

METRICS FOR IMAGE SURFACE APPROXIMATION BASED ON TRIANGULAR MESHES

EDUARDO SANT'ANA DA SILVA, ANDERSON SANTOS AND HELIO PEDRINI✉

Institute of Computing, University of Campinas, Campinas-SP, Brazil, 13083-852
e-mail: eduardo.santanadasilva@gmail.com, mr.acarlos@gmail.com, helio@ic.unicamp.br
(Received August 2, 2016; revised November 26, 2017; accepted December 3, 2017)

ABSTRACT

Surface approximation plays an important role in several application fields, such as computer-aided design, computer graphics, remote sensing, computer vision, robotics, architecture, and manufacturing. A common problem present in these areas is to develop efficient methods for generating, processing, analyzing, and visualizing large amount of 3D data. Triangular meshes constitute a flexible representation of sampled points that are not regularly distributed in space, such that the model can be adaptively adjusted to the data density. The choice of metrics for building the triangular meshes is crucial to produce high quality models. This paper proposes and evaluates different measures to incrementally refine a Delaunay triangular mesh for image surface approximation until either a certain accuracy is obtained or a maximum number of iterations is achieved. Experiments on several data sets are performed to compare the quality of the resulting meshes.

Keywords: accuracy metrics, point cloud, surface approximation, triangular meshes.

INTRODUCTION

Advances in techniques and equipments for image acquisition have allowed the extraction of large data volumes. Several knowledge domains require the construction of surface models with high level of details, such as computer vision, digital entertainment, cartography, computer graphics, finite element methods, scientific visualization, manufacturing, architecture, remote sensing, and virtual reality.

Despite the reduction of costs with storage devices, the availability of images at higher resolutions demands the development of efficient mechanisms for storing, manipulating, transmitting and visualizing data.

A digital image is commonly represented by a two-dimensional matrix that stores intensity values sampled at a discrete set of points. The matrix grid must have a resolution sufficiently small to capture the details necessary for the application under consideration. In order to provide a variable-resolution approximation of digital images, some hierarchical representations have been developed in the literature, such as quadtrees (Sullivan and Baker, 1994; Scholefield and Dragotti, 2014) and pyramids (Burt and Adelson, 1983; Tran *et al.*, 1987).

In contrast to the use of regular grid representations, where a set of sampled data points is stored at uniform intervals, polygonal meshes (Araújo *et al.*, 2015; Bommès *et al.*, 2013; Botsch *et al.*, 2010; Maglo *et al.*, 2012) can be more adaptive to irregularity of the data. In particular, triangular

meshes are a common structure for modeling surface as a network of adjacent triangles, whose vertices represent the data samples. Important features, such as corners or edges, can be directly incorporated into the model. Furthermore, the triangular structure can be adjusted to fit variable density of data points, avoiding redundancy. In fact, triangular meshes can be used to compress (Maglo *et al.*, 2012) data, while preserving relevant surface characteristics. Such data reduction improves storage requirements, transmission capacities, and rendering performance.

Several methods for constructing triangular meshes from dense data sets have been proposed in the literature (Lipman, 2012; Ma *et al.*, 2013; Moraes *et al.*, 2017; Nivolières *et al.*, 2012). Most surface approximation approaches can be classified as refinement or decimation methods. Refinement methods (Boissonnat *et al.*, 2009; Brown, 1997; Hu *et al.*, 2009; Dey and Ray, 2010) start with a simple initial approximation of the surface and incrementally add new data points to the triangulation until a given approximation criterion is achieved, such as desired accuracy or maximum number of iterations. On the other hand, decimation methods (Cignoni *et al.*, 1998; Shaffer and Garland, 2001; Luebke *et al.*, 2002; Salinas *et al.*, 2015) start with a triangulation containing the entire set of sampled data points and iteratively simplify it until a specified approximation criterion is satisfied.

