Linear Correlations between Spatial and Normal Noise in Triangle Meshes

Ying Yang, Norbert Peyerimhoff, and Ioannis Ivrissimtzis

Abstract—We study the relationship between the noise in the vertex coordinates of a triangle mesh and normal noise. First, we compute in closed form the expectation for the angle θ between the new and the old normal when uniform noise is added to a single vertex of a triangle. Next, we propose and experimentally validate an approximation and lower and upper bounds for θ when uniform noise is added to all three vertices of the triangle. In all cases, for small amounts of spatial noise that do not severely distort the mesh, there is a linear correlation between θ and simple functions of the heights of the triangles and thus, θ can be computed efficiently. The addition of uniform spatial noise to a mesh can be seen as a dithered quantization of its vertices. We use the obtained linear correlations between spatial and normal noise to compute the level of dithered quantization of the mesh vertices when a tolerance for the average normal distortion is given.

Index Terms—Triangle mesh, normal noise, vertex quantization

1 Introduction

The geometric information of a triangle mesh is encoded on its vertex coordinates. A modification of these coordinates, as result, for example, of vertex quantization, can change the appearance of the rendered mesh in two ways. Either directly, in the form of a spatial perturbation of the vertices, or indirectly, by changing triangle normals used by the rendering algorithm. Generally, the induced normal perturbations are much more distractive to the human eye than the spatial perturbations, see Fig. 1.

That means that although most mesh processing operations manipulate spatial information, we are mainly interested in the indirect effects the spatial manipulation has on the normal information. In this paper, our aim is to develop an understanding of the relationship between spatial and normal perturbations, so that users can modify geometric information knowing the effects of the applied mesh processing operations on the visually important normal information.

1.1 Summary of Results

For small vertex perturbations caused by small amounts of added spatial noise, the first result derived in Section 2 is surprisingly simple: when a small amount of noise is added to a single triangle vertex, the expected change of the normal is an almost linear function of the inverse of the height at that vertex. Based on this result, we derive heuristic bounds and an approximate formula for the expected normal change when a small amount of noise is added to all three vertices of the triangle. In Section 3, we validate the bounds and the approximation by comparing them to the average normal change on test meshes when actual noise is added to the mesh vertices and also measure the time performance of the proposed algorithm.

In Section 4, we present applications of the obtained results. As a first application, we apply the normal perturbation estimators to compute optimal levels of *dithered quantizations* of the mesh vertices when a tolerance for the accuracy of the normals is given. As a second application, we propose an adaptive high-capacity data hiding algorithm for 3D triangle meshes. The algorithm computes for each vertex the appropriate dithered quantization level i, retains the i most significant bits of each vertex coordinate and hides the secret message into the less significant bits.

1.2 Contributions and Limitations

The main contributions of the paper are:

1. An exact closed-form formula, and a linear approximation of it, for the expectation of the angle between the old and the new normal when noise is added to a single vertex of a triangle.
2. An approximation, and heuristic lower and upper bounds, for the expectation of the normal perturbation when noise is added to all three vertices of a triangle.
3. The fast computation of the dithered quantization level of a vertex when a tolerance for the degradation of the normals is given.
4. A data hiding algorithm that can claim maximum capacity for the given tolerance of normal degradation.

The main limitation of the first contribution comes from the error of the linear approximation, which is small but not negligible for small amounts of added noise. Moreover, the error increases significantly when the amount of added noise becomes large.

- *Y. Yang and I. Ivrissimtzis are with the School of Engineering and Computing Sciences, Durham University, South Road, Durham DH1 3LE, United Kingdom. E-mail: {ying.yang, ioannis.ivrissimtzis}@durham.ac.uk.*
- *N. Peyerimhoff is with the Department of Mathematical Sciences, Durham University, South Road, Durham DH1 3LE, United Kingdom. E-mail: norbert.peyerimhoff@durham.ac.uk.*

Manuscript received 11 May 2011; revised 11 Nov. 2011; accepted 3 Apr. 2012; published online 10 Apr. 2012.
Recommended for acceptance by B. Levy.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2011-05-0106.
Digital Object Identifier no. 10.1109/TVCG.2012.106.

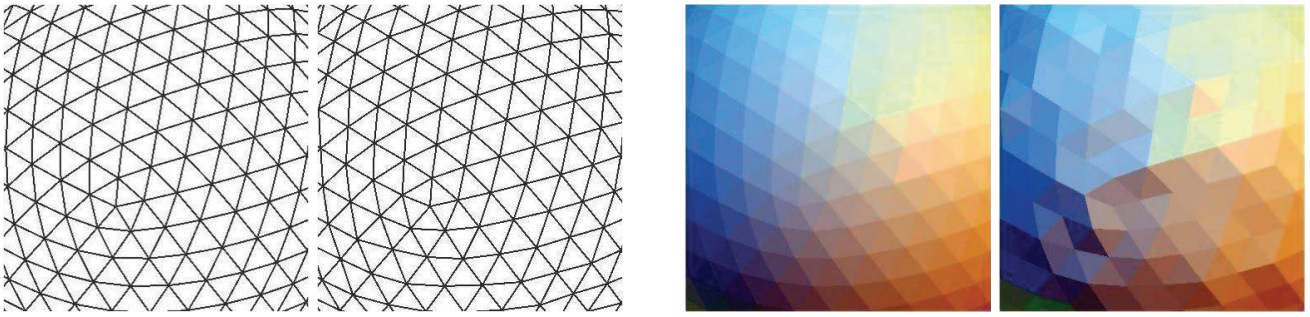

Fig. 1. Left: The wireframe renderings of a smooth and a noisy mesh. Right: The flat shaded renderings of the same meshes.

The approximation error is also a limitation of the second contribution. Moreover, the upper and lower bounds are heuristic and we have no mathematical proof that they always hold.

Regarding the third contribution, the extra limitation is the assumption that adding uniform spatial noise with support the shape of a cube is equivalent to dithered quantization. This is only approximately true when the edges of the cube do not align with the axes of the coordinate system.

The main limitation of the high-capacity data hiding method is its fragility, that is, it cannot survive attacks without loss of information.

1.3 Related Work

The problem of finding appropriate quantization levels for a mesh has been encountered in the classic predictive mesh compression algorithms of Touma and Gotsman [1] and Alliez and Desbrun [2]. In predictive encoding, the importance of removing redundant bits is further magnified by the fact that the predictions of the least significant bits are less accurate and thus more difficult to compress. Nevertheless, in all existing work the choice of quantization level is left to the user.

Face or vertex normals are used by most rendering algorithms, from the classic Gouraud and Phong algorithms, to the more computationally intensive BRDF-based rendering by Walter et al. [3]. Vertex normals can be computed from face normals in various ways, typically as a weighted mean of the face normals, such as Jin et al. [4]. If instead the vertex normals are separately encoded as a part of the mesh file, they are usually represented by vectors with three 32 bit coordinates. Meyer et al. [5] study the quantization error introduced by such representations and propose efficient normal encoding methods.

The randomization of the least significant bits of the vertex coordinates adds a high-frequency stochastic component to the geometry of the mesh. Uncertainty in polygonal meshes has been studied by Pauly et al. [6] and Kalaiah and Varshney [7]. Yoon et al. [8] propose methods for noise estimations on 3D point sets, while Sun et al. [9] study laser scan noise. In [10], Sorkine et al. discuss the visual impact of the high-frequency noise introduced by spatial quantization and an alternative quantization method based on the mesh Laplacian is proposed.

As Schuchman [11] and Gray and Stockman [12] show, dithering is a technique with strong theoretical foundations and is commonly used in audio and image processing, see, for example, Roads [13] and Akarun et al. [14]. The purpose of dithering is to avoid coarse quantization artifacts, that is, unwanted regular patterns that may distract the eye or the ear.

