— Supplementary Material —
Visual Contrast Sensitivity and Discrimination for 3D Meshes and their Applications

Georges Nader, Kai Wang, Franck Hétroy-Wheeler, and Florent Dupont

In this Supplementary Material, we first provide in Section 1 some details of the proposed contrast estimation algorithm. In Section 2 we add some details about the properties of the human visual system. Then in Section 3 a simple comparison with Daly's model is given, to show that directly applying Daly's model in the case of 3D meshes does not lead to reasonable results. We present in Section 4 some complementary results of the Just Noticeable Distortion (JND) profile, including JND maps under different circumstances and noised models obtained by different modulations considered in the subjective validation, a comparison with the JND threshold computed in [NWHWD16], a comparison with image-based methods, some details about the computation time, and a discussion about the vertex displacement threshold for interactive scenes. In Section 5 we present some practical information about our implementation of the proposed adaptive mesh subdivision method with a discussion about its integration in real-time applications. In Section 6 we present details and results of another application of our perceptual model, *i.e.*, how the proposed JND algorithm can be used to guide mesh simplification. Finally, in Section 7 we include some additional details regarding the experimental protocol used to measure the contrast threshold.

1 Contrast Estimation for Smooth-Shaded 3D Meshes

We provide the details on how the optimization problem in Eq. (4) of the paper can be solved. First we rewrite Eq. (3) as:

$$d_i(\alpha,\beta)^2 = ||\alpha\vec{A} + \beta\vec{B} + \vec{C}||^2 \ , \tag{S1}$$

where $\vec{A} = \widehat{\vec{n}_3\vec{n}_1}$, $\vec{B} = \widehat{\vec{n}_3\vec{n}_2}$, and $\vec{C} = \widehat{\vec{L}\vec{n}_3}$. The reason behind changing the notation of Eq. (3) is to be able to solve Eq. (4) for Eqs. (3) and (6) in a similar way. For Eq. (6) we will have $\vec{A} = \widehat{\vec{n}_3\vec{n}_1} - \widehat{\vec{l}_3\vec{l}_1}$, $\vec{B} = \widehat{\vec{n}_3\vec{n}_2} - \widehat{\vec{l}_3\vec{l}_2}$, and $\vec{C} = \widehat{\vec{l}_3\vec{n}_3}$.
By developing Eq. (S1) we obtain:

$$d_i(\alpha,\beta)^2 = \alpha^2\vec{A}.\vec{A} + \beta^2\vec{B}.\vec{B} + 2\alpha\beta\vec{A}.\vec{B} + 2\alpha\vec{C}.\vec{A} + 2\beta\vec{C}.\vec{B} + \vec{C}.\vec{C}. \tag{S2}$$

Equation (S2) represents a paraboloid, so solving Eq. (4) boils down to finding the minimum and maximum points on that paraboloid such that $\alpha + \beta \leq 1$ and $\alpha, \beta \in [0,1]$. The minimum point is computed by:

$$\alpha = \frac{\vec{C}.\vec{B}\cdot\vec{A}.\vec{B} - \vec{C}.\vec{A}\cdot\vec{B}.\vec{B}}{\vec{B}.\vec{B}\cdot\vec{A}.\vec{A} - \vec{A}.\vec{B}\cdot\vec{A}.\vec{B}} \ , \beta = -\frac{\vec{C}.\vec{B} + \alpha\cdot\vec{A}.\vec{B}}{\vec{B}.\vec{B}}. \tag{S3}$$

If the computed α and β do not respect the minimization constraints, then theirs values are adjusted accordingly:

$$(\alpha,\ \beta) = \begin{cases} \left(0,\ \min\left(-\frac{\vec{C}.\vec{B}}{\vec{B}.\vec{B}},\ 0\right)\right) & \text{if } \alpha < 0 \\ \left(\min\left(-\frac{\vec{C}.\vec{A}}{\vec{A}.\vec{A}},\ 0\right),\ 0\right) & \text{if } \beta < 0 \\ \left(\min\left(\frac{\vec{C}.\vec{B}\cdot\vec{A}.\vec{B} - \vec{C}.\vec{A}\cdot\vec{B}.\vec{B}}{\vec{B}.\vec{B}\cdot\vec{A}.\vec{A} - \vec{A}.\vec{B}\cdot\vec{A}.\vec{B}}, 1\right),\ 1-\alpha\right) & \text{if } \alpha+\beta > 1 \end{cases}$$

It is easy to show that the maximum distance will always correspond to $(\alpha,\beta) = (0,1),\ (1,0)$ or $(0,0)$.
Having computed α and β it is simple to compute the position or normal of the corresponding point using the barycentric coordinates $[\alpha, \beta, 1-\alpha-\beta]$.

