FACTA UNIVERSITATIS
Series: **Automatic Control and Robotics** Vol. 13, N° 2, 2014, pp. 107 - 116

A RUNLENGTH CODED ADAPTIVE QIM FOR THE CULTURAL HERITAGE 3D MODELS AUTHENTICATION*

UDC ((621.391:004)+513)):(347.6+008)

Dejan Rančić, Bata Vasić

University of Niš, Faculty of Electronic Engineering, Republic of Serbia

Abstract. *We describe a simple and effective quantization scheme for a digital 3D cultural heritage models authentication and watermarking. It is based on runlength coding which converts a class of deletion channels that have infinite memory into memoryless channels. We consider a novel application of this technique in multimedia watermarking and authentication using quantization index modulation operating on the three dimensional mesh vertices, which are invariant to geometric and topological transformation. Vector of the vertex indices is extracted from a huge digital 3D model using our powerful vertex extraction tool. The coding recovers the data hidden in the vertices removed by the process of mesh simplification.*

Key words: *3-D geometry, 3D mesh, authentication, cultural heritage, quantization.*

1. INTRODUCTION

In the era of digitalization of cultural heritage an authorship protection issue still remains unsolved. Moreover, very quick grow of the worldwide user number increases a risk of unauthorized use or even misuse of such digital models. Thereby, all kind of digitalized models are exposed to the risk including one-dimensional (1D) audio records, two-dimensional (2D) images and drawings, as well as a recent achievement of three-dimensional (3D) geometrical representations of the cultural heritage. Protection of the last one mentioned is insufficiently researched due to the brand new problems with the impossibility of an unique 3D representation and the lack of synchronization of protection

---

Received November 25, 2014
Corresponding author: Dejan Rančić
Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Niš, Republic of Serbia
E-mail: dejan.rancic@elfak.ni.ac.rs
* **Acknowledgement**: This paper was realized as a part of the projects "Digitalization and visualization of the most important museum collections of traditional costumes and jewellery by introducing the new 3D technology (VIRTUAL MUSEUM)" co-founded by European Union through Bulgaria - Serbia IPA cross border programme, Ref. no 2007CB16IPO006-2011-2-92.

data in an infinite 3D space. The faced problem dealing with such spatial data is analogous to the synchronization problem in telecommunication channels.

Synchronization errors occur also in many data transmission and storage applications. Time misalignment between the transmitted and received data bits is difficult to handle, and even though such channels have been studied for decades, they are not characterized in terms of channel capacity. The main obstacle is the fact that channels with synchronization errors have infinite memory, i.e. a synchronization error affects all subsequent symbols. In this paper, we introduce a coding scheme by which it is possible to transform certain channels with synchronization errors into memoryless channels.

We apply this concept to a secure watermarking and authentication scheme with guaranteed robustness to mesh optimization and simplification proposed in [1]. It combines sparse quantized index modulation (QIM) for data hiding with run-length modulated LDPC codes for recovering deleted watermark bits. The first step in the watermark recovery is detection of QIM bits, while the second step is deletion correction. A similar scheme was recently proposed by Coumou and Sharma [2] for audio signal watermarking. However, their method is based on the insertion/deletion/substitution correction scheme of Davey and MacKay [3] which combines marker codes for providing synchronization and LDPC codes for error correction. We provide simulation-based analysis of the watermark robustness subject to mesh simplification. Mesh simplification deletes vertices, and the role of our codes is to mitigate this effect.

The rest of the paper is organized as follows. The second section gives relevant QIM and notations and definitions as well as solution for increasing watermark robustness to affine transformations. Section III presents principles of deletions correcting three-dimensional (3D) watermark and authentication codes. Also, in this section we talk about Runlenght coding and the basic of error-correction coding. Finally, in fourth section we show numerical results of vertex deletion probability and error probability performance.

2. The Shape Driven Adaptive Quantization

Firstly, we briefly discuss the embedding and detection in QIM [4]. Then we introduce quantization of polar and spherical coordinates to adapt quantization to main 3D shape features. In same time this adaptation ensures the embedded data robustness to affine transformations.