The choice of quantization level for the vertex coordinates can be seen as a choice of level-of-detail. The low-level quantizations correspond to coarse meshes with few triangles and the high-level quantizations, which contain more geometric detail, correspond to fine meshes. In multiresolution techniques, the level-of-detail of a mesh can be controlled either by a subdivision algorithm, see, for example, Kobbelt et al. [15] and Guskov et al. [16], or by a sequence of unitary mesh editing operations, such as edge collapses. The latter approach has been successfully applied into adapting the resolution of a mesh to the camera view, see, for example, Hoppe [17], Pasman and Jansen [18], and Hu et al. [19].

The embedding of a digital signal into a carrier, here a mesh, is referred in the literature by the terms steganography/data hiding and watermarking. Steganographic/data hiding methods aim at achieving high-embedding capacity, usually at the expense of robustness, while watermarking methods are evaluated principally on their robustness against attacks. In the majority of the existing steganographic algorithms, the carrier is the geometry of the mesh rather than the connectivity which has lower capacity. Chao et al. [20] propose a method with high-embedding capacity and low-embedding distortion, which however is not robust against attacks. Cayre and Macq [21] propose a blind data hiding method which considers each triangle as a two-state geometric object, i.e., 0 or 1 depending on the position of the projection of a vertex onto its opposite edge. The theoretical capacity of this method is one bit per vertex. Motivated by this idea, Wang and Cheng [22] increase the embedding capacity and reduce the distortion by using a multilevel embedding procedure. They further extended their work in [23], achieving even higher embedding capacity and lower distortion.

Instead of using the geometry or the connectivity as carrier, Bogomjakov et al. [24] propose a 3D distortion-free steganographic method which exploits the redundancy in the indexed representation of a mesh by permuting the order in which faces and vertices are stored. Tu et al. [25] improve the efficiency in the original mapping of Bogomjakov et al. [24] further increasing its capacity.

2 Normal Noise Estimation

In our model, the noise added to a vertex P of the triangle mesh is described by a random variable u with distribution $\mathbf{p}(u)$, $u \in \mathbf{R}^3$. To measure the effect of the added random

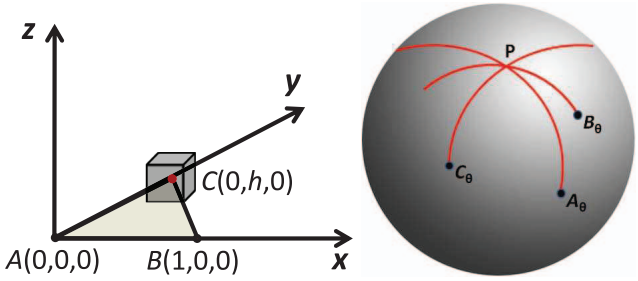

Fig. 2. Left: Adding noise to a single triangle vertex. Right: Normal orbits on the Gaussian sphere.

variable on the normal of a triangle T incident to P, we compute the expectation $\mathrm{E}(\theta(u))$ for the angle $\theta(u)$ between the normal of T and its normal after u has been added to P. The expectation is given by

$$\mathrm{E}(\theta(u)) = \int_\Omega \mathbf{p}(u)\theta(u)du, \tag{1}$$

where Ω is the support of $\mathbf{p}(u)$.

The expectation in (1) can be computed by standard numerical integration methods for any given probability distribution that may appear in practice, for example, the Gaussian or the uniform distributions. In what follows, we study uniform noise with cubic shaped support. In practice, such types of noise are the result of the randomization of the least significant bits of the vertex coordinates and appear in applications such as dithered quantization and data hiding. In particular, we will compute the integral in (1) in a closed form and find local linear approximations of it. To simplify the notation we will write θ instead of $\theta(u)$ when the context removes any ambiguity.

2.1 Adding Uniform Noise to a Single Vertex

Let $T = ABC$ be a triangle. We will compute $\mathrm{E}(\theta)$ when u is added to the vertex C with $\mathbf{p}(u)$ the uniform distribution with support a cube of edgelength $2l$, centered at C with one face parallel to ABC, see Fig. 2(left). In this case, $\mathbf{p}(u)$ is equal to the constant $1/8l^3$ inside the cube and zero outside.

By a basis change of the Cartesian xyz coordinate system, including scaling, we may assume without loss of generality that $A = (0,0,0)$, $B = (1,0,0)$, and C lies in the xy-plane. We notice that moving $C + u$ parallely to the x-axis by adding a displacement $t\vec{AB}$ does not change $\mathrm{E}(\theta)$ because the two triplets of points $(A, B, C + u)$ and $(A, B, C + u + t\vec{AB})$ define the same plane, hence have the same normal. That means that without loss of generality we may assume that $C = (0, h, 0)$, where h is the height of ABC at C, see Fig. 2(left).

Intuitively, the above simplification of the problem is based on the fact that if we fix A, B and add the random variable u to C then $\mathrm{E}(\theta)$ depends on the distance h of C from AB only. Formally, the simplification is a consequence of the equality

$$\theta(u) = \theta(u + t\vec{AB}), \tag{2}$$

giving,

$$\int_{\Omega + t\vec{AB}} (1/8l^3)\theta(u + t\vec{AB})du = \int_\Omega (1/8l^3)\theta(u)du. \tag{3}$$

Let $u = (x, y, z)$, we have

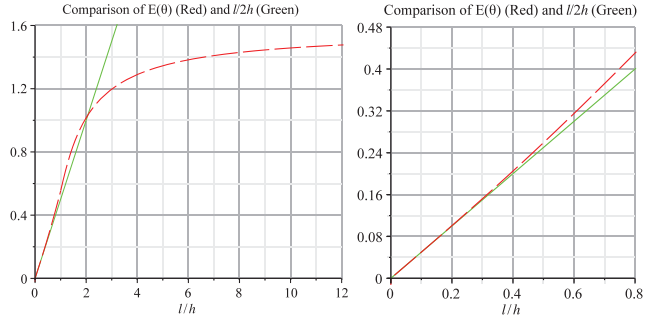

Fig. 3. Left: Plot of $\mathrm{E}(\theta)$ and its tangent at 0, $h = 1$. Right: A close-up of the graph.

$$\theta = \begin{cases} \arctan(|z|/(h+y)), & \text{if } y > -h \\ \pi/2, & \text{if } y = -h \\ \pi + \arctan(|z|/(h+y)), & \text{if } y < -h. \end{cases} \tag{4}$$

The simple form of the components of (4) allows the computation of $\mathrm{E}(\theta)$ in closed form. First, we notice that because of the invariance of θ when u is reflected through the xy-plane, it suffices to compute $\mathrm{E}(\theta)$ for $z \geq 0$ only. Moreover, the invariance of θ when a vector parallel to the x-axis is added to u means that we do not have to integrate along x.

For $l < h$, which is the main case as it represents the tolerable amounts of noise, we get

$$\begin{aligned} E(\theta) &= \frac{1}{2l^2}\int_{-l}^{l}\int_{0}^{l} \theta \, dz \, dy \\ &= \frac{1}{2l^2}\int_{h-l}^{h+l}\int_{0}^{l} \arctan(z/y) \, dz \, dy, \end{aligned} \tag{5}$$

and the definite integral can be directly computed from the indefinite integral

$$\begin{aligned} \int\int \arctan(z/y) \, dy \, dz &= yz \, \arctan(z/y) \\ &+ [(z^2 - y^2)\log(y^2 + z^2) + y^2]/4 + c, \end{aligned} \tag{6}$$

giving,

$$\begin{aligned} \mathrm{E}(\theta) = \frac{1}{8r^2} log\left(\frac{((1-r)^2 + r^2)^{1-2r}(1+r)^{2(r+1)^2}}{((1+r)^2 + r^2)^{1+2r}(1-r)^{2(1-r)^2}}\right) \\ + \frac{1+r}{2r}\arctan\left(\frac{r}{1+r}\right) - \frac{1-r}{2r}\arctan\left(\frac{r}{1-r}\right), \end{aligned} \tag{7}$$

where $r = l/h$.