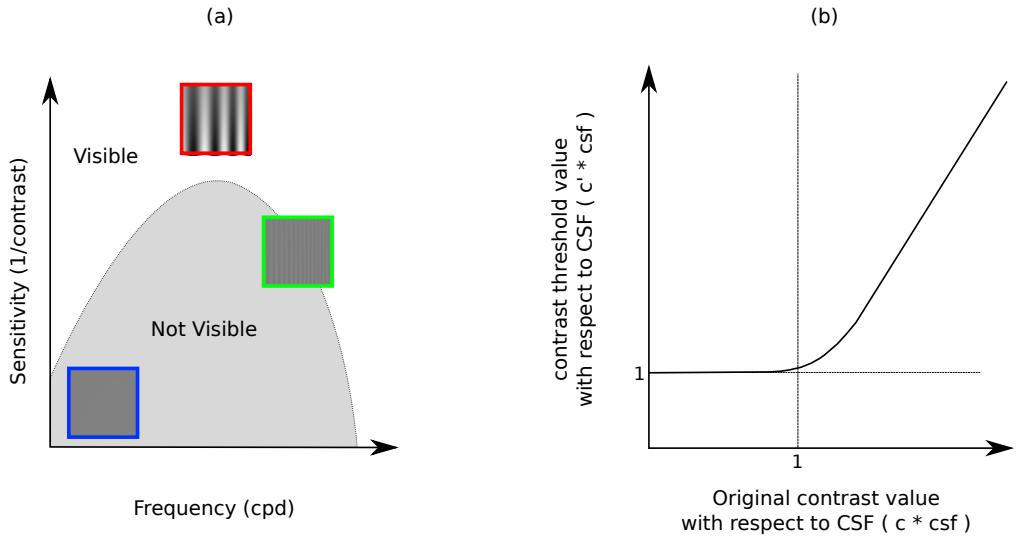


Figure 1: Figures illustrating the main properties of the human visual system: A typical contrast sensitivity function with sinusoidal gratings as the stimuli (a) and the mathematical model describing the general behavior of the contrast discrimination capability between two contrasts (b).

2 Perceptual Properties of the Human Visual System

In this section we present some additional figures to help the reader better understand the corresponding presentation about the properties of the human visual system in the paper. Figure 1.(a) shows an example of the contrast sensitivity function (CSF) which represents the pass-band characteristic of the human visual system. As the frequency of the visual stimulus increases, the visual system becomes more sensitive to the change in contrast contained in the visual pattern until reaching a certain value beyond which the sensitivity starts to drop again. Therefore, for a certain frequency range (usually between 2 and 5 cpd), the visual system is very sensitive, being able to observe even very small contrast value change in the visual stimulus. Figure 1.(b) shows a curve that mathematically describes the capability of the human visual system to discriminate between two contrasts. When the original contrast, c, is not visible (*i.e.*, $c * \text{csf} < 1$), then the contrast threshold, c', is the one given by the CSF. As the original contrast value increases beyond its visibility threshold, it becomes harder for the human visual system to distinguish a change in the visible contrast. This is reflected by an increase in threshold, approximately in a linear manner. The slope of this linear part mainly depends on the nature, *i.e.*, the visual regularity, of the visual stimulus.

3 Comparison with Daly's Model

The difference in the type of visual artifacts caused by a geometric operation on 3D mesh in comparison to the ones caused by an image processing operation makes the straightforward usage of traditional perceptual models developed for 2D images incompatible in the case of 3D mesh processing. This is clearly shown in the results of our psychophysical experiments as the measured threshold value is different from the one as given by Daly's model. Figure 2 gives further verification. The computed vertex displacement threshold on the Max Planck mesh is lower when using Daly's model compared to our model which is derived from a series of specifically designed psychophysical experiments on 3D meshes. In addition, injecting a JND modulated noise onto the Max Planck mesh yields in both cases a distorted mesh with invisible noise. However, the distorted mesh obtained using our perceptual model has a much higher MRMS. This shows that Daly's model under-estimates visibility threshold in the case of 3D mesh processing. This is again probably due to the fact that the induced visual artifacts, *i.e.*, the considered visual stimuli, are quite different in the two models. Discussions about this point can also be found in the paper, in the paragraph just below Eq. (9).

4 Just Noticeable Distortion Profile

4.1 Vertex Displacement Maps

Figure 3 shows the vertex displacement threshold (JND profile) of several 3D models. Figure 3.(a) illustrates how the perceptual model can adapt to the mesh density as the displacement threshold is higher for the

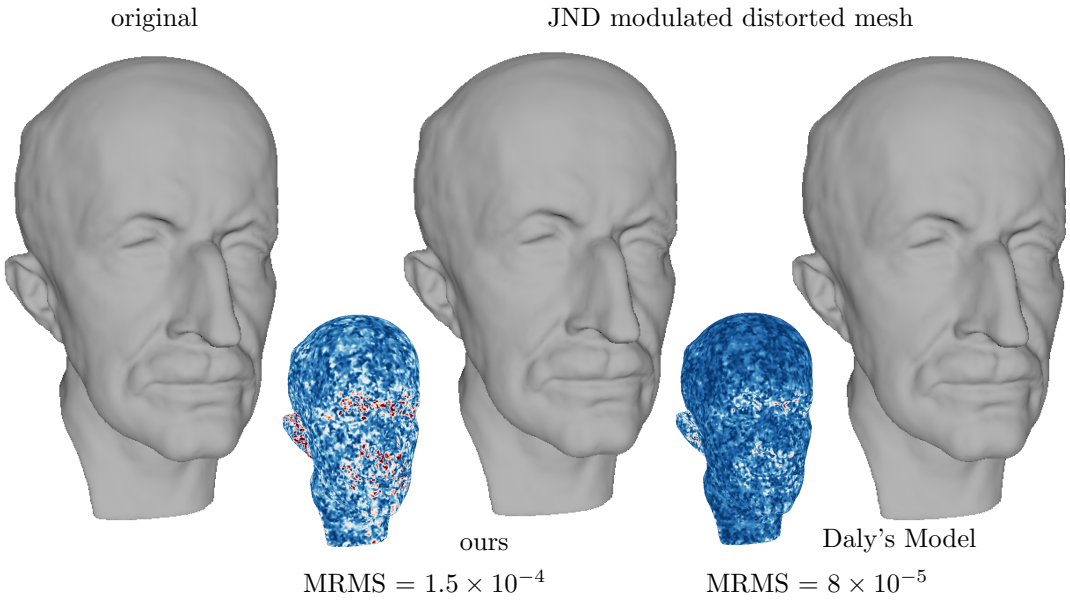


Figure 2: Comparison between the vertex displacement threshold obtained using our model and Daly's model, as well as the corresponding JND modulated noised meshes.