The vertex selection metric used in the refinement or decimation process is crucial to produce compact and precise triangular meshes, since it determines the

degree of fidelity between the original data and the approximated model. Therefore, adequate measures for determining the relevance of a point must be carefully designed and evaluated.

The main contribution of this paper is to propose and evaluate different metrics for inserting new points to an initial triangulation with accuracy within a specified error tolerance or maximum number of points. More specifically, we extend upon the set of mesh refinement metrics developed in a previous work (van Kaick and Pedrini, 2006), such that five new criteria for image surface approximation are defined and analyzed in this work. Several data sets are used to assess the effectiveness of the metrics.

The paper is organized as follows. Section 2 briefly describes relevant concepts and works related to surface modeling and image approximation. Section 3 presents the different metrics used to approximate a set of sampled data points through triangular meshes. In Section 4, experiments are conducted on several data sets to assess the effectiveness of the evaluated metrics. Section 5 concludes the paper with some final remarks and directions for future research.

BACKGROUND

Different spatial data structures can be employed to model surface from a set of points, however, they have critical effects in terms of storage requirements, degree of accuracy, retrieval flexibility, scalability, among other factors. Examples of spatial data structures (Samet, 1990; 2006), illustrated in Fig. 1, include a grid representation of regularly distributed data points, a triangular mesh, a quadtree and a bintree.

Several surface representations have been proposed in the literature (Araújo *et al.*, 2015; Bommers *et al.*, 2013; Botsch *et al.*, 2010; Maglo *et al.*, 2012; Mostafavian and Adams, 2015; Berger *et al.*, 2013; Garland and Heckbert, 1997), however, polygonal models are probably the most common choice for representing sampled data sets since they are supported by the vast majority of tools for modeling and rendering purposes.

A polygonal surface is a piecewise-linear surface defined by a set of polygons, typically a set of triangles. In addition to their simplicity and flexibility, polygonal models can represent surfaces at different levels of detail (Hu *et al.*, 2009; Luebke *et al.*, 2002; Morigi and Rucci, 2013) and accuracy, such that a coarse representation can be used to describe less relevant areas, while high

resolution can be focused on specific parts of interest. Furthermore, discrete definitions of differential operators have been proposed (Meyer *et al.*, 2003), such as curvature measure (Kim *et al.*, 2002) and geodesic curves (Polthier and Schmies, 2006).

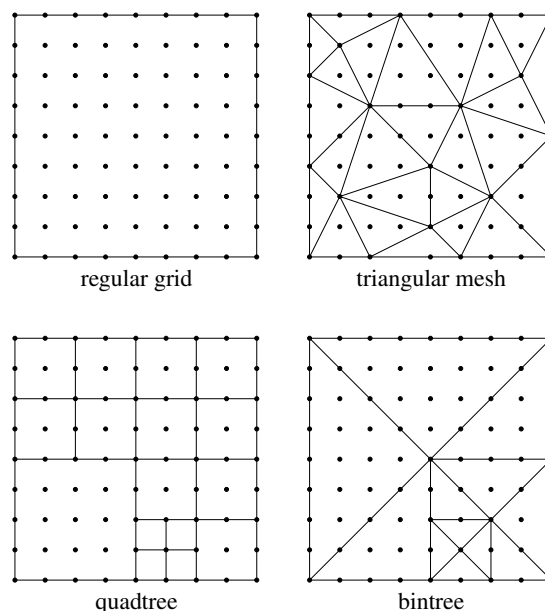


Fig. 1: Examples of spatial data structures.

Techniques for constructing triangle meshes can be generally categorized into refinement and decimation. In the refinement methods (Boissonnat *et al.*, 2009; Brown, 1997; Hu *et al.*, 2009; Dey and Ray, 2010), a new data point is iteratively inserted into the mesh, typically the point that presents the largest approximation error according to a specific metric. In the decimation methods (Cignoni *et al.*, 1998; Shaffer and Garland, 2001; Luebke *et al.*, 2002; Salinas *et al.*, 2015), the point with the smallest approximation error is iteratively removed from the current triangulation. In both approaches, local adjustments must be performed after each point operation in order to rebuild the mesh. The process of refinement or decimation finishes when a required number of points is satisfied or an error tolerance is reached.