Quantization Index Modulation (QIM)

Embedding procedure combines the n-dimensional vectors **u** and **x** and produces the watermarked sequence $\mathbf{y} \in \mathbf{R}^n$, where $\mathbf{u} \in \{0,1\}^n$ and $\mathbf{x} \in \mathbf{R}^n$ are the watermark sequence and the cover sequence, respectively. The difference **w=y-x** is the *watermarking displacement* signal and the *embedder* must keep the distortion $d(\mathbf{x},\mathbf{y})$ within a prescribed limit, i.e., $d(\mathbf{x},\mathbf{y}) \le nD$, where D is the maximum allowed distortion per dimension for every **x** and **u**. The distortion is typically defined as the simple Euclidian distance:

$$d_E(\mathbf{x},\mathbf{y}) = \|\mathbf{y}-\mathbf{x}\|^2 \text{, if } \mathcal{X} = \mathcal{Y} = \mathbb{R} \tag{1}$$

In some cases the Hamming distance d_H or Housdorff distance $d_{Housdorff}$ are more appropriate for computations than the expressed Eucledian distance.

$$d_H(\mathbf{x},\mathbf{y}) = \left|\{n : x_n \neq y_n\}\right|, \text{ if } \mathcal{X} = \mathcal{Y} = \{0,1\} \tag{2}$$

$$d_{Housdorff}(\mathbf{x},\mathbf{y}) = \max\left\{\sup_{x\in\mathbf{x}}\inf_{y\in\mathbf{y}} d(x,y), \sup_{y\in\mathbf{y}}\inf_{x\in\mathbf{x}} d(x,y)\right\} \tag{3}$$

The QIM operates independently on the elements u and x of the vectors $\mathbf{u}$ and $\mathbf{x}$. To embed the bit $u \in \{0,1\}$, the QIM requires two uniform quantizers $\boldsymbol{Q}_0$ and $\boldsymbol{Q}_1$ defined as the mappings:

$$\boldsymbol{Q}_u(x) = \Delta\left[\frac{1}{\Delta}\left(x - (-1)^u \frac{\Delta}{4}\right)\right] + (-1)^u \frac{\Delta}{4} \tag{4}$$

where, [ ] denotes the rounding operation. Thus, the quantization level of the "nominal" quantizer $\Delta\,[x/\Delta]$ is moved up or down by $\Delta/4$ depending on the value of u. Equivalently, the watermark bit u dithers the input x by the amount $\pm\Delta/4$. The watermark bit u determines the selection of a quantizer, so that $y = \boldsymbol{Q}_u(x)$. The minimum error produced by QIM is $\Delta/2$. Assuming uniform distribution of the quantization errors over the interval $[-\Delta/2, \Delta/2]$, the mean square error distortion is $\Delta^2/12$.

Ensuring robustness to affine transformations

Ignoring the 3D-object content in the QIM may leads to serious degradation because small Euclidean error may be perceived as a large distortion. Thus, we choose the vector $\mathbf{x}$ so that the application of QIM does not change visual quality [1]. To increase the watermark robustness to affine transformation, spherical coordinates are used. If V is a set of vertices in the 3D Euclidean space and an oriented edge from point u to point v is defined as the ordered pair (u,v), then F is a set of faces and 3D mesh is defined as a pair (V, F). The mass center of the vertices in the set V is calculated as:

$$\mathbf{k}^c = \frac{1}{|V|}\sum_{v\in V}\mathbf{u}_v^C \tag{5}$$

where $\mathbf{u}_v^C = (x_v, y_v, z_v)$ is the position of the vertex $\mathbf{v}$ given in Cartesian coordinates, and $|V|$ denotes the size of the set V. Following [5] and [6], to ensure invariance to translation and rotation, the coordinate system is changed by translating the coordinate origin to the mass center $\mathbf{k}^c$, and by aligning the principal component vector with the z-axis. After the translation, the vertex positions in the new Cartesian coordinate system are $\mathbf{v}_v^C = \mathbf{u}_v^C - \mathbf{k}^c$, where $\mathbf{v}_v^C = (x_v^{'}, y_v^{'}, z_v^{'})$.