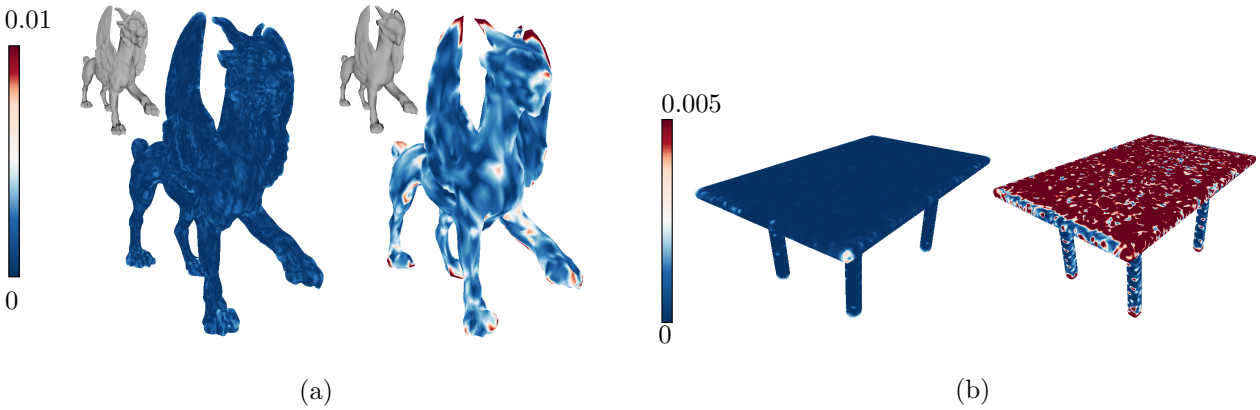
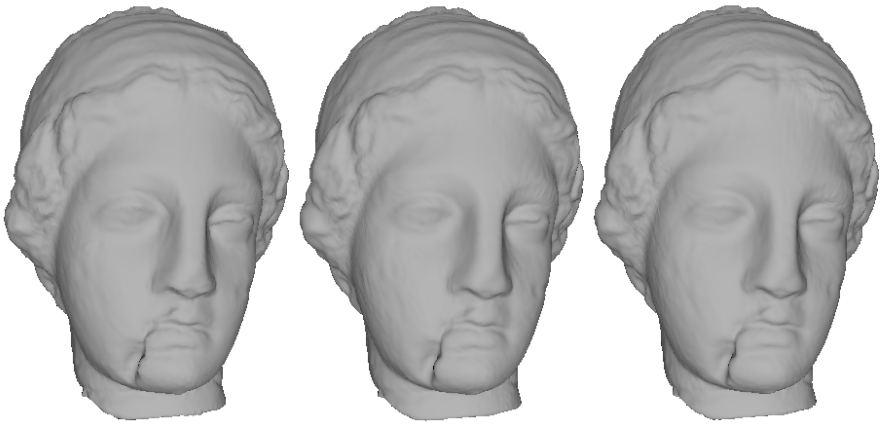


Figure 3: (a) The vertex displacement threshold map for a 3D model of two different detail resolutions: a high-resolution model (left) and a low-resolution one (right). (b) The vertex displacement threshold map for a 3D model with respect to a displacement in the normal (left) and tangent (right) direction.

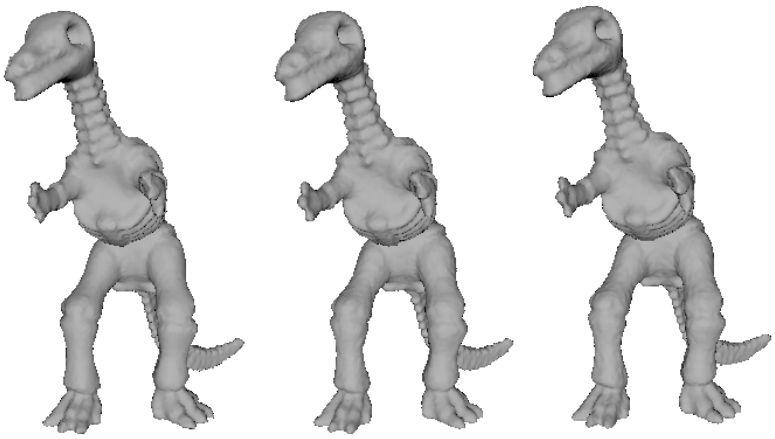
low-resolution model than the detailed one. Figure 3.(b) demonstrates that the algorithm for computing the displacement threshold takes into account the direction of vertex displacement as the displacement in the normal direction on a flat surface is barely tolerable while the displacement in the tangent direction can be tolerated.