The case $l > h$ in (4) differs from the case $l < h$ only by a constant and one just has to add

$$\frac{1}{2l^2}\int_{h-l}^{0}\int_{0}^{l} \pi \, dz \, dy = \frac{\pi(r-1)}{2r}, \tag{8}$$

to (7) to compute $\mathrm{E}(\theta)$ for that case. In both cases it turns out that $\mathrm{E}(\theta)$ depends on the quotient l/h rather than the individual values of h and l, meaning that the expectation is scale invariant.

Simple computations, using del'Hospital's rule when necessary, show that the derivative of $\mathrm{E}(\theta)$ at $r = 0$ is equal to $1/2$. Fig. 3 shows the graph of $\mathrm{E}(\theta)$ and its tangent at 0. We notice that for relatively small values of l/h, the tangent approximates well the curve of $\mathrm{E}(\theta)$ and can be used as an approximation of its value by

TABLE 1
The Linear Approximation of $\mathrm{E}(\theta)$

l/h	.1	.2	.3	.4	.5
$\mathrm{E}(\theta) - (l/2h)$	.0001	.0006	.0021	.0049	.0092

$$\mathrm{E}(\theta) \simeq r/2 = l/2h. \tag{9}$$

Table 1 shows the differences between the curve of $\mathrm{E}(\theta)$ and its tangent at 0 for several values of l/h. For the rest of the paper, we will mostly use this approximate value of $\mathrm{E}(\theta)$, even though the exact value can be computed from (7).

2.2 Adding Noise to All Three Vertices

The next step is to find heuristic bounds for $\bar{\mathrm{E}}(\theta)$, the expectation for the angle between the old and the new normal when noise is added on all three vertices of T. We also compute a point estimate of $\bar{\mathrm{E}}(\theta)$, using a simple linear function of the expectations $\mathrm{E}(\theta)$ computed on the three vertices T as described in Section 2.1.

Let $h_1 \leq h_2 \leq h_3$ be the heights of T at A, B, and C, respectively, and let $\mathrm{E}(\theta_1) \geq \mathrm{E}(\theta_2) \geq \mathrm{E}(\theta_3)$ be the corresponding expectations when noise is added to a single vertex of the triangle. We propose

$$\mathrm{E}_{min}(\theta) = \mathrm{E}(\theta_1), \tag{10}$$

as a heuristic lower bound for $\bar{\mathrm{E}}(\theta)$.

The heuristic argument is that if after adding noise to one of the triangle's vertices we proceed and add noise to the other two vertices too, we will increase the spatial uncertainty and thus increase the expectation for the normal perturbation $\bar{\mathrm{E}}(\theta)$. To see the same argument from a slightly different angle, a lower expectation for θ would mean that the normal of the triangle with noise on its three vertices is a better estimate of the original normal than the normal of the triangle with noise on a single vertex. As we cannot improve the estimate by adding more noise, we expect that $\bar{\mathrm{E}}(\theta) \geq \mathrm{E}(\theta_1) \geq \mathrm{E}(\theta_2) \geq \mathrm{E}(\theta_3)$.

For the heuristic construction of an upper bound for $\bar{E}(\theta)$, we treat the normal pertubations from the addition of noise on each vertex as independent. This is a valid assumption when the amount of noise added to each vertex is small and does not change significantly the heights of T. Under this independence assumption, an obvious upper bound for $\bar{\mathrm{E}}(\theta)$ is the sum

$$\mathrm{E}(\theta_1) + \mathrm{E}(\theta_2) + \mathrm{E}(\theta_3). \tag{11}$$

We notice that for an equilateral triangle T we have

$$\mathrm{E}(\theta_1) + \mathrm{E}(\theta_2) + \mathrm{E}(\theta_3) = 3\ \mathrm{E}_{min}(\theta), \tag{12}$$

meaning that for the equilateral and almost equilateral triangles that are common in high-quality meshes the pair of bounds in (10) and (11) is not tight enough in the sense that the quantization problems we deal with in the applications will have more than one solution. Indeed, in such problems the quantization levels are integers solutions in a logarithmic space of basis 2 and, ideally, for a unique solution we would like the lower and the upper bounds to differ by a factor of 2 at most. For the construction of a sharper heuristic upper bound, we sum the expected normal perturbations on the Gaussian sphere, using the fact that when noise is added to a single vertex, the trajectory on the Gaussian sphere of the possible normal perturbations is a circle.

Assume that the normal of T maps to the top of the Gaussian sphere P. By adding noise to the vertex A only, the orbit of the perturbed normal on the Gaussian sphere is a maximal arc with center at P. Let A_θ be a point on that arc such that $Arc(PA_\theta) = \mathrm{E}(\theta_1)$. Similarly, if we add noise to the vertex B only, the orbit of the perturbed normals is a maximal arc with center at P, and let B_θ be a point on that arc such that $Arc(PB_\theta) = \mathrm{E}(\theta_2)$. The point C_θ is constructed similarly, see Fig. 2(right). For a small l, the points P, A_θ, B_θ, C_θ are almost coplanar and we can approximate A_θ, B_θ, C_θ with their projections $A_\theta^{'}$, $B_\theta^{'}$, $C_\theta^{'}$ on the tangent of the Gaussian sphere at P. Working on that tangent plane instead of the sphere, we expect the angle corresponding to the sum

$$\max\{\pm \vec{PA_\theta^{'}} \pm \vec{PB_\theta^{'}} \pm \vec{PC_\theta^{'}}\}, \tag{13}$$

to be an upper bound because (13) will choose for each vector the direction that will maximize the sum.

Working on the tangent plane at P, we notice that the angles of the intersection of the three lines $PA^{'}$, $PB^{'}$, $PC^{'}$ are equal to the angles of the intersection of the heights of the triangle which are equal to the angles of the triangle. We also notice that the lengths of the vectors of (13) are proportional to the inverses of the triangle's heights, meaning that they are proportional to the triangle's edges. Simple arguments show that if h_a is the smallest height of the triangle

$$\max\{\pm\vec{PA_\theta^{'}} \pm \vec{PB_\theta^{'}} \pm \vec{PC_\theta^{'}}\} = 2\vec{PA_\theta^{'}} \simeq 2\vec{PA_\theta}, \tag{14}$$

and the proposed upper bound becomes

$$\mathrm{E}_{max}(\theta) = 2\vec{PA_\theta} = 2\ \mathrm{E}_{min}(\theta). \tag{15}$$

Finally, for obtaining a point estimation of $\bar{\mathrm{E}}(\theta)$ we use the simple linear approximation

$$\mathrm{E}_{appr}(\theta) = \eta \cdot (\mathrm{E}(\theta_1) + \mathrm{E}(\theta_2) + \mathrm{E}(\theta_3)). \tag{16}$$

The constant η is estimated experimentally. To estimate η we simulated the addition of a small amount of noise, $l = 2^{-10}$ in particular, to the vertices of an equilateral triangle of size 1, and after averaging over a large number of experiments we found the value $\eta = 0.608$.

Notice that, based on (9), (16) can be written as

$$\mathrm{E}_{appr}(\theta) = \eta \cdot l \cdot \left(\frac{1}{2h_1} + \frac{1}{2h_2} + \frac{1}{2h_3}\right) = \frac{\eta \cdot l}{2\rho}, \tag{17}$$

where ρ is the radius of the incircle of T. In other words, a consequence of the approximation $\mathrm{E}_{appr}(\theta)$ is that, for small amounts of noise, $\mathrm{E}(\theta)$ is linear to the inverse of the radius ρ of the incircle.

3 Tests and Validation

In this section, we test the obtained bounds and approximations on several well-known triangle meshes that are available online and measure the time performance of the proposed algorithm.