The 3D mesh is then rotated to align the principal component vector of the vertices with the z-axis, thus achieving robustness against rotation. Eigenvector that corresponds to the largest eigenvalue of the vertex coordinate covariance matrix is the principal component axis. Converting the Cartesian coordinates of the point $\mathbf{v}$ to spherical coordinates we achieve robustness to uniform scaling. Spherical coordinates are expressed as:

$$r_v = \sqrt{x_v + y_v + z_v}, \theta_v = \arccos\left(\frac{z_v}{r_v}\right)$$
$$\varphi_v = \arctan\left(\frac{y_v}{x_v}\right) \tag{6}$$

wherein $r_v \in [0,\infty)$, $\theta_v \in [0,\pi]$ and $\varphi_v \in [0,2\pi)$, are the radial distance, inclination angle, and the azimuth angle, respectively of the point **v**. Thus the vertex position in spherical coordinates is $\mathbf{v}_v^S = (r_v, \theta_v, \varphi_v)$. The original variant of the QIM operates only on the radial distance from the center of mass, i.e., moves a point **v** with the coordinates $(r_v, \theta_v, \varphi_v)$ to a point $(\boldsymbol{Q}_{u_v}(r_v), \theta_v, \varphi_v)$, where u_v is the bit hidden in the point v. The new center of the mass $\mathbf{k}^{c'}$ of the watermarked object has the following Cartesian coordinates:

$$x^{c'} = y^{c'} = \sum_{1\le i\le n} \boldsymbol{Q}_{u_i}(r_i) \sin\varphi_v \sin\theta_v$$
$$z^{c'} = \sum_{1\le i\le n} \boldsymbol{Q}_{u_i}(r_i) \cos\varphi_v \tag{7}$$

If a hidden binary sequence contains equal number of zeros and ones, and if the vertex positions are uncorrelated with the quantization levels, then for a sufficiently large object $(x^{c'}, y^{c'}, z^{c'})$ converges to a zero vector**.** The above conditions are satisfied in practice, and the watermarked object is therefore invariant to affine transformations

3. Deletion Correcting Authentication Codes

After an optimizing probability of zeros and ones in Distribution transformer, watermark signal is encoded by binary error-correction code, and converted to *runs*. The next block of our system is *Synchronization Error Channel*, whereby watermarking process is completed and signal is prepared for transmission through the channel. We have already mentioned that channels with synchronization errors have infinite memory. Therefore, we use a coding scheme which enables to transform certain channels with synchronization errors into memoryless channels.

Runlength coding

Transformation of channel with infinite memory to a memoryless channel, represents the bits at its input by runs of bits. The runlenghts are selected according to channel synchronization statistics and represent zeros and ones. We explain the main idea considering a case in which binary zeros are represented by runs of length *two*, and binary ones with runs of length *three* [7]. If the sequence of information bits is $b = (1,1,0,0,1,0)$, then the runlength encoded sequence c is initialized $c = (111\ 000\ 11\ 00\ 111\ 00)$.

In our case the runlength sequence is embedded into selected vertices of a 3D object by using sparse-QIM: first, the watermark binary sequence is processed in the distribution transformer to optimize probability of zeros and ones, and encoded by a binary LDPC code. Then, conversion to runs and transmission through the channel are realized.

Symbols have different lengths, thus optimizing probability of symbols is necessary. The maximizing the mutual information, $I(\mathbf{X};\mathbf{Y})$, between the input alphabet **X** and the output alphabet **Y**, over all input distributions of **X**. determines the Shannon capacity of memoryless channels, which is in discrete case obtained by maximizing the mutual information, $I(\mathbf{p})$, over all input probability distribution vectors:

$$\mathbf{p} = (p(x))_{x \in X} \tag{8}$$

If the costs of transmission of different symbols are not equal, for the channel where symbol lengths are not equal, $c(x)$ as length of symbol x, can be viewed as the cost of transmission of the symbol x. If we define **c**, the transmission cost vector, $\mathbf{c} = (c(x))_{x \in X}$. In such channels, we use the notion of *unit-cost capacity* [19] defined as:

$$C_{unit} = \frac{\max_{\mathbf{p}} I(\mathbf{p})}{\mathbf{c}\mathbf{p}^T} \tag{9}$$

The capacity is a function of the synchronization error probability as well as the size of the input alphabet **X**. A spectrum of transmission schemes can be obtained by changing the alphabet-size at the input. In general, for a channel C, with information alphabet **X**, we can calculate an associated unit-cost capacity $C(C;|\mathbf{X}|)$, and determine optimal capacity-achieving input probabilities. Subsequently, the data is encoded by using an error-correcting code and then, as described previously, the information symbols are encoded in runs of channel bits.

