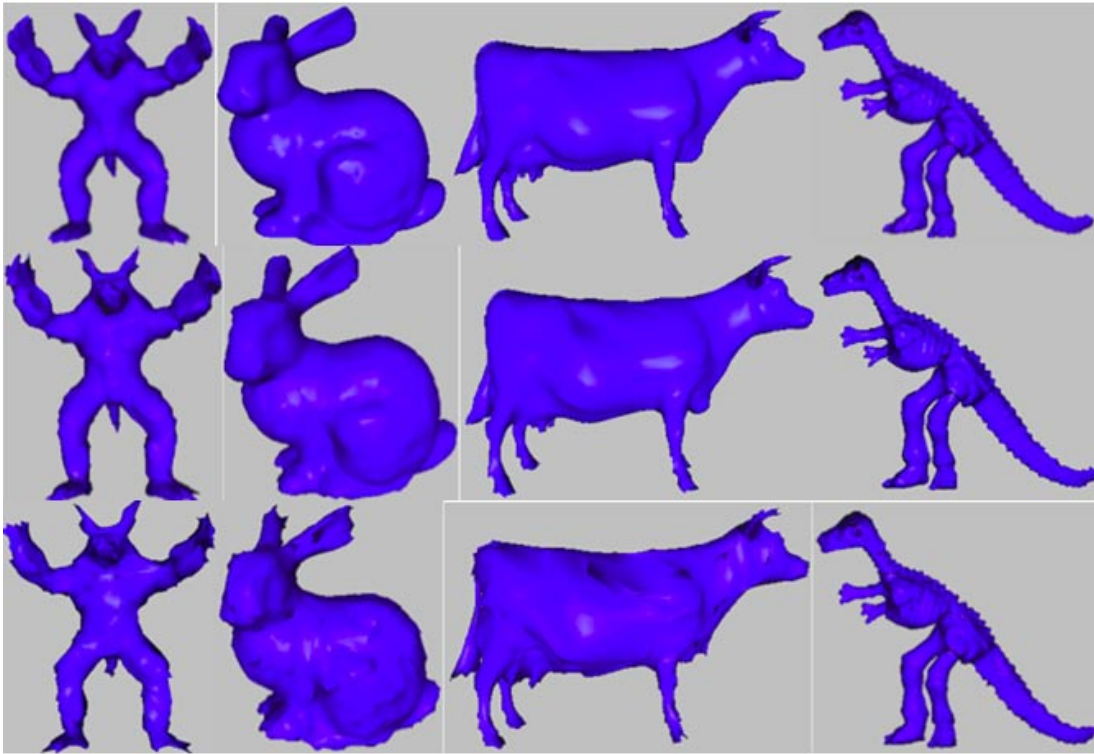
Fig. 7. Different models: Armadillo(1st,$1,752$vertices);bunny(2nd column, $2,503$ vertices); cow(3rd column, $2,904$ vertices) and dinosaur (4th column, $5,000$ vertices), reconstructed using our interleaving strategy. The original models are shown in the first row. Reconstructed models at 50% and 75% packet loss (average burst length = 16) are shown in the second and third rows respectively.

resolutions show satisfactory outcome outperforming other interleaving techniques in general, indicating that smoothness on the object surface was preserved even at 50% packet loss. In future work we will apply our JND model to enhance the QoE of stereoscopic and multi-viewpoint 3D, as well as animation data transmitted over mobile networks and devices.

APPENDIX A

Algorithms 1 - Computing parameter L
1. Given a mesh, obtain its histogram H (Table I);
2. Set $t = 0$ and while ($H.CumPercent[t] < \Gamma$) t++;
3. Set $MaxDist$ = the longest adjacent distance in H;
Given B, search the minimum L from 2 to $\lceil \frac{MaxDist}{B} \rceil$ satisfying confidence level Γ:
4. $L = 2$;
5. while ($B * L <= MaxDist$) do
6.   found = true;
7.   for ($m = 0$ to t) do
8.   if (L is divisible by H.distance[m])
9.   found = false;
10.   break;
11.   // for m;
12.   if (found) // found satisfying L
13.   break;
14. L++; // increase L
15. // while;
16. return L;

Algorithms 2 - Computing (i, j) coordinates for a vertex

Computing j:
Given a traversal order l of a vertex, the position j of the vertex in a packet is given by: $j = \lfloor l/(B \times L) \rfloor$ according to Equation 1;