3.1 Validation

Let

$$\mathrm{E}_{mesh}(\theta) = \frac{\sum_{j=1}^{M} \bar{\mathrm{E}}_j(\theta)}{M}, \tag{18}$$

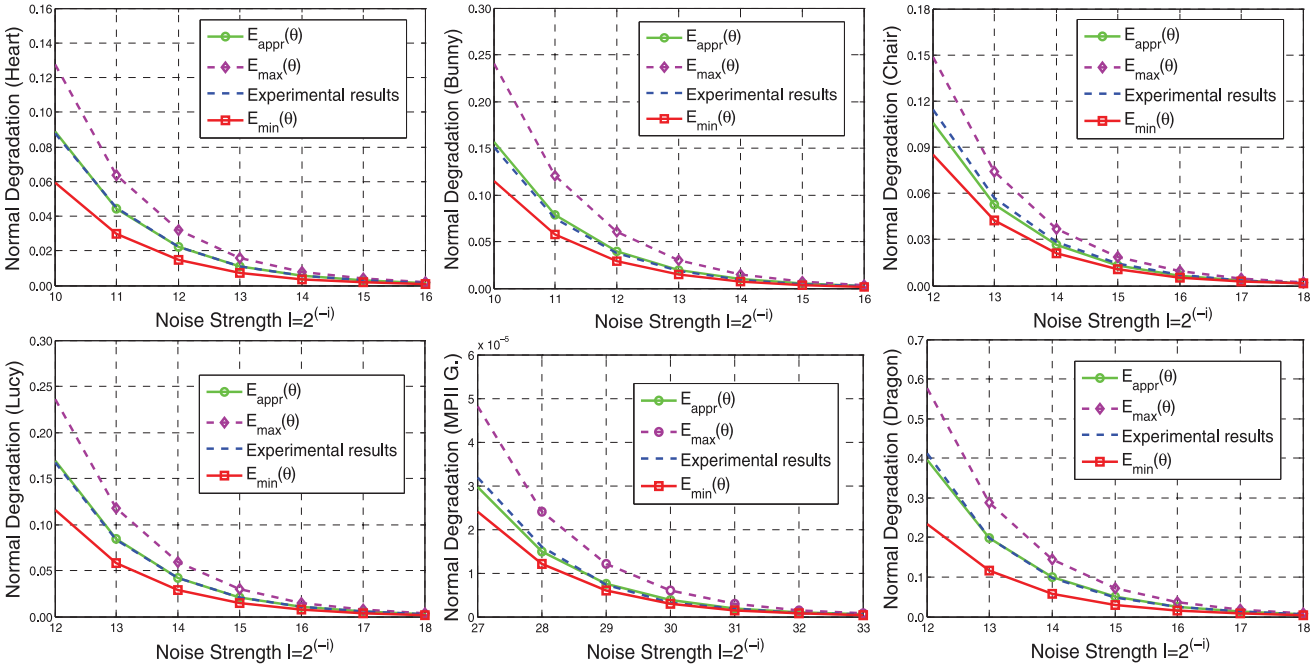

Fig. 4. Validation results for *Heart*, *Bunny*, *Chair*, *Lucy*, *MPII Geometry*, and *Welsh Dragon*. The x-axis corresponds to the values of i. The y-axis is in radians.

be the average normal degradation over the triangles of a mesh. Here, M is the number of triangles of the mesh and $\bar{\mathrm{E}}_j(\theta)$ denotes the expectation $\bar{\mathrm{E}}(\theta)$ for the normal degradation of the jth triangle.

To validate the accuracy of the bounds and the point approximation, we replaced $\bar{\mathrm{E}}_j(\theta)$ in (18) with $\mathrm{E}_{min}(\theta)$, $\mathrm{E}_{max}(\theta)$, and $\mathrm{E}_{appr}(\theta)$, respectively, and compared it with the average normal degradation we computed experimentally using synthetic added noise.

The amounts of added noise were powers of 2, that is, $l = 2^{-i}$. The values of l were small enough for the linear correlation between $\bar{\mathrm{E}}(\theta)$ and l/h to be strong. The results are summarized in Fig. 4.

As expected, $\mathrm{E}_{min}(\theta)$ and $\mathrm{E}_{max}(\theta)$ bound $\bar{\mathrm{E}}(\theta)$ from below and above and thus $\mathrm{E}_{mesh}(\theta)$ too. Moreover, $\mathrm{E}_{appr}(\theta)$ provides a good approximation of the experimental results. Notice that the graphs in Fig. 4 are drawn for values of i that are large enough for the linear approximation in (9) to be valid.

In some cases, if the mesh contains many small and skinny triangles the linear approximation may not be valid even for relatively large values of i. Fig. 5(top) shows that in the *MPII Geometry* model $\mathrm{E}_{appr}(\theta)$ is below the lower bound $\mathrm{E}_{min}(\theta)$ even for levels of quantization $i > 20$. In these cases, one should use the exact expectation in (7) instead of the approximation in (9). Indeed, Fig. 5(bottom) shows that the use of the exact expectation gives satisfactory results even for the range $12 \leq i \leq 18$.

3.2 Time Performance

Regarding its computational complexity, the algorithm is linear with the number of triangles. Given a normal degradation tolerance ϵ, our current nonoptimized *Matlab* implementation on an Intel Core 2 Duo E8400 3.00 GHz processor with 2 GB memory calculates the optimal quantization level for the 2M triangles of the *Welsh Dragon* in 56 seconds. Apart from programming optimizations, we

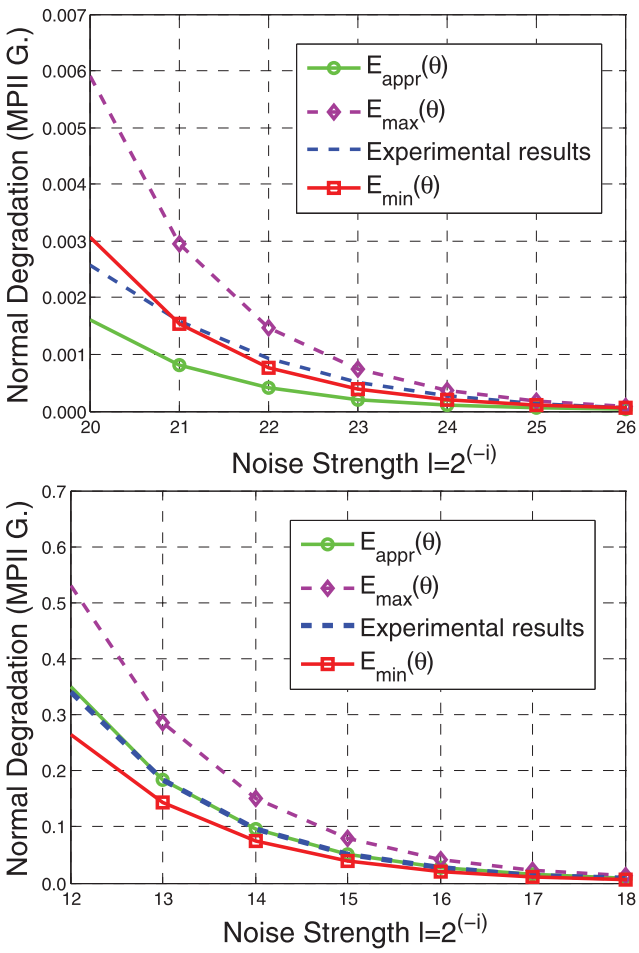

Fig. 5. For the *MPII Geometry* model, the linear approximation of the expectation gives poor results even for relatively high values $i > 20$. However, the use of the exact expectation gives satisfactory results even for the lower range $12 \leq i \leq 18$.