Error-correction encoding

In the context of 3D watermarking, the channel introduces only deletions ($p_i = 0$). We also assume that no two consecutive vertices are deleted in the process of simplification. This assumption is justified by the fact that the Ordered Statistics Vertex Extraction and Tracing Algorithm (OSVETA) [8] selects *stable* vertices, and two consecutive vertices are deleted extremely rarely. In other words, the space between two consecutive deleted vertices is sufficiently large, i.e. there are no *bursts* of synchronization errors. Separation of synchronization errors within a codeword may be also achieved by interleaving, i.e. by proper indexing of vertices. Consequently, we consider a variant of the channel described above in which the number of consecutive synchronization errors is restricted and operates on individual runs of binary sequences. Here, we define the term *run* in binary sequences as an occurrence of k consecutive, identical symbols.

Let C be an (n, k) LDPC code over the binary field GF(2). C is defined by the null space of H, an $m \times n$ parity-check matrix of C. H is the bi-adjacency matrix of G, which is a Tanner graph representation of C. G is a bipartite graph with two sets of nodes: n variable (bit) nodes $V = \{1, 2, \dots, n\}$ and m check nodes $C = \{1, 2, \dots, m\}$. A vector $\mathbf{x} = (x_1, x_2, \dots, x_n)$ is a codeword if and only if $\mathbf{x}H^T = 0$, where H^T is the transpose of H. In this paper we consider $(d_v; d_c)$-regular LDPC, i.e., codes with Tanner graphs G in which all variable nodes have degree d_v, and all check nodes have degree d_c. Such codes have rate $R \geq 1 - d_v/d_c$ [20]. We consider a channel with binary input and output alphabets. A bit may be inserted with a probability p_i, deleted with a probability p_d, or correctly transmitted with a probability $1 - p_i - p_d$.

Next, we denote a channel $\mathcal{C}$ with the parameter s_d, which is the maximum number of consecutive deletions as (0, s_d). It is a special case of the channel we introduced in [7]. The probability of j consecutive deletions is $(p_d)^j$ for $j = 1, \dots , s_d$. In each run of zeros or ones, only one of the $s_d + 1$ error events occurs, namely, (*i*) error-free transmission, or (*ii*) deletion of s_d bits, $j = 1, \dots , s_d$. We do not consider errors in bit-values introduced by channel since we consider only common transformations. In general, these parameters may be chosen to match the behavior of the object in which we hide the watermark. Consider the channel $\mathcal{C}$ with parameters (p_d, s), having input **x**. As we have shown in [7] if all the runs in **x** have lengths greater than s, then the number of runs in any output **y** produced by $\mathcal{C}$ is equal to the number of runs in **x**. Summarizing, when transmitted through $\mathcal{C}$, each run of s_t bits, $s_t > s$, results in a run of j bits, $j = s_t - s, \dots, s_t$. This observation leads naturally to an encoding scheme where symbols are encoded in runs of bits.

The encoding scheme described above results in a discrete memoryless channel with the set of input symbols l_0 and $l_0 + 1$, the set of outputs of the channel correspond to the all the possible values of runlengths at the receiver, i.e. $\{l_0 - 1, l_0$ and $l_0 + 1\}$ and transitions that are possible only if output runlength is equal to the input or shorter by one. It is easy to see that this methodology is not restricted to encoding of binary-valued sequences, and can be easily extended to larger alphabets. For example, an input sequence is grouped into pairs of symbols (00, 01, 10 and 11), and then each pair is represented by a unique runlength (00 by runlength 2, 01 by runlengt 3 etc.). For example, an input sequence 0101001011 is parsed as 01, 01, 00, 10, 11 and then runlength encoded to obtain the sequence (000, 111, 00, 1111, 00000).

4. EXPERIMENTAL RESULTS

For all further computations we use results of vertex stability in relation with optimization process obtained by OSVETA [8][9] for three high-density 3D mesh models, which are obtained from 3D scanning process. Scanned models are already optimized in order to remove topological errors produced by the scanner hardware. Textured 3D models with appropriate solid models are denoted as A and B and shown on the following figure (topological errors are ploted as red dots).

Fig. 1 3D models of national costumes (A and B) used in computations

For comparison and evaluation we have used 'Pro Optimizer' modifier from 3D Studio Max 2015 application [10] that is based on simplification algorithms [11], [12]. Experimental tests of stability 1000 vertices, selected by OSVETA algorithm compared to the randomly allocated group of 1000 vertices showed the superiority of our approach compared to random selection of vertices. The results are summarized in Table 1.