Two common strategies for generating and maintaining triangles meshes for image surface approximation purpose are Delaunay triangulation and data-dependent triangulation.

Delaunay triangulation (de Berg *et al.*, 2008; Boissonnat *et al.*, 2015; Cheng *et al.*, 2012) guarantees certain geometric properties. The circumcircle of any triangle in a Delaunay triangulation contains no other data points in its interior. Another interesting property is that the Delaunay triangulation of a set of points is dual to its Voronoi diagram (de Berg *et al.*,

2000; Preparata and Shamos, 1985). The Delaunay triangulation also maximizes the minimum angle of all triangles of the mesh, reducing the occurrence of thin triangles that tend to cause undesirable effects, such as numerical instability and visual artifacts (Scarlato, 1992; Rippa, 1992a). Based on the property of maximizing the minimum angle, Lawson (1977) proposed a local optimization procedure to swap the diagonals of a strictly convex quadrilateral to produce a most equiangular triangulation. An important consequence is that, after a finite number of swaps, the successive application of this procedure to all internal edges of an arbitrary triangulation will produce a Delaunay triangulation.

Data-dependent triangulation (Brown, 1991; Dyn *et al.*, 1990; Quak and Schumaker, 1990) maintains the mesh topology based on information of the input points to approximate the image surface. Although data-dependent triangulation methods can generate compact meshes, long and thin triangles are usually produced, which can lead to certain disadvantages for rendering purpose (Kolingerová *et al.*, 2010; Scarlato, 1992; Rippa, 1992a). However, such approaches can be suitable for the reconstruction of features, such as edges in images, where a function has high second-order derivatives in one direction when compared to other directions (Rippa, 1992b; Tóth *et al.*, 2007; Wang *et al.*, 2001).

In this work, we used the Delaunay triangulation to incrementally approximate image surfaces due to its efficient operations to maintain a valid topology after each vertex insertion, as well as visual quality of the generated meshes.

The development of compact representation schemes for approximating image surface plays an important role in the reduction of storage and transmission costs. Associated with this subject, a large number of lossless and lossy image compression techniques have been proposed in the literature (Held and Marshall, 1996; Rabbani and Jones, 1991), such as transform coding (DeVore *et al.*, 1992; Skodras *et al.*, 2001), deflation (Rigler *et al.*, 2007; Xie *et al.*, 2008), chain codes (Ren *et al.*, 2002; Sánchez-Cruz *et al.*, 2007), chroma subsampling (Chen *et al.*, 2009; Lin *et al.*, 2013), fractal compression (Barnsley and Hurd, 1993; Fisher, 2012), among others.

In the context of image surface approximation, several approaches have been developed to compress triangular meshes (Peng *et al.*, 2005) in order to reduce the size of the resulting models. These methods usually explore two types of information, geometry and topology.

Geometry compression methods (Chow, 1997; Van Aerscht *et al.*, 2009; Khodakovsky *et al.*, 2000) aim to reduce the numerical information associated with the mesh vertices, such as position, intensity, normal vectors, and texture. Deering (1995) introduced the concept of geometry compression, where vertex positions, normals and colors were quantized to less than 16 bits through delta compression and modified Huffman encoding. Taubin and Rossignac (1998) developed a geometry compression technique where vertex positions were quantized through a spanning tree, achieving 12 bits per vertex.