TABLE 2
For Each of $\mathrm{E}_{min}(\theta)$, $\mathrm{E}_{appr}(\theta)$, and $\mathrm{E}_{max}(\theta)$, the Left, Middle, and Right Columns Show the Quantization Levels Corresponding to $\epsilon = 0.1^\circ$, $\epsilon = 1^\circ$, and $\epsilon = 10^\circ$, Respectively

	#Tri	$\mathrm{E}_{min}(\theta)$			$\mathrm{E}_{appr}(\theta)$			$\mathrm{E}_{max}(\theta)$		
Heart	37690	15	11	8	15	12	9	16	12	9
Bunny	69666	16	12	9	16	13	9	17	13	10
Chair	6664	17	14	10	17	14	11	18	15	11
Lucy	525814	18	14	11	18	15	11	19	15	12
MPII Geometry	70761	20	16	12	20	16	13	21	17	13
Welsh Dragon	2210635	19	15	12	19	16	13	20	16	13

TABLE 3
The Experimental Values of Normal Degradation ψ^i Corresponding to the Quantization Levels Given by the $\mathrm{E}_{appr}(\theta)$ Column in Table 2

	$\epsilon = 0.1^\circ$	$\epsilon = 1^\circ$	$\epsilon = 10^\circ$
Heart	$\psi^{15} = 0.111$	$\psi^{12} = 0.884$	$\psi^{9} = 7.064$
Bunny	$\psi^{16} = 0.099$	$\psi^{13} = 0.750$	$\psi^{9} = 12.276$
Chair	$\psi^{17} = 0.099$	$\psi^{14} = 0.851$	$\psi^{11} = 7.461$
Lucy	$\psi^{18} = 0.105$	$\psi^{15} = 0.837$	$\psi^{11} = 13.543$
MPII Geometry	$\psi^{20} = 0.092$	$\psi^{16} = 0.892$	$\psi^{13} = 6.496$
Welsh Dragon	$\psi^{19} = 0.123$	$\psi^{16} = 0.982$	$\psi^{13} = 7.684$

can significantly speed up the computation by using the much simpler (10) instead of (17). Indeed, as (17) is between the lower bound in (10) and the upper bound in (15), and as the lower and the upper bounds differ by 2, the results of (10) will differ from the results of (17) by one bit at most.

Finally, we note that the linear correlation between spatial and normal noise can be used without any computational cost as a rule of thumb that one extra bit will halve the normal error. This simplicity is the main motivation for using the approximations of section 2.2 for all the examples, instead of using the exact formula obtained from (7).

4 Applications

In this section, we discuss how the proposed method can be used for obtaining optimal dithered quantizations as well as for adaptive high-capacity mesh data hiding.

4.1 Optimal Dithered Quantizations

A simple dithered quantiser for the mesh vertices retains the i most significant bits of each vertex coordinate and replaces any less significant bit with random bits. Such dithered quantisers avoid the blocky artifacts created by quantisers that set all less significant bits to zero.

Given a tolerance ϵ for the average normal degradation $\mathrm{E}_{mesh}(\theta)$, the bounds and the point estimate of Section 2 can be used to compute optimal levels for the dithered quantization of the vertices. The main assumption is that the randomization of all bits that are less significant than bit k can be approximated by the addition of uniform noise with cubic support of edgelength $2l = 2^{-k}$.

In particular, we can compute the optimal level of quantization i as

$$i = \arg\max_{k \in \mathbb{Z}} \{k \mid \mathrm{E}_{mesh}(\theta) \leq \epsilon\}, \tag{19}$$

where, depending on the application, the average $\mathrm{E}_{mesh}(\theta)$ is computed using the lower bound $\mathrm{E}_{min}(\theta)$, or the upper bound $\mathrm{E}_{max}(\theta)$, or the estimate $\mathrm{E}_{appr}(\theta)$. For instance, if we use $\mathrm{E}_{min}(\theta)$, from (9), (10), (18), and (19), we get

$$i = \left\lceil -2 - \log_2 \left(\frac{M \cdot \epsilon}{\sum_{j=1}^{M} 1/h'_j} \right) \right\rceil. \tag{20}$$

Similarly, if we use $\mathrm{E}_{appr}(\theta)$, then from (17), (18), and (19), we obtain

$$i = \left\lceil -2 - \log_2 \left(\frac{M \cdot \epsilon}{\eta \cdot \sum_{j=1}^{M} 1/\rho_j} \right) \right\rceil. \tag{21}$$

In the above two equations, $\lceil \cdot \rceil$ stands for the mathematical ceiling function and h'_j and ρ_j are the minimum height and the incircle radius of the jth triangle, respectively.

Table 2 summarizes the results for several values of the tolerance ϵ. The quantization levels for *MPII Geometry* were computed using the exact (7) instead of (17).

We notice that the size of the mesh influences the quantization level. The reason is that the number of triangles in the common natural meshes is correlated with their average size, with the large meshes mainly consisting of small triangles with normals that are more sensitive noise. For an information-theoretic explanation, we can say that larger meshes contain more geometric detail, which requires more bits per vertex coordinate for its representation. The shape of the mesh triangles also influences the level of quantization. In particular, after taking the number of mesh triangles into account, the CAD meshes *Chair* and *MPII Geometry*, which have a significant portion of skinny triangles, require more bits per coordinate compared to scan meshes such as *Lucy* and *Welsh Dragon* which mainly consist of well-shaped triangles. For an information-theoretic explanation, CAD meshes usually contain more information per triangle than the scan meshes and thus, require more bits per vertex coordinate.

Table 3 shows that the experimental values of the normal degradation computed as

$$\psi^i = \frac{\sum_{j=1}^{M} \mathrm{angle}\,(\hat{\mathbf{n}}_j, \hat{\mathbf{n}}^i_j)}{M}, \tag{22}$$

where $\hat{\mathbf{n}}_j$ and $\hat{\mathbf{n}}^i_j$ are the normals of the jth triangle of the original mesh and the mesh quantized at i bits, respectively. For a given tolerance ϵ, the level of quantization i is the one corresponding to $\mathrm{E}_{appr}(\theta)$ as shown in Table 2. We notice that, in all cases, the experimental results differ from the prescribed tolerance by a factor smaller than two, meaning that the computed quantization levels differ from the optimal for that tolerance levels by one bit at most. On the other hand, we also notice that in four cases the experimental normal degradation exceeds the tolerance, even though the quantization levels corresponding to $\mathrm{E}_{appr}(\theta)$ happen to be the same as the ones corresponding to $\mathrm{E}_{min}(\theta)$. In Section 1.2, in the discussion of the limitations of the method, we discuss several sources of error that might be responsible for this discrepancy.

Optimally quantized test meshes for several values of ϵ are shown in Fig. 6 and close-ups of some of them are shown in Fig. 7. As expected, for all test meshes, the quantizations corresponding to tolerances $\epsilon = 0.1^\circ$ and $\epsilon = 1^\circ$ are almost indistinguishable between them and visually equivalent to

