3D MESH STEGANALYSIS USING LOCAL SHAPE FEATURES

Zhenyu Li [*] *and Adrian G. Bors*

Department of Computer Science, University of York, York YO10 5GH, UK
Email:{zl991, adrian.bors}@york.ac.uk

ABSTRACT

Steganalysis aims to identify those changes performed in a specific media with the intention to hide information. In this paper we assess the efficiency, in finding hidden information, of several local feature detectors. In the proposed 3D steganalysis approach we first smooth the cover object and its corresponding stego-object obtained after embedding a given message. We use various operators in order to extract local features from both the cover and stego-objects, and their smoothed versions. Machine learning algorithms are then used for learning to discriminate between those 3D objects which are used as carriers of hidden information and those are not used. The proposed 3D steganalysis methodology is shown to provide superior performance to other approaches in a well known database of 3D objects.

Index Terms— Steganalysis, 3D features, Fisher Linear Discriminant

1. INTRODUCTION

Steganalysis is a method used to identify whether a certain media was modified with the aim to hide information. The media considered in this study is 3D graphics, and the information can be hidden by watermarking, which produces robust changes or by steganography, which embeds a larger payload. Steganalysis was considered on digital images [1, 2, 3, 4], audio signals [5] and video [6].

A 3D mesh is a collection of vertices, edges and faces that defines the shape of a polyhedral object, which can be used in many applications including 3D signal processing and computer graphics. As the popularity of the 3D printing is rocketing, the 3D meshes are playing an increasingly important role. Ohbuchi et al. [7] proposed two 3D information hiding algorithms using triangle similarity quadruple and tetrahedral volume ratio. Cayre and Macq [8] presented a steganography approach for 3D triangle meshes whose key idea is to consider a triangle as a two-state geometrical object. Cho et al. [9] proposed two blind robust watermarking algorithms based on modifying the mean or variance of the distribution of vertex norms. Luo and Bors [10] proposed two surface-preserving 3D watermarking algorithms by changing the mean or variance of of geodesic distance distributions, while in [11] they proposed a watermarking method which optimally displaces vertices while minimizing surface distortion. Among the information hiding methods embedding higher payload, we mention a multi-layer 3D steganography method [12] based on vertex projection onto the principal axis. This steganography method can reach a high payload given by the number of embedding layers (decided by the user) multiplied with the number of vertices in the mesh. However, some of the bits embedded by this method may be lost in the retrieval stage.

The 3D steganalysis algorithm proposed in [13] uses a feature set composed from the vertex position, the norms in Cartesian and Laplacian coordinate system [14], the dihedral angle of edges and the face normals, while the quadratic classifier is used for identifying those 3D shapes that contain hidden information. More recently, Yang et al. proposed a steganalytic algorithm in [15], specifically designed for the mean-based watermarking algorithm proposed in [9], which estimates the number of bins through exhaustive search and then detects the presence of the secret message by a tailor-made normality test. The steganalysis method proposed in this paper relies on extracting a novel feature set from the 3D mesh of the graphical object. We propose to use Fisher Linear Discriminate (FLD) ensemble [1] for 3D steganalysis. Before presenting the feature set, a brief introduction of the 3D steganalysis framework is given in Section 2. Then, the local feature set that consists of a simplified variation of the features proposed in [13], is used in the conjunction with the proposed 3D features, consisting of statistics of vertex normals and of local shape curvature, are presented in detail in Section 3. The experimental results are provided in Section 4, while the conclusions of this study are outlined in Section 5.

2. 3D STEGANALYSIS FRAMEWORK

Initially, the 3D mesh is aligned with the coordinate axes and then is scaled to the size of a unit cube centered at (0.5, 0.5, 0.5). The calibration step consists of aligning the reference mesh representing the stego-object $\hat{\mathbf{M}}$ with the cover-object $\mathbf{M}$. In our analysis it is expected that the difference between a mesh and its reference is larger for a stego mesh than for a cover mesh. In most 3D information hiding algorithms, the changes produced to the stego object, following the em-

[*] The first author acknowledges the scholarship received from Zhengzhou Institute of Information Science and Technology.

bedding of information into its surface, can be associated to noise variations. So the cover meshes will have fewer alterations in the structure of their surface when smoothed than the stego meshes. For a certain 3D object the differences between its cover mesh and its smoothed version will be smaller that the differences between the stego mesh and its corresponding smoothed version. In our approach, the reference mesh is obtained by applying one iteration of Laplacian smoothing.