Computing i:
We first get the packet distribution with j equal to 1 in network pakcets using Equation 1.
1.   for ($k = 0$ to ($B * L - 1$)) do
2.   Packet[k] = ($k \bmod B$) $* L + \lfloor k/B \rfloor$
3.   //for k
Then we find the matcing i,
4. nMod = $v \bmod (B * L)$;
5. for ($i = 0$ to ($B * L - 1$))
6.   if(nMod == Packet[i]) then
7.   break;
8.   // for i
9. return i;

Algorithm 3 - Computing vertex travesal order v given (i, j)

Use Equation 1
1. return($j * B * L + (i \bmod B) * L + \lfloor i/B \rfloor$);

ACKNOWLEDGMENT

REFERENCES

[1] G. Al-Regib, Y .Altunbasak and I Rossignac, "Error-resilient Transmission of 3D Models," ACM Trans. on Graphics, Vol.24, No.2, pp.182-208,2005.

[2] Q. Au,Y. Pei and J.W. Modestino, "Robust H.264 Video Coding and Transmission over Bursty Packet-Loss Wireless Networks," *IEEE 58th Vechcular Technology Conference (VTC)*,Oct 2003.

[3] P.H. Charles and C. Hansen, "Geometric Optimization," *Proc. Visualization* 1993, pp.189-195.

[4] Z. Chen,B. Bodenheimer and J. Barnes, "Robust Transmission of 3D Geometry over Wireless Networks," *Web3D*,2003, pp.161-172

[5] I. Cheng and A. Basu, "Perceptually Optimized 3D Transmission over Wireless Network," *IEEE Trans. on Multimedia,* Vol.9, No.2, pp.386-396, Feb 2007.

[6] I. Cheng and P. Boulanger, "A 3D Perceptual Metric using Just-Noticeable-Difference," *EUROGRAPHICS 2005*, Aug/Sep Dublin Ireland.

[7] I. Cheng and P. Boulanger, "Adaptive Online Transmission of 3D TexMesh Using Scale-Space and Visual Perception Analysis," *IEEE Trans. on Multimedia*, Vol.8, No.3, pp.550-563, Jun 2006.

[8] I. Cheng and P. Boulanger, "Feature Extraction on 3D TexMesh Using Scale-Space Analysis and Perceptual Evaluation," *IEEE Trans. on Circuits and Systems for Video Technology* Special Issue, Vol.15, No.10, Oct 2005.

[9] I. Cheng, R. Shen, X. Yang and P. Boulanger, "Perceptual Analysis of Level-of-Detail: The JND Approach," *International Symposium on Multimedia*, Dec 2006, pp.533-540, San Diego.

[10] I. Cheng, L. Ying, K. Daniilidis and A. Basu, "Robust and Scalable Transmission of Arbitrary 3D Models over Wireless Networks," *EURASIP Journal on Image and Video Processing IVP/890482*, 30 pages, 2008.

[11] P. Cignoni, C. Rocchini and R. Scopigno, "Metro: Measuring Error on Simplified Surfaces," *Computer Graphics Forum, Black-Well Publishers,* Vol.17(2), Jun 1998, pp.167-174.

[12] A. Ganz, Z. Ganz and K. Wongthayarawat, "Multimedia Wireless Networks: Technologies, Standards and QoS," *Prentics Hall PTR*,2004

[13] M. Garland and P. Heckbert, "Simplification using Quadric Error Metrics," *Proc. SIGGRAPH 1997*, pp.209-216

[14] E.N. Gilbert, "Capacity of a Burst-Noise Channel", *Bell System Tech.*, Vol.39, pp.1253-1265, Sep 1960.

[15] H. Hoppe, "Progressive Meshes," *Proc. SIGGRAPH 1996*, pp.99-108

[16] Y.J. Liang, J.G. Apostolopoulos and B. Girod, "Analysis of Packet Loss for Compressed Video: Does Burst-Length Matter?" *IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, Apr 2003.

[17] V. Linden and R. Hambleton,"Handbook of Modern Item Response Theory", *London, Springer Verlag*, 1997.

[18] Y. Pan, I. Cheng and A. Basu, Quality Metric for Approximating Subjective Evaluation of 3D Objects, *IEEE Trans. on Multimedia*, Vol.7, No.2, pp.269-279, Apr 2005.

[19] J. Peng, C.S. Kim and C.C. Kuo, Technologies for 3D Mesh Compression: A Survey, *Journal of Visual Communication and Image Representation*, 16(2005) pp.688-733, 2005.