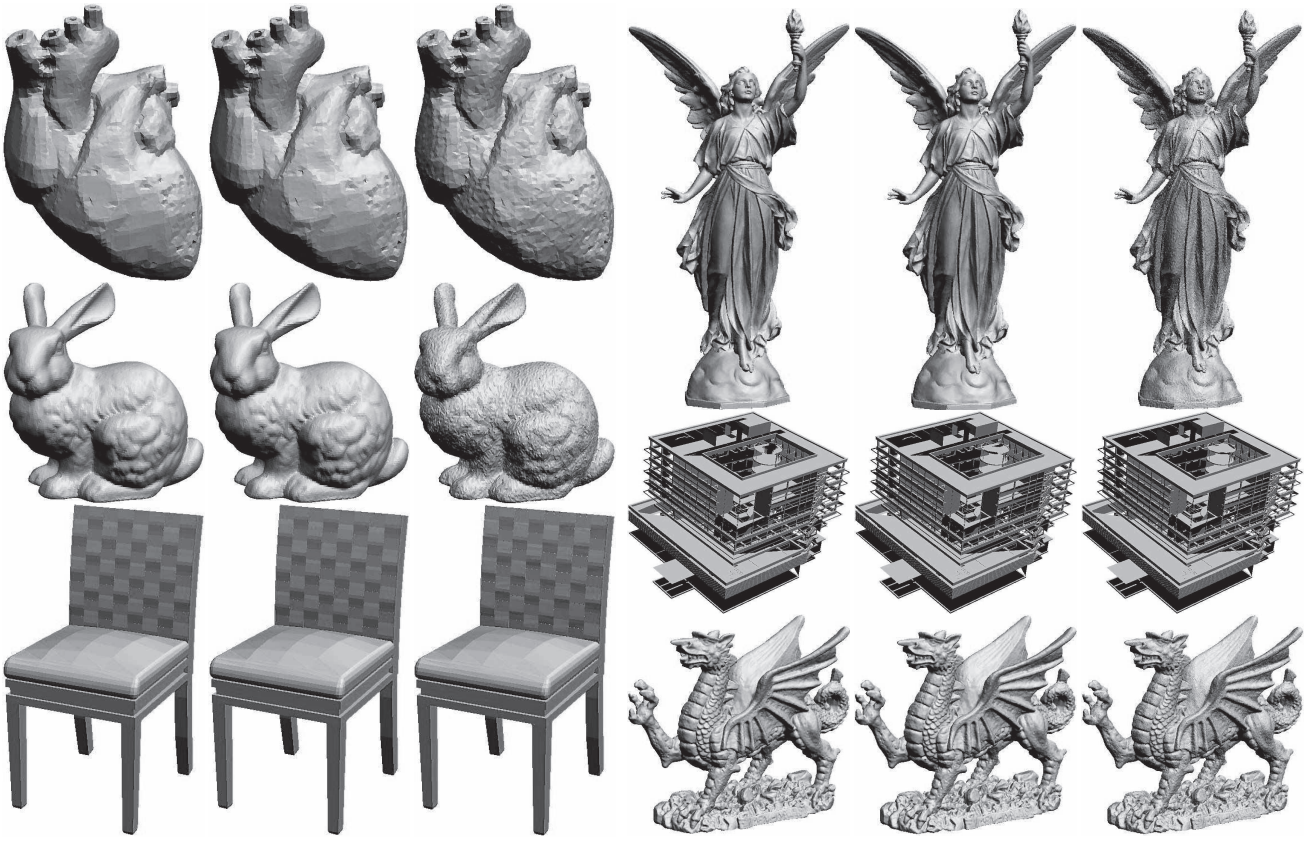

Fig. 6. For each mesh group, the left, middle, and right show the dithered quantized meshes for $\epsilon = 0.1°$, $\epsilon = 1°$, and $\epsilon = 10°$, respectively.

the originals. By contrast, the meshes that have suffered an average $\epsilon = 10°$ normal degradation are of noticeably inferior visual quality.

In some applications, the average normal degradation over all mesh triangles may not be a satisfactory indicator of mesh quality. One such case is when the size or the shape of the mesh triangles is not uniform across the mesh. Another issue limiting the visual relevance of the average normal degradation is that the human eye sensitivity to normal perturbations depends on the smoothness of the surface areas where the degradation occurred. In particular, the eye is more sensitive to normal perturbations in the smoother regions of a surface. Hence, some applications might require a normal degradation tolerance that is smaller at the smoother areas of the surface. In such cases, computing a separate optimal level of quantization for each mesh vertex may give a more satisfactory solution.

To do this, we use the Gaussian curvature κ_n given by

$$\kappa_n = 2\pi - \sum_{i=1}^{v_n} \alpha_{n,i}, \quad 1 \leq n \leq N, \tag{23}$$

as a measure of mesh smoothness at a mesh vertex, where N is the number of the mesh vertices, v_n is the degree of the nth vertex, and $\alpha_{n,i}$ is the ith angle incident the nth vertex. The smaller the absolute value $|\kappa_n|$ is, the flatter the surface at the vicinity of the nth vertex.

The normal degradation tolerance ϵ_j at the jth triangle is given by

$$\epsilon_j = \min_n \{\alpha - \phi(\kappa_n; \alpha, \mu, \sigma) + \epsilon_0\}, \ 1 \leq j \leq M, \tag{24}$$

where $\phi(x; \alpha, \mu, \sigma)$ is the Gaussian function with peak value α, mean μ, and standard deviation σ, the minimum is over the three vertices of the jth triangle and ϵ_0 is the threshold of normal degradation that is considered acceptable for a planar surface. The quantization level i of each triangle is then computed by

$$i = \arg\max_{k \in \mathbb{Z}} \{k \mid \bar{\mathrm{E}}_j(\theta) \leq \epsilon_j\}. \tag{25}$$

Finally, the quantization level of a vertex is computed as the maximum level of quantization of the incident triangles.

Fig. 8 shows adaptive quantizations corresponding to parameter values $\alpha = 0.7979$, $\mu = 0$, $\sigma = 0.5$, $\epsilon_0 = 0.0017 \simeq 0.1°$ and $\bar{\mathrm{E}}_j(\theta)$ approximated by $\mathrm{E}_{appr}(\theta)$. As expected the scanned meshes tend to have a more uniform and smooth distribution of quantization levels compared to the CAD meshes. Despite the higher cost of computing an adaptive, curvature depended quantization, the method is still local and thus, linear in complexity with respect to the size of the mesh.

4.2 Adaptive High-Capacity Data Hiding

The challenge in designing a good data hiding algorithm lies in balancing two conflicting requirements: high-*embedding capacity* and low-*embedding distortion*. The obtained relations between spatial and normal noise describe a tradeoff between embedding capacity, in the form of unused bits in the vertex coordinates, and visual distortion, in the form of normal degradation. As such, they can be directly used to inform least significant bit data hiding algorithms about the embedding capacity of the carrier for a given distortion tolerance.

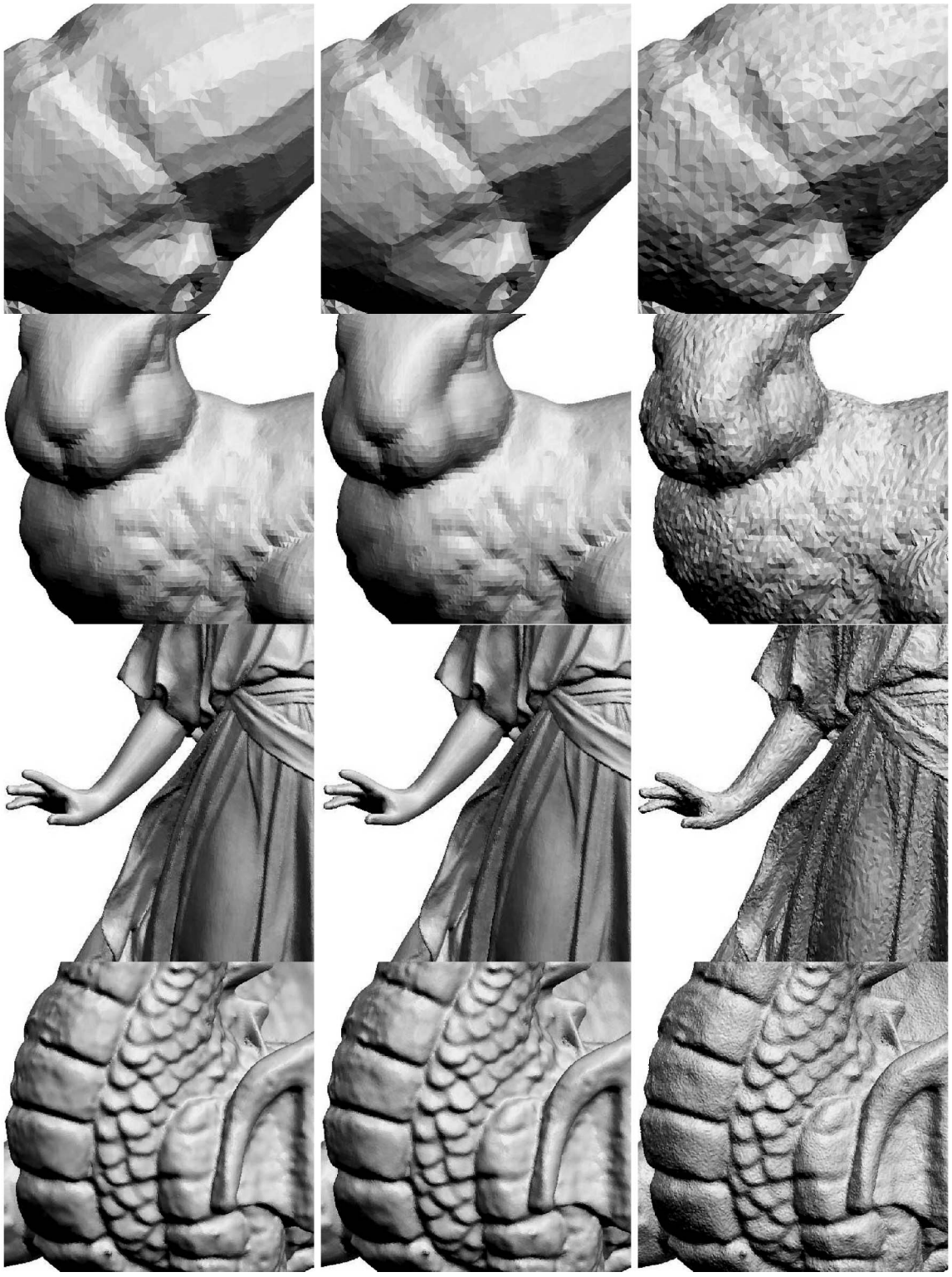

Fig. 7. Close-ups of the quantized *Heart*, *Bunny*, *Lucy*, and *Welsh Dragon* models in Fig. 6. The left, middle, and right columns correspond to $\epsilon = 0.1°$, $\epsilon = 1°$, and $\epsilon = 10°$, respectively.