4.2 Distorted Meshes with Different Noise Modulations

Figure 4 displays three distorted versions of the Venus and Dino models. The three Venus models have an almost identical MRMS value of about 1.7×10^{-4} and the three Dino models have an MRMS value of about 4.8×10^{-4}. The distorted models have been obtained using the modulations described in Section 5.1 of the paper. The left version of both the Venus and Dino models is injected with a noise proportional to the vertex displacement threshold with $\beta_{\text{jnd}} = 1$ while the middle and the right versions are respectively injected with a noise proportional to the mesh roughness and with a uniform noise. While the injected noise is visible on the middle and right models (especially on forehead of Venus and thigh of Dino), it is very difficult to notice it on the left model. This again demonstrates that the proposed perceptual model is able to add the largest amount of invisible noise onto the mesh surface.



(a)



(b)

Figure 4: Distorted models of the (a) Venus and (b) Dino meshes with almost the same MRMS value. The left mesh corresponds to a JND modulated noise, the middle one corresponds to a roughness modulated noise and the right one corresponds to a uniform noise.

4.3 Comparison with [NWHWD16]

In [NWHWD16], the authors presented a method for computing the local contrast in the case of a flat shading rendering algorithm. Therefore, the resulting vertex displacement threshold is not compatible for the case when the model is rendered using a smooth shading algorithm. In this paper, we have presented an analytical approach that allows us to compute the local contrast in a triangular face that is compatible with a smooth shaded rendering. Figure 5 presents side by side the JND profile for the Bunny model in a smooth shading mode and a flat shading mode [NWHWD16]. In general we have observed that the displacement threshold relative to a smooth shading rendering is 5 to 10 times bigger than the one relative to a flat shading mode. This difference is due to the way how the surface normals and contrast are computed in each mode. In [NWHWD16] (flat shading mode), the contrast is computed using two the normals of a pair of faces, while in this paper the contrast is evaluated using the vertex normals within a triangular face which reflects a smooth shading rendering. In general, the displacement of a vertex causes a bigger rotation in the normal direction of the faces adjacent to the displaced vertex in a flat shaded rendering, compared to the rotation of normal direction of the vertices in the 1-ring neighborhood computed in a smooth shaded rendering. In consequence, for the same displacement magnitude, the change in contrast is bigger in a flat shading mode compared to a smooth shading mode, which explains the lower visibility threshold. Moreover, in [NWHWD16], the perceptual model only works with directional light and does not take into account neither the global luminance nor the visual regularity when computing the displacement threshold.

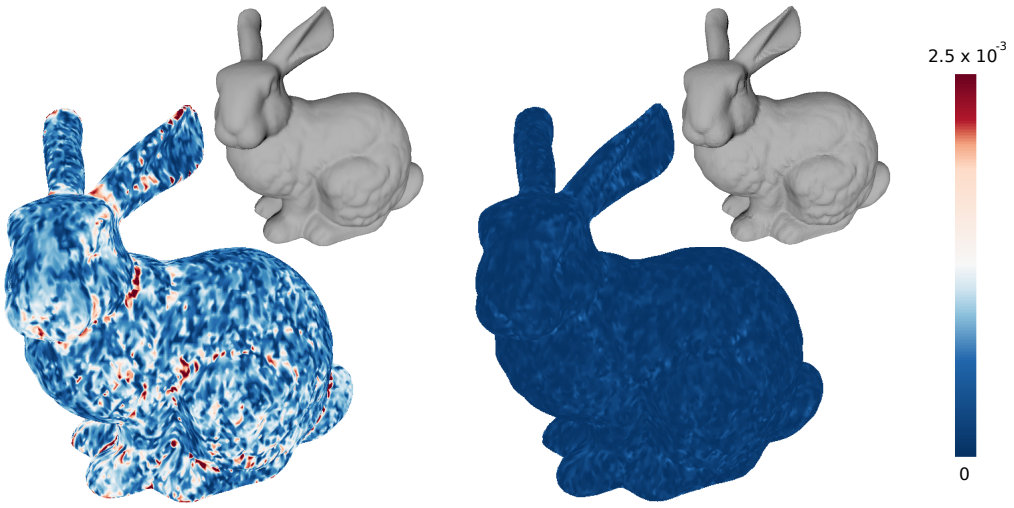


Figure 5: The vertex displacement profile for the Bunny model computed using contrast estimation method tailored for a smooth shaded rendering and a flat shaded one [NWHWD16].

4.4 Comparison with an Image-Based Method

In this section we compare the proposed vertex displacement threshold method to the HDRVDP2 metric [MKRH11], an image-based method. HDRVDP2 is a popular perceptual metric that extends Daly's VDP to HDR images and its code is freely available at `http://hdrvdp.sourceforge.net/wiki/`.

The first comparison consists of using the HDRVDP2 metric to detect the optimal quantization levels of the models presented in Section 5.2 of the paper. To do so we have adopted the same technique used in [LLV16]. We have rendered the models from 64 viewpoints located on a sphere around the object. For each viewpoint we compare the reference model with the distorted one using HDRVPD2. Finally we aggregate the score from each viewpoint using a Minkowski pooling to obtain a final score indicative of the perceptual distance between the reference and the distorted model.