[20] G. Taubin, 3D Geometry Compression and Progressive Transmission, *Proc. EUROGRAPHICS*, 2009.

[21] D. Tse and P. Viswanath, Fundamentals of Wireless Communication, *Cambridge University Press*, 2005.

[22] C. Touma and C. Gotsman, Triangle Mesh Compression, *Graphics Interface*, 1998, pp. 26-34.

[23] The Virtual Reality Modeling Language (16). ISO/IEC 14772-1, 1997.

[24] S.H. Yang and J.M. Lin, Burst-Packet-Loss Concealment for MPEG-2 Video, *Taiwan Telecommunication Workshop*, Dec 2002.

[25] G. Wittenburg and J. Schiller, A Quantitative Evaluation of the Simulation Accuracy of Wireless Sensor Networks, *Proc. Of 6. Fachgesprach Drahtlose Sensornetze der GI/ITG-Fachgruppe Kommunikation und Verteilte Systeme*, Aachen, Germany, Jul 2007.

[26] D. Wu and R. Negi, Effective Capacity: A Wireless Channel Model for Support of QoS, *IEEE Trans. on Wireless Communications*, 2002.

[27] X.M. Zhang, Y.Q. Shi and W.Q. Xu, Optimal 2-D Interleaving for Robust Multimedia Transmission, *International Conference on Information Technology: Research and Education (ITRE)*, Aug 2003.

[28] Y.Q. Shi, X.M. Zhang, Z.-C. Ni and N. Ansari, Interleaving for Combating Bursts of Errors, *IEEE Circuits and Systems Magazine*, 2004.

[29] J.-Y. Choi and J. Shin, Content-Aware Packet-Level Interleaving Method for Video Transmission Over Wireless Networks, *Wired/Wireless Internet Communications, Lecture Notes in Computer Science*, Volume 3510/2005, 2005.

[30] R. Hasimoto-Beltran and A. Khokhar, Pixel level interleaving schemes for robust image communication, *Seventeenth IEEE Symposium on Reliable Distributed Systems*, 1998.

[31] J. Rossignac, "Surface Simplification and 3D Geometry Compression", In *Handbook of Discrete and Computational Geometry (second edition), CRC Press*, 2003.

[32] I. Cheng, L. Ying and K. Daniilidis, A Curvature-Driven Probabilistic Strategy for Transmission of Arbitrary 3D Meshes over Unreliable Networks, *4th Int'l Symposium on 3DPVT*, Atlanta, USA, June 2008.

Irene Cheng (M'02–SM'09) received the Ph.D. degree in computing science from the University of Alberta, Edmonton, Canada. She is currently the Scientific Director of the Multimedia Research Centre and an Adjunct faculty in the Department of Computing Science, University of Alberta. She also holds an Adjunct position in the Faculty of Medicine and Dentistry. She is the Chair of the IEEE NCS EMBS Chapter, the Chair of the Communications Society MMTC 3DRPC Interest Group, and a board member of the SMC TC on Human Perception in Vision, Graphics and Multimedia. Dr. Cheng was the General Chair of IEEE ICME 2011 and was awarded a visiting professorship at INSA, Lyon in 2011. She held or was offered several fellowships from iCORE, NSERC and other organizations in the past. She has two books and more than 100 papers published in international journals and peer-reviewed conferences. Her research interests include incorporating human perception—Just-Noticeable-Difference—following psychophysical methodology, to improve multimedia transmission techniques. She is also engaged in research on 3-D TV and perceptually motivated technologies in multimedia, including online education.
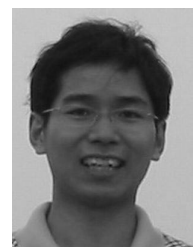
Lihang Ying received his Ph.D. degree in computing science from the University of Alberta, Canada in 2008. He is now working in the City of Edmonton, Alberta, Canada. His research interests include multimedia delivery over unreliable networks, bandwidth monitoring, perceptual factors, and transmission strategies.

Anup Basu (M'90–SM'02) received the Ph.D. degree in computer science from the University of Maryland, College Park. He was a Visiting Professor at UC Riverside, a Guest Professor in Graz, and the Director at the Hewlett-Packard Lab, at the University of Alberta, where, since 1999, he has been a Professor at the Department of Computing Science, and is currently an Alberta Innovates Research Chair. He originated the use of foveation for image, video, stereo, and graphics communication in the early 1990s, an approach that is now widely used in industrial standards. He also developed the first correspondence free active camera calibration and tracking methods, and a single camera panoramic stereo.