In order to identify changes produced by hiding information into 3D objects, we propose using statistics of results provided by local feature detectors, extracted from pairs of cover and stego-objects, before and after Laplacian smoothing. The discriminative features should capture those differences between the two meshes that have most likely been caused by an information hiding algorithm. In the next section we introduce a set of new feature detectors using the vertex normals and local shape curvature estimators. We also consider using statistics of various combinations of outputs provided by multiple 3D feature detectors and compare their results with single feature detectors.

In the training step, the steganalyzer is implemented through a machine learning method using the feature vectors extracted from both the cover and stego meshes as the input. The training set is formed by pairs of cover features and the corresponding stego features, extracted as differences with respect to their smoothed versions. This is quite important as it has been shown that breaking the cover-stego pairs may lead to a suboptimal performance [16]. Furthermore, training the classifier is a crucial issue, as the steganalyzers trained by different machine learning methods will show different performances. In the experimental results from Section 4, we propose to use the FLD ensemble analysis [1], as a method for identifying the payload carrying 3D shapes. In the detection procedure, the steganalyzer will determine whether a certain mesh was embedded with a payload or not, according to the parameters learnt during the training stage and to the statistical properties of the features extracted from the mesh.

3. LOCAL FEATURE SET

In the following we describe some local features detectors used for identifying the changes in the 3D meshes produced by information hiding algorithms. They are applied to both cover and stego-objects.

3.1. Yang's Features

The 40-dimensional feature YANG40 is a simplified variation of the 208-dimensional feature YANG208 [13], which uses the same principle of detecting features from the basic graphical object structure, representing its vertices, edges and polygons. With regard to the mesh vertices, the absolute differences between the locations of vertices of the cover mesh **M** and its smoothed version **M**' on x, y and z axes in both the Cartesian and Laplacian coordinate systems[14] are computed, $\mathbf{f}_1$, $\mathbf{f}_2$ and $\mathbf{f}_3$ for Cartesian coordinate systems, while, $\mathbf{f}_4$, $\mathbf{f}_5$ and $\mathbf{f}_6$ for the Laplacian coordinate systems. Then we consider the vertex norms, representing the distance from vertex locations to the center of the object. Moreover, the absolute differences between the vertex norms of each two corresponding vertices in the meshes **M** and **M'**, representing features $\mathbf{f}_7$ and $\mathbf{f}_8$ for the Cartesian and Laplacian coordinate systems, respectively, are considered.

With respect to the edges and faces of the mesh, the absolute differences between the dihedral angles of **M** and **M'**, are considered:

$$\mathbf{f}_9(i) = |\mathrm{dihedral}(e(i)) - \mathrm{dihedral}(e'(i))|, \qquad (1)$$

where $i = 1, 2, \ldots, |E|$, where $|E|$ represents the number of edges from the object **M**, and $\mathrm{dihedral}(\cdot)$ is the function computing the dihedral angle for an edge e of the cover mesh and for its corresponding smoothed version e'. The absolute value of the angles between the face normals of **M** and **M'** are computed:

$$\mathbf{f}_{10}(i) = \arccos \frac{\vec{\mathbf{N}}_{f(i)} \cdot \vec{\mathbf{N}}_{f'(i)}}{\left\| \vec{\mathbf{N}}_{f(i)} \right\| \cdot \left\| \vec{\mathbf{N}}_{f'(i)} \right\|} \qquad (2)$$

where $i = 1, 2, \ldots, |F|$ where $|F|$ represents the number of faces in the object **M**.

Finally, the 40-dimensional feature vector of YANG40 consists of the first four statistical moments of their overall distribution of their logarithm, representing their mean, variance, skewness and kurtosis. The log transformation is used in order to reduce the range of available feature values identified in the two objects, and to assign higher weights for smaller positive values for balancing their contribution, [13]. Compared to the 208-dimensional feature YANG208, YANG40 doesn't compute the first 8 vectors $\{\mathbf{f}_i | i = 1, 2, \ldots, 8\}$ on vertices with valence less, equal, or greater than six separately which would increase the dimensionality of the feature but according to our tests it does not have any obvious improvement on the total performance of the 3D streganalyzer.