Table 1 The number of vertices deleted by optimization

	0%	20%	40%	60%	80%	90%
#A Total VR	73901	59120	44360	29560	14780	7390
#A Random	0	184	389	591	789	913
#A OSVETA	0	120	264	456	709	845
#B Total VR	95858	76686	57514	38343	19171	9585
#B Random	0	182	394	591	799	895
#B OSVETA	0	73	227	363	597	774

The first row gives the Total number of Vertices Remaining after simplification (Total RV) with a given percent of remained vertices. The second and third rows give the number of removed vertices out of 1000 vertices selected randomly and selected by the OSVETA. The nonzero numbers in the 0% column correspond to the total number of vertices in the original object. Probability of vertex deletion as a function of the percent of deleted vertices is shown in next figure (Fig. 2).

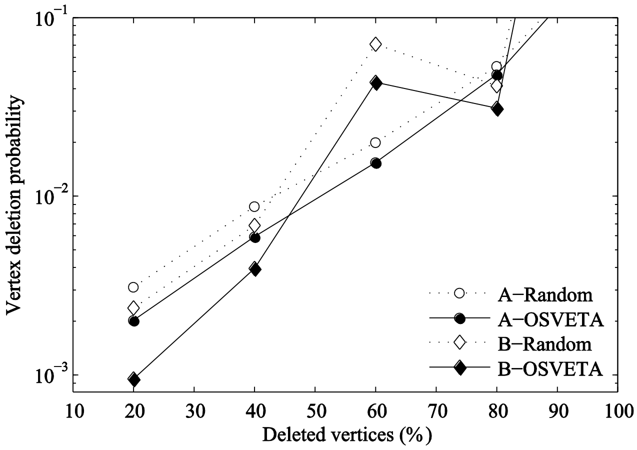

Fig. 2 Probability of vertex deletion p_d as a function of the face.

The important fact is that the OSVETA provides an extremely low probability of deleting two consecutive selected vertices. Actually, in 1000 selected vertices of all optimization levels we did not find any deleted pair of consecutive vertices. This justifying to the (p,1) deletion channel assumption.

In order to demonstrate the feasibility of implementation of our encoding methodology, we conducted experiments to simulate transmission of coded information through the channel (p, 1). For our simulations, we chose synchronization-error probability (p) ranging from 0:01 to 0:05. Two codes were chosen as candidates for simulation. The codes were constructed by methods described in [13], and provide guarantees on error-correction capability under iterative error-correction decoding for BSC. The code parameters (code length, n, and the number of message bits, k) are as follows: $n = 2212$, $k = 1899$, $R_{eff} = 0.34$ for Code 1, and $n = 848$, $k = 661$, $R_{eff} = 0.32$ for Code 2, where R_{eff} denotes the effective rate, i.e. the number of information bits transmitted per channel use. For simplicity, in the simulations the equiprobable inputs are used.

Since the average cost of a bit transmission is 2.5 (symbols of duration 2 and 3 are used half of the time in average), there runlength coding introduces a rate loss of 2/5. The Effective rate is thus $R_{eff} = (2/5)R$ where R is the rate of the LDPC code.

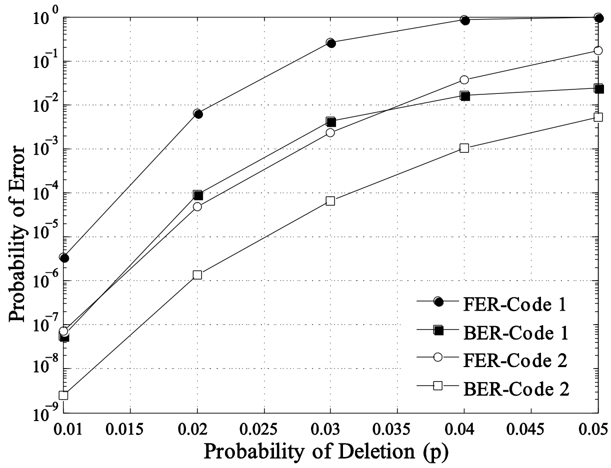

Fig. 3 Codeword probability of error as a function of deletion probability

On figure Fig. 3 [1] the bit-error rates (BER) and frame-error (FER) rates are shown with respect to the vertex deletion probability parameter p. Sum-product algorithm was used for decoding the codes. For example to ensure that at most one in a million vertices is unrecoverable ($BER \leq 10^{-6}$) by an LDPC code (Code 1), the OSVETA has to provide a raw deletion probability of no more than p=0.02. From Fig. 2 we see that this is readily achieved when face threshold is less than 6. The proposed scheme handles even higher FT if the maximum BER requirement is relaxed.