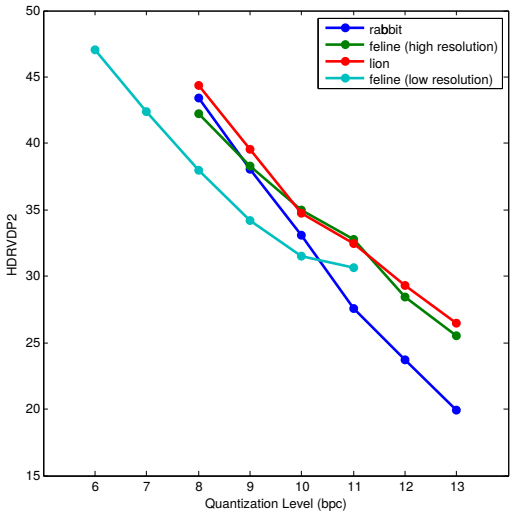


Figure 6: The HDRVDP2 score versus the quantization levels of four models.

Figure 6 shows that it would not be possible, by using HDRVDP2, to obtain the same optimal quantization levels as the ones obtained with our method (Fig. 10 of the paper). The problem lies mainly with the Lion model whose HDRVDP2 scores seem to be exaggerated. A further investigation of the Lion model shows that this exaggeration of the quantization distortion visibility comes from the nature of the model's surface. HDRVDP2 appears to be exaggerating the visibility of the distortions in the rough parts of the Lion's surface.

To illustrate this point, we present in Fig. 7 the results of a second comparison with HDRVDP2, *i.e.*,

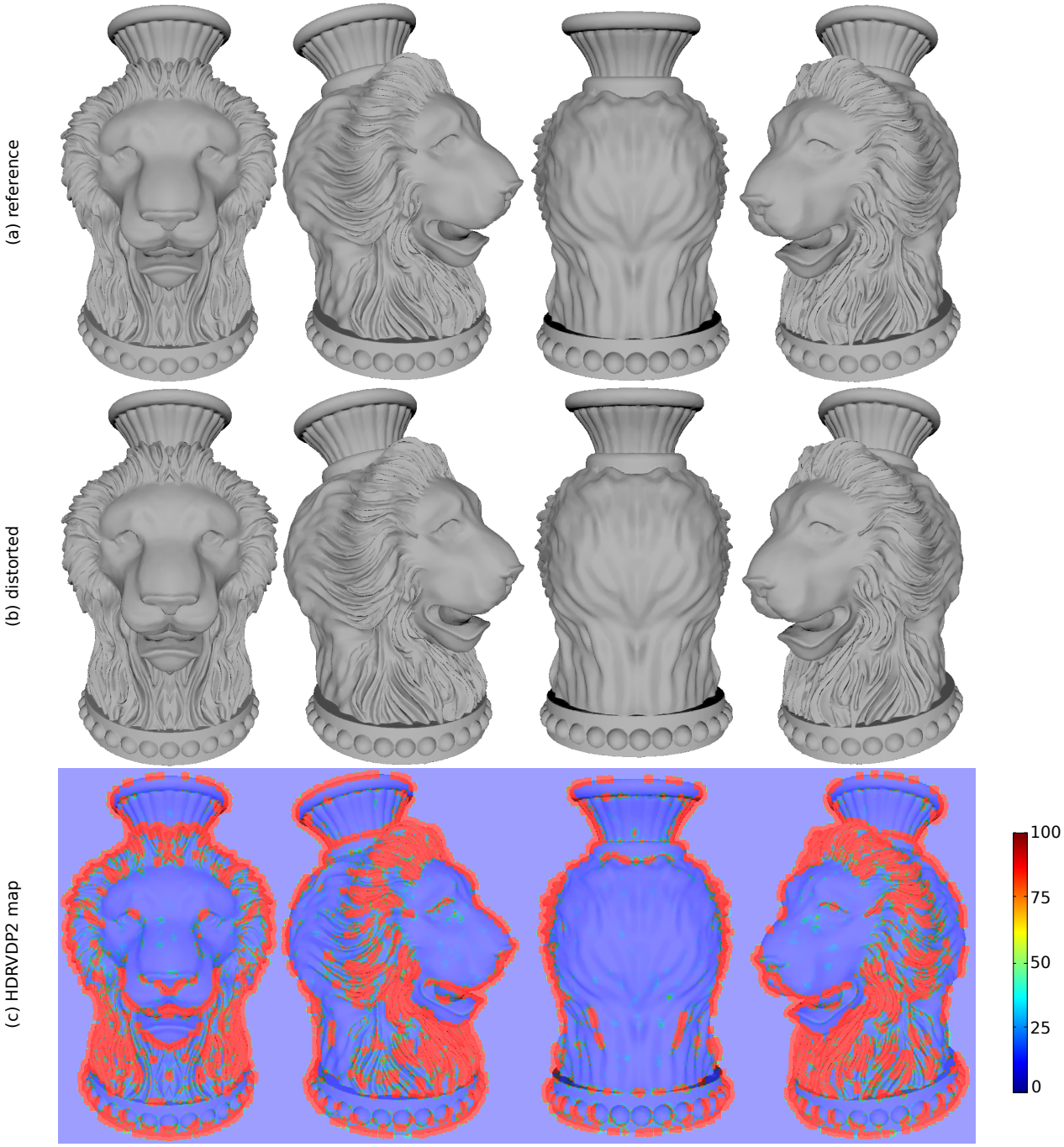


Figure 7: The HDRVDP2 visibility map for a distorted Lion model whose noise is below the JND threshold.