3.2. Vertex Normal Features

A vertex normal is the weighted sum of the normals of the faces that contain that vertex [17]. It can be computed using the formula below

$$\vec{\mathbf{N}}_{v(i)} = \sum_{f(j) \in F_{v(i)}} \frac{\mathrm{Area}(f(j)) \cdot \vec{\mathbf{N}}_{f(j)}}{\left\| e_{(v(i), v'_{f(j)})} \right\|^2 \cdot \left\| e_{(v(i), v''_{f(j)})} \right\|^2} \qquad (3)$$

where $F_{v(i)}$ is the set of faces that contain the vertex $v(i)$, $v''_{f(j)}$ and $v'_{f(j)}$ are the two vertices adjacent to vertex $v(i)$ in the face $f(j)$, $e_{(v_1, v_2)}$ represents the edge connecting vertices v_1 and v_2. $\mathrm{Area}(f(j))$ represents the area of the face $f(j)$.

For the 11th feature, we calculate the angle between the vertex normals of each two corresponding vertices $v(i)$ and $v'(i)$ from meshes **M** and **M'**, namely

$$\mathbf{f}_{11} = \arccos \frac{\vec{\mathbf{N}}_{v(i)} \cdot \vec{\mathbf{N}}_{v'(i)}}{\left\| \vec{\mathbf{N}}_{v(i)} \right\| \cdot \left\| \vec{\mathbf{N}}_{v'(i)} \right\|}, \qquad (4)$$

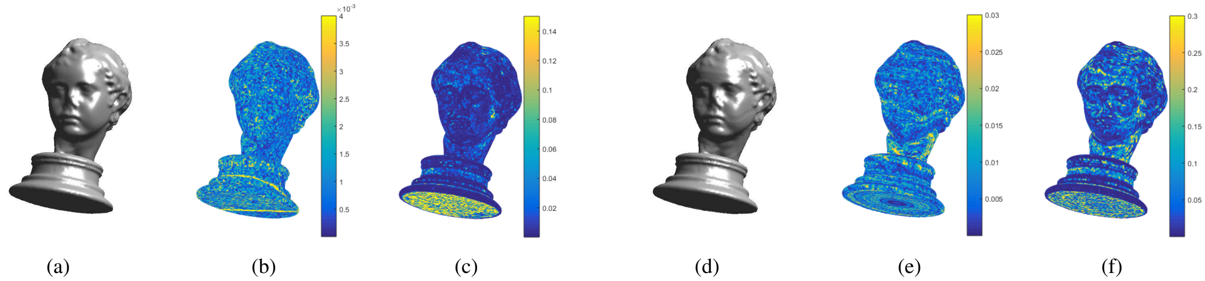


Fig. 1: 3D feature detection on stego-objects. (a) and (d) are the stego-objects obtained after hiding information by the steganography method from [12] and the watermarking from [9], respectively; (b) and (e) show the differences of vertex normals $\mathbf{f}_{11}$ between (a), (d) and their corresponding cover-object, respectively; (c) and (f) show the differences of the curvature ratios $\mathbf{f}_{13}$ between (a), (d) and their corresponding cover-object, respectively.

where $i = 1, 2, \ldots, |V|$, where $|V|$ represents the total number of vertices from the mesh $\mathbf{M}$.

3.3. Curvature Features

The surface curvature was used for characterizing 3D shapes for recognition and object retrieval. In this research study we propose to use the surface curvature for 3D steganalysis. The Gaussian curvature and curvature ratio are known to model well surface variation [18]. In differential geometry, the two principal curvatures at a given point of a surface measure how the surface bends by different amounts in different directions and is given by the eigenvalues of the shape operator at that point. We use the method from [19] in order to obtain the two principal curvatures at the location of each vertex. The Gaussian curvature is given by:

$$K(i) = \kappa_1(i)\kappa_2(i) \tag{5}$$

where $\kappa_1(i)$ is the minimum curvature and $\kappa_2(i)$ is the maximum curvature at a given location on the surface and $i = 1, 2, \ldots, |V|$. In our study we found the curvature ratio proposed in [18], defined as

$$\kappa_3(i) = \frac{\min(|\kappa_1(i)|, |\kappa_2(i)|)}{\max(|\kappa_1(i)|, |\kappa_2(i)|)}, \tag{6}$$

is effective for 3D steganalysis as well. These two properties, the Gaussian curvature and the curvature ratio, can describe locally well the shape of 3D meshes while being sensitive to any small changes.