We note that the input distribution is not optimized, i.e. the inputs were uniformly distributed. We also note that the use of distribution transformers introduce additional loss in the effective rate. However, a detailed discussion of this is beyond the scope of this analysis. We also note that although effective rates of codes for binary alphabet are low, by using codes over higher alphabets superior guarantees on maximum achievable rates may be obtained. This problem is left for future research.

6. Conclusion

A combination of QIM and error correction coding is an essence of our 3D mesh authentication method, which we have presented in this paper. Stabile vertices are selected by OSVETA algorithm, whereupon QIM is performed on spherical coordinates of these vertices. Runlength-encoded watermark bits are subjected to error correction code that ensures recovery of bits which are deleted by simplification. The runlength coding also makes the watermarking process equivalent to a memoryless channel and its conceptual simplicity allows using powerful codes that has developed for these channels.

Carefully designed error-correction codes give the strength of the proposed watermarking coding scheme. The sophisticated OSVETA algorithm provides a low probability of selected vertices deletion, whereas iterative decoding algorithm represents a computational simplicity in recovering of vertices, which are deleted by simplification process.

References

[1] B. Vasic, B. Vasic, "Simplification resilient LDPC-coded sparse-QIM watermarking for 3D-meshes," IEEE Transaction on Multimedia, vol. 15, no. 7, pp. 1532–1542, 2013. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6522818

[2] D. Coumou, G. Sharma, "Insertion, deletion codes with feature-based embedding: a new paradigm for watermark synchronization with applications to speech watermarking," IEEE Transactions on Information Forensics and Security, vol. 3, no. 2, pp. 153–165, 2008. [Online]. Available: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4488221&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D4488221

[3] M. C. Davey, D.J.C. Mackay, "Reliable communication over channels with insertions, deletions, and substitutions," IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 687–698, 2001. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.8974

[4] B. Chen, G. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," IEEE Transactions on Information Theory, vol. 47, no. 4, pp. 1423–1443, 2001. [Online]. Available: http://www.fim.uni-linz.ac.at/lva/Rechtliche_Aspekte-/2001SS/Stegano/leseecke/quantization%20index%20modulation.pdf

[5] A. Kalivas, A. Tefas, I. Pitas, "Watermarking of 3D Models Using Principal Component Analysis," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 676–679, 2003. [Online]. Available: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1200061&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F8535%2F26996%2F01200061.pdf%3Farnumber%3D1200061

[6] B. Vasic, "Increasing 3D mesh watermark robustness to affine transformations using barycenter as spherical coordinate origin," in *Proceedings 56th Conference for Electronics, Telecommunications, Computers, Automation, And Nuclear Engineering - ETRAN 2012*, Serbia, Zlatibor, 2012. [Online]. Available: http://etran.etf.rs/etran2012/Program_ETRAN_2012.pdf

[7] R. Krishnan, B. Vasic, "Coding for correcting insertions and deletions in bit-patterned media recording," in *Proceedings IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp 1–5, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6134512

[8] B. Vasic, "Ordered statistics vertex extraction and tracing algorithm (OSVETA)," Advances in Electrical and Computer Engineering, vol. 12, no 4, pp. 25–32, 2012. [Online]. Available: http://www.aece.ro/archive/2012/4/2012_4_4.pdf

[9] B. Vasic, "Ordered statistic vertex extraction and tracing algorithm (OSVETA)," MatLab Software. [Online], Available: http://www.batavasic.com/research/OSVETA.zip

[10] M. Kauffman, "Optimizing Your Autodesk® 3ds Max® Design Models for Project Newport", Autodesk University 2009, [Online] Available: http://au.autodesk.com/?nd=material&session_material_id=6296

[11] P. Heckbert, M. Garland, "Survey of polygonal surface simplification algorithms," Siggraph 97 Course Notes, no. 25, ACM Press, New York, 1997. [Online]. Available: http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA461098

[12] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, W. Stuetzle, "Mesh Optimization," in *Proceedings of SIGGRAPH 93*, Anaheim, USA, pp. 19–26, 1993. [Online]. Available: https://dl.acm.org/purchase.cfm?id=166119&CFID=416637604&CFTOKEN=86550331
[13] B. Vasic, O. Milenkovic, S. McLaughlin, "Scrambling for nonequiprobable signalling," IEEE Electronics Letters, vol. 32, no. 17, pp. 1551–1552, 1996. [Online]. Available: http://dx.doi.org/10.1049/el:19961021