the visibility map given by the HDRVPD2 method when comparing images of a reference Lion model with a distorted one from various viewpoints. The distorted model was obtained by injecting in Lion a vertex displacement noise that is below the threshold given by our method. To be more precise, the noise was injected according to the method described in Section 5.1 of the paper with $\beta_{jnd} = 0.85$. Therefore, according to our method the injected noise should be invisible which is the case if we visually compare the first two rows of Fig. 7. However, when looking at the visibility map given by the HDRVDP2 method we notice that the distortion visibility is particularly exaggerated in the rough regions of the Lion model. This exaggeration of distortion visibility on surface regions with high roughness and curvature can be attributed to a possible limitation of image-based approaches for estimating the visibility. A local geometric distortion will change the geometry of the object which will be reflected in slight change of pixel position in addition to the change in contrast. This change in pixel position is considerable when the surface is curved because it can cause a local displacement of pixels without changing the global shape. Since in image-based approaches, the perceptual analysis is in general done in a per-pixel basis, then these local pixel displacements caused by the geometric distortions are

detected to be of high amplitude, leading to an exaggeration in visibility.

Additionally, Fig. 6 shows that the perceptual score given by the HDRVDP2 metric is also exaggerated for low-resolution meshes indicating that it was not able to accurately adapt to the change in mesh density. This is probably due to the fixed pixel window width used to perform the perceptual analysis, suggesting that the visibility of all distortions are evaluated regarding almost the same spatial frequency. However, this is not necessarily the case in a 3D mesh whose density might be variable and thus leading to the presence of visual stimuli of different spatial frequencies. Finally, image-based methods such as HDRVDP2 have an intrinsic limitation compared to our method, *i.e.*, for image-based methods it is necessary to generate the rendered 2D image before applying any perceptual analysis. This will ultimately make integrating these methods in geometric operations a more complicated task and such methods also mix geometric distortions with rendering noise in the perceptual analysis. Therefore we think that image-based methods are more suited for evaluating and/or guiding rendering algorithms rather than geometric operations.

4.5 Execution Time

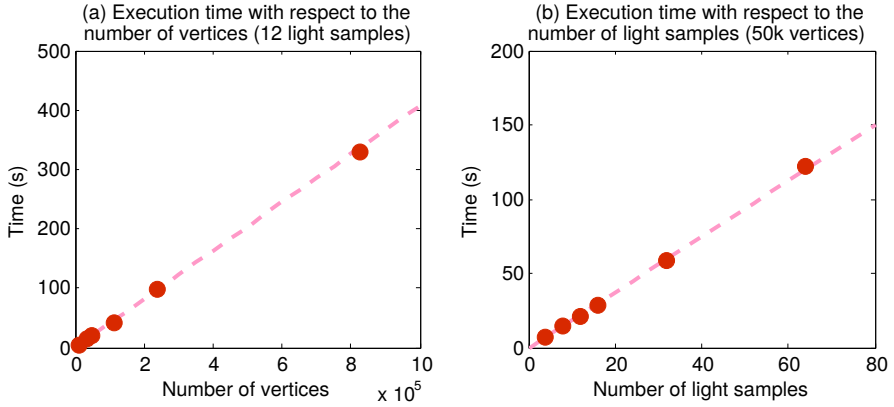


Figure 8: Vertex displacement threshold execution time.

In a light-independent mode, we compute the vertex displacement threshold relative to several light directions sampled from a local hemi-sphere around a vertex. A quick theoretical analysis shows that the complexity of the algorithm for computing the vertex displacement threshold is linear with respect to the number of light samples and to the number of vertices of the 3D mesh. Figure 8 also shows the execution time required to compute the vertex displacement threshold on an entire mesh with respect to the number of vertices and light samples.

4.6 Discussion about the Vertex Displacement Threshold for Interactive Scenes

In an interactive scene, the user can manipulate the mesh model displayed on the screen. Therefore, the light direction and position of the model are susceptible to be changed over the course of the viewing session. Changing the light direction will result in a change of local contrast. On the other hand, changing the position and/or the scale of the model can lead to either a change in local contrast or a change in local spatial frequency. If the light is fixed relative to the view point and the model is rotated or translated with regard to the X and/or Y axis of the view frame (*i.e.*, distance from camera is not changed), then this will cause a change in the angles between the surface normal and the light direction, which is the same as changing the light direction and therefore causes a change in contrast. If the model is scaled or translated with regard to the Z axis of the view frame (*i.e.*, distance from camera is changed), then this will cause a change in spatial frequency as the perceived size of the visual stimulus is altered. In all of those cases, a varying local contrast and local spatial frequency will cause a change in displacement threshold.

First, in order to account for the change in contrast which is caused by changing the light direction with respect to the surface normal (fixing the light and changing the model's position or fixing the model and changing the light) we compute the JND profile in the light independent mode. As mentioned in the paper, this is done by computing the displacement threshold from a number of light directions sampled from a hemi-sphere around the local vertex and then choosing the lowest displacement value. The idea here is that the light independent threshold corresponds to the one relative to the worst possible light condition. A practical discussion about the efficient sampling of the hemi-sphere can be found in the Supplementary Material of

Figure 9: A vertex noise at the JND level computed in a light dependent mode might become visible when the light direction is altered. This is not the case for JND computed under light independent mode.

[NWHWD16]. Figure 9 shows the Venus model injected with a vertex noise at the JND level computed with a light dependent and light independent mode. Notice how, when the light is changed, the noise relative to the light independent threshold remains invisible while the one relative to the light dependent mode becomes visible.