We compute the absolute difference of the Gaussian curvatures and of the curvature ratios at the location of each pair of vertices $\{(v(i), v'(i) | v(i) \in \mathbf{M}, v'(i) \in \mathbf{M}', i = 1, 2, \ldots, |V|\}$ in the given objects:

$$\mathbf{f}_{12} = |K(v(i)) - K(v'(i))|, \tag{7}$$

$$\mathbf{f}_{13} = |\kappa_3(v(i)) - \kappa_3(v'(i))|. \tag{8}$$

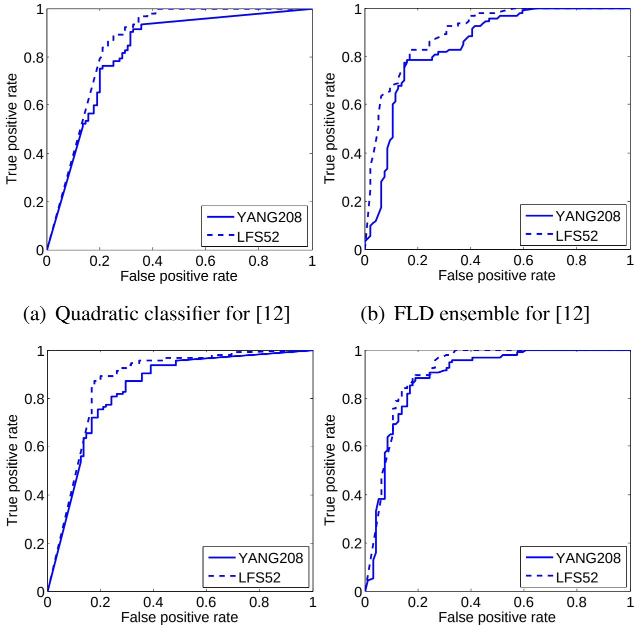
Finally, the four statistical moments, representing the mean, variance, skewness and kurtosis, are calculated for the logarithm of the vectors $\mathbf{f}_{12}$ and $\mathbf{f}_{13}$.

We now have a local feature set of 52 dimensions, called LFS52, which consists of three components: the 40-dimensional feature YANG40, the 4-dimensional vertex normal feature VNF4 and the 8-dimensional curvature feature vector CF8. Combinations of various 3D features are considered during the experiments provided in the next section.

4. EXPERIMENTAL RESULTS

In the following we provide the results where we apply the feature operators described in the previous section for 3D steganalysis. We consider the Princeton Mesh Segmentation project [20] database, which consists of 354 meshes of 3D objects. We consider two different steganography approaches for hiding information in the 3D objects. The first method is a steganography method proposed in [12], while the second is the mean-based watermarking method proposed in [9]. For the former method we set the number of layers at 10, which corresponds to a high capacity while the number of intervals is set as 10000. During the embedding, all the vertices in the mesh are carrying payloads, except for three vertices which are considered as the bases for the extraction process. Meanwhile, for the watermarking method proposed in [9], we consider the incremental step size as $\Delta k = 0.001$, the strength factor $\alpha = 0.04$ and the payload message as 64 bits. We consider 260 objects for training, while the other 94 are used for testing. An example of the stego-object "Head Statue" is shown in Fig. 1a by [12], while in Fig. 1d we show the same object watermarked by [9]. For the statistical significance of the results and for removing the chance factor, we consider 30 different splits of the given 3D object database, into training and testing data sets. The final results are indicated by the median of the average of probability of false negatives (missed detections) and false positives (false alarms) from all 30 trials.