To make the data hiding algorithm blind and robust against vertex permutation, we use two orderings $\pi_1(\cdot)$ and $\pi_2(\cdot)$ of the vertex set, which are obtained from the projections of the sets of unmarked $\mathbf{v}$ and marked $\mathbf{v}'$ quantized vertices onto their respective *first principal axes*. These two orderings will assist the extraction process in finding the embedding order of the message bits, avoiding thus potential synchronization problems. The details of the embedding and extraction algorithms are as follows:

Embedding. Given the carrier triangle mesh with set of vertices $\mathbf{v} = (v_1, v_2, \ldots, v_N)$, represented in 32 bits, and a normal degradation tolerance ϵ, we compute the vector $\mathbf{i} = (i_1, i_2, \ldots, i_N)$ of quantization levels as in Section 4.1. All unused vertex coordinate bits are set to zero. The message

Fig. 8. Vertices with high- and low-quantization levels are shown in red and blue color, respectively. From left to right: the ranges of the quantization levels are [6, 20], [4, 21], [12, 26] and [2, 32] bits per vertex coordinate.

bits are embedded by a bit-replacement operation. More specifically, to mark the jth vertex, we retain the i_j most significant bits and the c least significant bits of its coordinates and sequentially replace the other $32 - i_j - c$ bits with message bits. The embedding uses the vertex ordering $\pi_1(\mathbf{v})$. The vertex set obtained this way is denoted by $\mathbf{v}' = (v'_1, v'_2, \ldots, v'_N)$.

The c least significant bits of the vertex coordinates of $\mathbf{v}'$ will carry side information recording changes in the quantization levels between the vertices of the marked and unmarked meshes. In particular, we compute the vector of quantization levels $\mathbf{i}' = (i'_1, i'_2, \ldots, i'_N)$ of $\mathbf{v}'$ and from it and the original quantization levels $\mathbf{i}$ the corrective vector

$$\mathbf{e} = (i_1 - i'_1, i_2 - i'_2, \ldots, i_N - i'_N). \tag{26}$$

We sort the components of $\mathbf{e}$ according to the ordering $\pi_2(\cdot)$ and $\pi_2(\mathbf{e})$ is compressed and embedded into the c least significant bits of the vertex coordinates of $\mathbf{v}'$. The embedding follows the vertex ordering $\pi_2(\mathbf{v}')$. Notice that $\mathbf{e}$ can be compressed very efficiently, see Table 4. Thus, we can always use $c = 1$ and embed it in the least significant bits of the vertex coordinates.

Extraction. Given the marked mesh and the normal degradation tolerance ϵ, the key for extracting the message bits is to recover the orderings $\pi_1(\cdot)$ and $\pi_2(\cdot)$.

To compute $\pi_2(\cdot)$, we set the c least significant bits of the marked mesh to zero and quantize the resulting vertex set, which is a permutation of $\mathbf{v}'$, as in Section 4.1. The quantized vertices are projected onto their first principal axis. Notice that because the first principal axis is invariant under vertex permutation the ordering obtained from the projection is $\pi_2(\cdot)$. Having recovered $\pi_2(\cdot)$, the corrective vector $\pi_2(\mathbf{e})$ is obtained by extracting and decompressing the c least significant bits of the input marked mesh in the order of $\pi_2(\cdot)$.

To recover $\pi_1(\cdot)$, we first obtain the permuted quantized levels $\pi_2(\mathbf{i})$ of the original mesh by

$$\pi_2(\mathbf{i}) = \pi_2(\mathbf{e}) + \pi_2(\mathbf{i}'). \tag{27}$$

Next, we quantize the vertices of $\pi_2(\mathbf{v}')$ at the original levels $\pi_2(\mathbf{i})$, obtaining the $\pi_2(\cdot)$ ordering of the unmarked quantized vertices. Using again the invariance of first principal axis under vertex permutation, the projection of the unmarked quantized vertices on their first principal axis gives $\pi_1(\cdot)$.

Having recovered $\pi_1(\cdot)$, we can now extract the message from the marked vertices $\pi_1(\mathbf{v}')$ and the quantization levels $\pi_1(\mathbf{i})$ of the unmarked vertices. To extract information from the jth vertex, we just remove the i_j most significant bits and the c least significant bits of its coordinates and consider the remaining $32 - i_j - c$ bits as the embedded message.

Fig. 9 shows the models of *Bunny* and *Lucy* marked with various normal degradation tolerances ϵ. The quantization levels were computed using (25) with equal $\epsilon_j = \epsilon$ for all triangles. As expected, the meshes with lower degradation

TABLE 4
For Each of $\epsilon = 0.1°$, $\epsilon = 1°$, and $\epsilon = 10°$, the Left and the Right Columns Show the Average Embedding Capacity in Bits per Vertex and the Ratio of Nonzero Components in the Corrective Vector $\mathbf{e}$ to the Number of Mesh Vertices, Respectively

	$\epsilon = 0.1°$		$\epsilon = 1°$		$\epsilon = 10°$	
Heart	≈ 48.59	1.28×10^{-3}	≈ 58.40	1.22×10^{-2}	≈ 68.26	1.59×10^{-1}
Bunny	≈ 46.79	1.15×10^{-4}	≈ 56.14	1.37×10^{-2}	≈ 67.58	5.46×10^{-1}
Chair	≈ 41.26	3.25×10^{-2}	≈ 51.46	2.93×10^{-4}	≈ 60.66	1.27×10^{-1}
Lucy	≈ 39.85	1.62×10^{-3}	≈ 49.88	1.61×10^{-2}	≈ 59.74	1.54×10^{-1}
MPII Geometry	≈ 37.63	6.69×10^{-4}	≈ 47.60	2.61×10^{-2}	≈ 57.34	2.13×10^{-1}
Welsh Dragon	≈ 37.61	1.99×10^{-3}	≈ 47.00	2.77×10^{-4}	≈ 56.01	5.99×10^{-3}

Fig. 9. The left, middle, and right columns show the marked meshes for $\epsilon = 0.1^\circ$, $\epsilon = 1^\circ$, and $\epsilon = 10^\circ$, respectively.

tolerance are of higher visual quality. In particular, the embedding distortion is almost invisible for $\epsilon = 0.1^\circ$ and $\epsilon = 1^\circ$, while it is clearly visible for $\epsilon = 10^\circ$.

Table 4 reports the embedding capacity of each of the test meshes under various degradation tolerances, and the ratio of nonzero components in the corrective vector $\mathbf{e}$ to the number of mesh vertices. Capacity results for 16 or 24 bit quantizations of the same meshes, which often appear in practice, can be obtained from the results in Table 4 after subtracting 16 or 8 bits per coordinate, respectively, provided that these coarser quantizations do not degrade the normal beyond the prescribed tolerance.

We notice that, as expected, the capacity increases monotonically with ϵ, which implies that the extra capacity is achieved at the expense of increased embedding distortion. As expected, in terms of capacity the proposed method outperforms all of the previously reported algorithms [20], [21], [26], [27], [28] for hiding information in the geometry of a 3D mesh.