Second, if in the interactive session the user is allowed to change the distance between the view point and the object by either changing the view distance or scaling the model, then the perceived spatial frequency is affected. Similarly to the first case, computing a threshold that works with any distance boils down to computing a threshold that works with the worst distance. Since a change in distance (or scale) affects only spatial frequency then computing the JND relative to the worst possible frequency would result in a threshold that is adaptive to various distances (or scales). This worst possible frequency is the one corresponding to the peak of the CSF curve since it is by definition the most sensitive frequency (*i.e.*, the frequency with the lowest visibility threshold). According to our experimental study we have found that the CSF peaks at around 3.5 cpd. Figure 10 shows the Venus model injected with a vertex noise at the JND threshold level computed according to the most sensitive frequency and that of a JND under a certain fixed distance. The Venus model is a dense mesh, therefore the frequencies relative to the first row of Fig. 10 are high ($\approx$10 cpd). As the camera approaches the model, the size of the visual stimuli becomes bigger and therefore the frequency decreases. This means that as the model becomes closer to the camera, the frequency becomes more sensitive (according to the CSF) and thus should reduce the value of the visibility threshold. Indeed, when injecting in the model a vertex noise relative to a JND computed according to the original distance, it becomes visible when zooming in on the model (see Fig. 10.(c)). On the other hand, when computing the JND according to the most sensitive frequency, the noise remains invisible regardless of the distance since it has already been taken into account as the worst possible distance (see Fig. 10.(b)).

It is important to note that computing the threshold with regard to a certain distance or light has also its own benefits. For instance, having a threshold relative to a view distance allows us to generate a LOD for a certain model as described in Section 6 of this Supplementary Material. The change in threshold caused by the change of distance will stop the edge collapse operation at different degrees of simplification and thus

(a) reference

(b) vertex noise relative to a JND computed according to the most sensitive frequency

(c) vertex noise relative to a distance dependent JND

Figure 10: A vertex noise at the JND threshold level computed according to the most sensitive frequency (b) and a certain distant distance as in the first row (c).

creating an adaptive LOD.

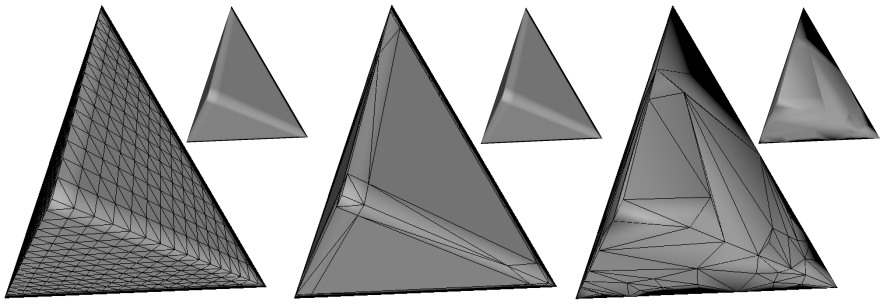
5 Adaptive Mesh Subdivision

We have implemented the proposed adaptive subdivision prototype on the CPU. Subdividing a 3D model requires two steps. The first step consists of creating the new faces when the perceptual criterion judges that a certain face should be subdivided. The second step involves applying a re-meshing operation whose goal is to clean up the mesh topology so that the subdivided mesh can be rendered properly. The execution of this adaptive subdivision method is fast. For instance it took about 9ms to subdivide the Bimba model presented in Fig. 13.(a) of the paper with approximately 7.5ms required to apply the first step and 1.5ms to perform the re-meshing operation. In an interactive setting running at 30 frames per second, each frame have a computation budget of 33ms. So we think that it would be possible, with a CPU based implementation, to obtain a real-time performance with the proposed adaptive subdivision method (if the application at hand is not heavy on CPU usage). In addition, the proposed subdivision criterion requires the information present within a single face (*i.e.*, the 3 vertex normals) in order to compute the contrast, we think it would be possible to implement this method on the GPU so that it can be used in more demanding real-time applications.

6 Perceptually Driven Mesh Simplification

Due to page limit of the paper, we present in this Supplementary Material the details and results of the mesh simplification application of the proposed perceptual model and JND profile.

Mesh simplification algorithms have been the target of many perceptual analysis methods throughout the literature. Similarly to [NWHWD16], we show how the vertex displacement threshold can help guide the simplification process. More specifically, the computed vertex displacement threshold is the most beneficial for controlling the visual outcome of the simplification algorithm as it allows us to naturally define an automatic stopping criterion which ensures that the resulting mesh is visually similar to the original model. This is particularly important when it comes to generating multiple levels of details (LOD) of a highly detailed



original (a) (b)

Figure 11: (a) The perceptually driven simplification is able to prevent perceptually relevant edges from being collapsed as it outputs a model that is visually similar to the original one. (b) The output model using the method of Lindstrom and Turk [LT98] to the same number of vertices is visually different from the original mesh.

original - 50k 23k 14k

Figure 12: LODs generated using the perceptually driven simplification method at different viewing distances. The degree of simplification automatically adapts to the viewing distance.

model.