The 3D features described in Section 3 are extracted from each of the cover-objects from the database, their corresponding stego-object as well as from their smoothed versions. The differences in the stego and cover-objects identified by the differences in vertex normals are shown in Fig. 1b and e, while those identified by the curvature ratios are shown in Fig. 1c and f, when detecting the messages embedded by [12] and [9], respectively. In the training stage we propose to use the FLD



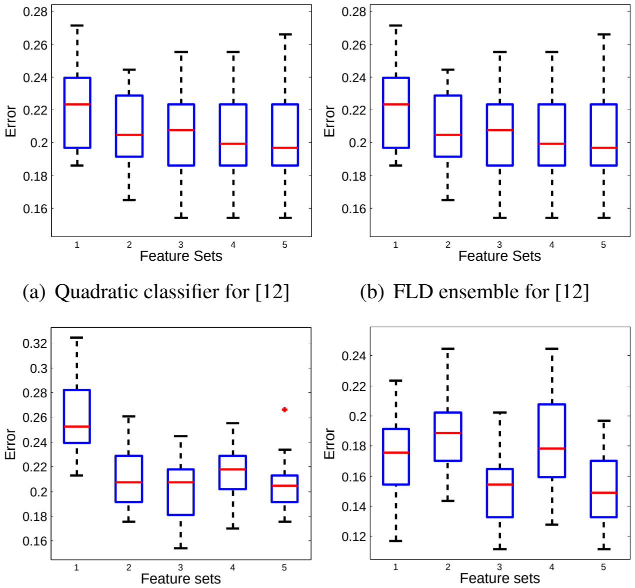
(a) Quadratic classifier for [12]

(b) FLD ensemble for [12]

(c) Quadratic classifier for [9]

(d) FLD ensemble for [9]

Fig. 2: ROC curves of the steganalyzers.



(a) Quadratic classifier for [12]

(b) FLD ensemble for [12]

(c) Quadratic classifier for [9]

(d) FLD ensemble for [9]

Fig. 3: Steganalysis detection errors for two information hiding algorithms when using quadratic classifiers and FLD ensembles as steganalyzers. Labels 1 to 5 represent the results for the feature sets YANG208, YANG40, YANG40+VNF4, YANG40+CF8 and LFS52.

ensemble [1] analyzer for identifying the 3D stego-objects, and we compare its results with those of the quadratic discriminant. The quadratic discriminant fits multivariate normal distribution to the given statistics [21] extracted from the cover-objects and was used for 3D steganalysis in [13]. FLD ensemble is the most used classifier in image steganalysis because of its powerful ability to find non-linear separation boundaries in a reasonably short time [1]. The random subspace dimensionality and the number of base learners for the FLD ensemble method are found by minimizing the out-of-bag (OOB) estimate of the testing error. The FLD ensemble method used in this research study represents a version adapted from that proposed in [22].

Receiver Operating Curves (ROC) curves for the steganalysis results when using quadratic classifiers in Fig. 3a and c, while the results for the FLD ensembles are shown in Fig. 3b and d, when considering the objects embedded by the methods from [12] and [9], respectively. The results provided the proposed LFS52 feature sets are compared to those of the YANG208 feature sets used in [13]. From these ROC curves it is clear that the area under the ROC curves for steganalyzers using LFS52 are larger than those when using the feature set YANG208 in the case of both information hiding algorithms.

We compare the proposed local feature set LFS52 with the features set YANG208, its simplified variation (YANG40), the combination of YANG40 and vertex normal feature VNF4, and the combination of YANG40 and curvature feature CF8. Fig. 3 shows the distributions of detection errors for the steganalyzers trained as quadratic classifiers and FLD ensembles using the five feature combinations mentioned above, when detecting the messages embedded by either [12] or [9]. It can be observed from Fig.3 that LFS52 has the best performance among all these five combinations of features. Although YANG40 is simplified from YANG208, it provides better results than YANG208. Usually, by combining either VNF4 or CF8 with YANG40 the results are better than by just using the YANG40 feature set, but the improvement of adding CF8 is smaller than that of adding VNF4.

5. CONCLUSION

In this research study we propose to use a new local feature set for 3D mesh steganalysis, which includes the vertex normals and the local curvature. The first four statistical moments, representing the mean, variance, skewness and kurtosis, are used for defining distributions of local features in 3D objects, which aim to identify small changes between pairs of meshes. The paired meshes correspond to cover- and stego-objects and their smoothed counterparts. Machine learning based classifiers are then used to distinguish statistical differences between cover-objects and their smoothed versions from those that occur between stego-objects and their smoothed versions. The quadratic discriminant and the FLD ensembles are used for identifying the 3D objects where information was hidden. The proposed methodology is shown to provide better results than other 3D steganalysis approaches.