Discussion. The main advantages of the proposed data hiding algorithm are:

1. It is a high-capacity method which, for $\epsilon = 1^\circ$, is able to embed at least 10 more bits per vertex than the current state of the art for embedding information on mesh geometry [20].
2. It gives the user control over the tradeoff between capacity and embedding distortion. That is, we can embed the appropriate payload knowing that we do not violate the application's requirements on visual quality.

The main disadvantage of the algorithm is its fragility. Indeed, even slight attacks may lead to the loss of most of the embedded information.

5 Conclusion

We studied the relationship between spatial and normal noise in a triangle mesh. We explicitly computed the expected angle $E(\theta)$ between the old and the new normal when a small amount of uniform random noise with cubic support of edgelength $2l$ is added to a single vertex. We noticed that for small amounts of noise there is a linear correlation between $E(\theta)$ and l/h which allows the development of simple heuristic methods for estimating $E(\theta)$ when noise is added to all three vertices of the triangle. As an application, we computed optimal levels for dithered quantizations of triangle meshes when a tolerance for $E(\theta)$ is given and showed how it can be used in a high-capacity data hiding algorithm.

In the future, we plan to study the relationship between spatial noise and the accuracy of the discrete curvatures of a mesh. The aim, again, is to develop methods for estimating mesh quality when a dithered quantiser is applied to the vertex coordinates.

References

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

[2] P. Alliez and M. Desbrun, "Progressive Compression for Lossless Transmission of Triangle Meshes," *Proc. ACM SIGGRAPH,* pp. 195-202, 2001.

[3] B. Walter, S. Zhao, N. Holzschuch, and K. Bala, "Single Scattering in Refractive Media with Triangle Mesh Boundaries," *Proc. ACM SIGGRAPH,* pp. 1-8, 2009.

[4] S. Jin, R. Lewis, and D. West, "A Comparison of Algorithms for Vertex Normal Computation," *The Visual Computer,* vol. 21, nos. 1/2, pp. 71-82, 2005.

[5] Q. Meyer, J. Susharpmuth, G. Susharpner, M. Stamminger, and G. Greiner, "On Floating-Point Normal Vectors," *Computer Graphics Forum,* vol. 29, pp. 1405-1409, June 2010.

[6] M. Pauly, N. Mitra, and L. Guibas, "Uncertainty and Variability in Point Cloud Surface Data," *Proc. Symp. Point-Based Graphics (SoPBG),* pp. 77-84, 2004.

[7] A. Kalaiah and A. Varshney, "Statistical Geometry Representation for Efficient Transmission and Rendering," *ACM Trans. Graphics,* vol. 24, no. 2, pp. 348-373, 2005.

[8] M. Yoon, I. Ivrissimtzis, and S. Lee, "Variational Bayesian Noise Estimation of Point Sets," *Computers and Graphics,* vol. 33, no. 3, pp. 226-234, 2009.

[9] X. Sun, P. Rosin, R. Martin, and F. Langbein, "Noise Analysis and Synthesis for 3D Laser Depth Scanners," *Graphical Models,* vol. 71, no. 2, pp. 34-48, 2009.

[10] O. Sorkine, D. Cohen-Or, and S. Toledo, "High-Pass Quantization for Mesh Encoding," *Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing,* pp. 42-51, 2003.

[11] L. Schuchman, "Dither Signals and Their Effect on Quantization Noise," *IEEE Trans. Comm. Technology,* vol. CT-12, no. 4, pp. 162-165, Dec. 1964.

[12] R. Gray and T. Stockham, "Dithered Quantizers," *IEEE Trans. Information Theory,* vol. 39, no. 3, pp. 805-812, May 1993.

[13] C. Roads, *The Computer Music Tutorial.* The MIT Press, 1996.

[14] L. Akarun, Y. Yardunci, and A. Cetin, "Adaptive Methods for Dithering Color Images," *IEEE Trans. Image Processing,* vol. 6, no. 7, pp. 950-955, July 1997.

[15] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel, "Interactive Multi-Resolution Modeling on Arbitrary Meshes," *Proc. ACM SIGGRAPH,* pp. 105-114, 1998.

[16] I. Guskov, W. Sweldens, and P. Schröder, "Multiresolution Signal Processing for Meshes," *Proc. ACM SIGGRAPH,* pp. 325-334, 1999.

[17] H. Hoppe, "View-Dependent Refinement of Progressive Meshes," *Proc. ACM SIGGRAPH,* pp. 189-198, 1997.

[18] W. Pasman and F.W. Jansen, "Scheduling Level of Detail with Guaranteed Quality and Cost," *Proc. Seventh Int'l Conf. 3D Web Technology (Web3D '02),* pp. 43-51, 2002.

[19] L. Hu, P.V. Sander, and H. Hoppe, "Parallel View-Dependent Refinement of Progressive Meshes," *Proc. Symp. Interactive 3D Graphics and Games (I3D '09),* pp. 169-176, 2009.

[20] M.-W. Chao, C.-H. Lin, C.-W. Yu, and T.-Y. Lee, "A High Capacity 3D Steganography Algorithm," *IEEE Trans. Visualization and Computer Graphics,* vol. 15, no. 2, pp. 274-284, Mar./Apr. 2009.

[21] F. Cayre and B. Macq, "Data Hiding on 3-d Triangle Meshes," *IEEE Trans. Signal Processing,* vol. 51, no. 4, pp. 939-949, Apr. 2003.

[22] C.-M. Wang and Y.-M. Cheng, "An Efficient Information Hiding Algorithm for Polygon Models," *Computer Graphics Forum,* vol. 24, pp. 591-600, 2005.

[23] Y.-M. Cheng and C.-M. Wang, "An Adaptive Steganographic Algorithm for 3d Polygonal Meshes," *The Visual Computer,* vol. 23, no. 9, pp. 721-732, 2007.

[24] A. Bogomjakov, C. Gotsman, and M. Isenburg, "Distortion-Free Steganography for Polygonal Meshes," *Computer Graphics Forum,* vol. 27, no. 2, pp. 637-642, 2008.

[25] S.-C. Tu, W.-K. Tai, M. Isenburg, and C.-C. Chang, "An Improved Data Hiding Approach for Polygon Meshes," *Visual Computer,* vol. 26, pp. 1177-1181, Sept. 2010.

[26] S. Zafeiriou, A. Tefas, and I. Pitas, "Blind Robust Watermarking Schemes for Copyright Protection of 3d Mesh Objects," *IEEE Trans. Visualization and Computer Graphics,* vol. 11, no. 5, pp. 596-607, Sept./Oct. 2005.

[27] K. Wang, F. Dennis, G. Lavoue, and A. Baskurt, "Hierarchical Watermarking of Semiregular Meshes Based on Wavelet Transform," *IEEE Trans. Information Forensics and Security,* vol. 3, no. 4, pp. 620-634, Dec. 2008.

[28] Y. Yang and I. Ivrissimtzis, "Polygonal Mesh Watermarking Using Laplacian Coordinates," *Computer Graphics Forum,* vol. 29, no. 5, pp. 1585-1593, 2010.

Ying Yang received the BE degree in information security and the ME degree in computer science and technology from the School of Computer and Communication, Hunan University, China, in 2006 and 2009, respectively. Currently, he is working toward the PhD degree at the School of Engineering and Computing Sciences, Durham University, United Kingdom, where his studies are funded by the Durham University Doctoral Fellowship. His research interests include digital watermarking, steganography, steganalysis, and shape analysis.

Norbert Peyerimhoff is a reader at the Department of Mathematical Sciences at Durham University, United Kingdom. His research interests include discrete geometry, combinatorial curvature, and spectral graph theory.

Ioannis Ivrissimtzis is a lecturer at the School of Engineering and Computing Sciences at Durham University, United Kingdom. His research interests include subdivision surfaces, surface reconstruction from unorganized point sets, and machine learning applications in computer graphics problems.