A popular class of mesh simplification algorithms consists of iteratively collapsing the edges of a 3D mesh. The visual quality of the simplified model will heavily depend on the order by which the edges are collapsed. Ideally, the best visual quality can be obtained if the edge collapse operation is carried out starting from the one with the least visual impact until the one with the most perceptual influence. This is exactly what we can do with the information provided by the vertex displacement JND profile.
Let $e = \{\vec{v}_1, \vec{v}_2\}$ be an edge of a 3D mesh. By analyzing the visibility of displacing $\vec{v}_1$ and $\vec{v}_2$ along e, we can estimate the perceptual influence of collapsing the edge e. Similarly to [NWHWD16], the impact of collapsing an edge e can be evaluated by:

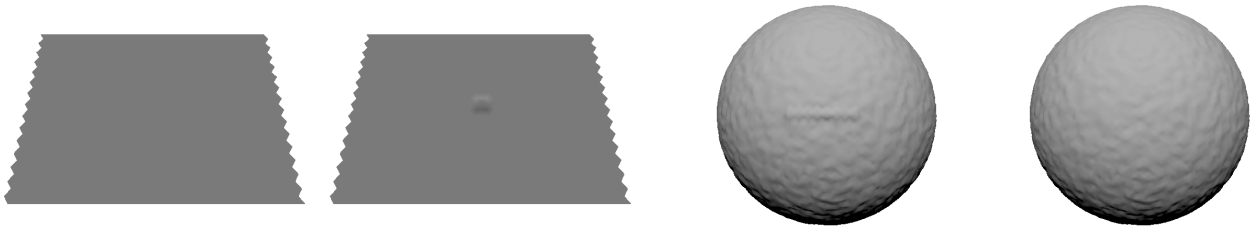
$$I = \frac{(\vec{v}_2 - \vec{v}_1).(\vec{v}_{\text{jnd}_2} - \vec{v}_{\text{jnd}_1})}{||e||^2} , \tag{S4}$$

where $\vec{v}_{\text{jnd}_1}$ and $\vec{v}_{\text{jnd}_2}$ are respectively the vertices obtained by displacing $\vec{v}_1$ (resp. $\vec{v}_2$) by exactly the displacement threshold value in the direction of $\vec{v}_1\vec{v}_2$ (resp. $\vec{v}_2\vec{v}_1$) and $||e||$ is the length of the corresponding edge. A negative value of the perceptual impact indicates that the collapse operation will remain unnoticed by a human observer while a positive value means that the collapse will be visible (refer to [NWHWD16] for more details). So if we collapse the edges according to an order of increasing perceptual impact, then we can control the visual quality of the output mesh. Intuitively, by stopping the simplification process when all the edges have a positive perceptual impact we obtain a most simplified mesh that is visually similar to the detailed input. For instance, in Fig. 11 we show a simplified tetrahedron obtained using the proposed perceptual

simplification method. Notice that the simplification process did not collapse the edges that are adjacent to the hard edges of the tetrahedron as they have a big influence on the interpolation of luminance and thus they have a high perceptual impact. This is however not the case for traditional simplification methods such as [LT98] which focus on preserving the geometry of the object without taking into consideration the shading.

Additionally, since the proposed perceptual model is capable of adapting to the various parameters of the display (size, resolution and brightness) and to the distance between the model and the viewpoint, then the perceptual impact of an edge will adapt as well. In consequence the degree of simplification applied to a detailed mesh will depend on those parameters. This is important as it will be possible to generate multiple LODs by simply specifying the distances between the 3D model and the viewpoint (see Fig. 12).

7 Experimental Protocol



(a) plane stimulus (b) sphere stimulus

Figure 13: The visual stimuli as presented to the subjects during the threshold measurement experiments.

Our psychophysical experiments consist in measuring the visibility threshold relative to the effects of contrast sensitivity and discrimination in a smooth shading rendering. To do so, we have proposed an experimental protocol as follows. We have designed the task given to the participants to be as precise and efficient as possible at the same time. The efficiency of the task is an important criterion since long experiments will be both time consuming and more importantly tiring for the participants which might affect their performance. Hence, we have used the QUEST method to adjust the intensity of the stimulus after each trial and designed the task as follows. We have displayed two objects side by side on the screen, one of which exhibits one displaced vertex (or a series of displaced vertices) in its central area (Fig. 13). The task given to the observers was a slightly altered *Yes/No* procedure. More precisely, one of the displayed object contains the displaced vertex/vertices and the other object acts as a reference. The subjects were then asked to respond by *Yes* or *No* to the following question: *Are the two objects different?* One of the advantages of a *Yes/No*-type procedure is the fast convergence of QUEST-like method in such a procedure, which leads to shorter experiments and is thus less tiring for inexperienced subjects. Finally, it is worth mentioning that the stimuli were presented to subjects in a full-screen mode and that there was no graphical user interface displayed, as illustrated by Fig. 13. Subjects gave responses to *Yes/No* questions via keyboard.

References

[LLV16] G. Lavoué, M.-C. Larabi, and L. Vasa. On the efficiency of image metrics for evaluating the visual quality of 3D models. *IEEE Transactions on Visualization and Computer Graphics*, 2016.

[LT98] P. Lindstrom and G. Turk. Fast and memory efficient polygonal simplification. In *Proc. of IEEE Visualization Conference*, pages 279–286, 1998.

[MKRH11] R. Mantiuk, K. J. Kim, A. G. Rempel, and W. Heidrich. HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Transactions on Graphics*, 30(4):40:1–40:13, 2011.

[NWHWD16] G. Nader, K. Wang, F. Hétroy-Wheeler, and F. Dupont. Just noticeable distortion profile for flat-shaded 3D mesh surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 2016.