6. REFERENCES

[1] Jan Kodovskỳ, Jessica Fridrich, and Vojtěch Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 432–444, 2012.

[2] Zhenyu Li, Zongyun Hu, Xiangyang Luo, and Bin Lu, "Embedding change rate estimation based on ensemble learning," in *Proc. of the first ACM workshop on Information hiding and multimedia security*. ACM, 2013, pp. 77–84.

[3] Rémi Cogranne, Cathel Zitzmann, Lionel Fillatre, Florent Retraint, Igor Nikiforov, and Philippe Cornu, "A cover image model for reliable steganalysis," in *Information Hiding*. Springer, 2011, pp. 178–192.

[4] Gokhan Gul and Fatih Kurugollu, "SVD-based universal spatial domain image steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 349–353, 2010.

[5] Y. Ren, T. Cai, M. Tang, and L. Wang, "Amr steganalysis based on the probability of same pulse position," *IEEE Transactions on Information Forensics and Security*, vol. PP, no. 99, pp. 1–11, 2015.

[6] Keren Wang, Hong Zhao, and Hongxia Wang, "Video steganalysis against motion vector-based steganography by adding or subtracting one motion vector value," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 741–751, 2014.

[7] Ryutarou Ohbuchi, Hiroshi Masuda, and Masaki Aono, "Embedding data in 3D models," in *Interactive Distributed Multimedia Systems and Telecommunication Services*. Springer, 1997, pp. 1–10.

[8] François Cayre and Benoît Macq, "Data hiding on 3-D triangle meshes," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 939–949, 2003.

[9] Jae-Won Cho, Rémy Prost, and Ho-Youl Jung, "An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142–155, 2007.

[10] Ming Luo and Adrian G. Bors, "Surface-preserving robust watermarking of 3-D shapes," *IEEE Transactions on Image Processing*, vol. 20, no. 10, pp. 2813–2826, 2011.

[11] Adrian G. Bors and Ming Luo, "Optimized 3D watermarking for minimal surface distortion," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1822–1835, 2013.

[12] Min-Wen Chao, Chao-hung Lin, Cheng-Wei Yu, and Tong-Yee Lee, "A high capacity 3D steganography algorithm," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 2, pp. 274–284, 2009.

[13] Ying Yang and Ioannis Ivrissimtzis, "Mesh discriminative features for 3D steganalysis," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 10, no. 3, pp. 27:1–27:13, 2014.

[14] Ying Yang and Ioannis Ivrissimtzis, "Polygonal mesh watermarking using laplacian coordinates," in *Computer Graphics Forum*. Wiley Online Library, 2010, vol. 29, pp. 1585–1593.

[15] Ying Yang, Ruggero Pintus, Holly Rushmeier, and Ioannis Ivrissimtzis, "A steganalytic algorithm for 3D polygonal meshes," in *Proc. IEEE Int. Conf. on Image Processing*. IEEE, 2014, pp. 4782–4786.

[16] Valentin Schwamberger and Matthias O Franz, "Simple algorithmic modifications for improving blind steganalysis performance," in *Proc. of the 12th ACM workshop on Multimedia and Security*. ACM, 2010, pp. 225–230.

[17] Nelson Max, "Weights for computing vertex normals from facet normals," *Journal of Graphics Tools*, vol. 4, no. 2, pp. 1–6, 1999.

[18] John Rugis and Reinhard Klette, "A scale invariant surface curvature estimator," in *Advances in Image and Video Technology*, pp. 138–147. Springer, 2006.

[19] Szymon Rusinkiewicz, "Estimating curvatures and their derivatives on triangle meshes," in *Proc. of Int. Symposium 3D Data Processing, Visualization and Transmission*, 2004, pp. 486–493.

[20] Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser, "A benchmark for 3D mesh segmentation," in *ACM Transactions on Graphics*. ACM, 2009, vol. 28, pp. 73:1–73:12.

[21] Wojtek Krzanowski, *Principles of multivariate analysis*, Oxford University Press, 2000.

[22] Remi Cogranne and Jessica Fridrich, "Modeling and extending the ensemble classifier for steganalysis of digital images using hypothesis testing theory," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2627–2642, 2015.