Topology compression methods aim to reduce the connectivity information which associates vertices of the mesh with edges and faces. Hoppe (1996) described a technique for transferring a mesh progressively, from a coarse approximation to a refined sequence of new vertices. Each vertex is transferred once, where 5 bits were used to identify two adjacent vertices. Other progressive compression methods were developed by Alliez and Desbrun (2001); Cohen-Or *et al.* (1999). Rossignac (1999) developed a method for compressing 3D triangular meshes using 1.3 and 2 bits per triangle. Decomposition of triangle meshes into triangle strips were explored to encode connectivity (Chow, 1997; Evans *et al.*, 1996; van Kaick *et al.*, 2004; Silva *et al.*, 2002). Turán (1984) explored the mesh connectivity information as a triangulated planar graph encoded with less than $12n$ bits, in a triangulation with n vertices. Keeler and Westbrook (1995) improved Turán's results by encoding planar graphs with $4.6n$ bits through a triangle-spanning tree.

MATERIAL AND METHODS

The surface representation of an image can be generally viewed as a $2\frac{1}{2}$ -dimensional modeling problem (Pedrini, 2000), where a function of two variables, $z = f(x, y)$, expresses the altitude z of the surface at a point (x, y) .

In our approach, a triangular mesh is constructed from a set of irregularly spaced data points. A Delaunay triangulation is adopted due to the suitable geometric properties mentioned in the previous section. The triangulation of a set of data points in the plane can be defined in terms of a planar graph, where pairs of vertices are connected by edges intersected only at their endpoints, forming triangular faces.

A minimal approximation of the surface is initially constructed by consisting of two triangles over the data domain. This coarse mesh is repeatedly refined by the insertion of new points until either a specified error

tolerance is achieved or a given number of points is reached. The mesh is updated after each point insertion in order to maintain the Delaunay triangulation.

The order in which the points are inserted into the triangulation is crucial to produce a compact and accurate approximation. A common strategy is to iteratively add the worst fitting point to the current triangulation according to the maximum absolute error (vertical difference) between the original and approximated models until all points are fit within a specified error tolerance. This process is straightforward to implement and generates compact triangulated meshes, however, it is sensitive to noise or outliers due to the selection of points with highest error.

In this work, we propose and evaluate new criteria for mesh refinement, extending upon the set of metrics described by van Kaick and Pedrini (2006). Each metric associates an error to a triangle Δ . Let $f(p)$ and $g(p)$ be the pixel intensity values at a point $p(x,y)$ in the original and approximated images, respectively. It is worth mentioning that, at each step of the refinement algorithm, the point with the highest error in the triangle Δ according to one of the evaluated metrics is inserted into the triangulation.

The following metrics for point insertion are evaluated in our method: maximum absolute error (MAE), absolute vertical sum (AVS), squared vertical sum (SVS), Laplacian maximum error (LMAE), Laplacian absolute vertical sum (LAVS), maximum absolute error weighted by the triangle area (MAEA), maximum absolute error weighted by the standard deviation of the errors (MAES), absolute vertical sum weighted by the standard deviation of the errors (AVSS), squared vertical sum weighted by the triangle area and standard deviation of the errors (SVSAS), Jaccard coefficient (JC), Czenakowski distance (CZD), and squared vertical sum weighted by the triangle area and the mean of the values in the original model (SVSAM). These metrics are defined in the next equations:

$$\text{MAE}(\Delta) = \max_{p \in \Delta} |f(p) - g(p)|, \quad (1)$$

$$\text{AVS}(\Delta) = \sum_{p \in \Delta} |f(p) - g(p)|, \quad (2)$$

$$\text{SVS}(\Delta) = \sum_{p \in \Delta} [f(p) - g(p)]^2, \quad (3)$$

$$\text{LMAE}(\Delta) = \max_{p \in \Delta} \frac{|f(p) - g(p)|}{k(p)}, \quad (4)$$

$$\text{LAVS}(\Delta) = \sum_{p \in \Delta} \frac{|f(p) - g(p)|}{k(p)}, \quad (5)$$

$$\text{MAEA}(\Delta) = \max_{p \in \Delta} |f(p) - g(p)| A(\Delta), \quad (6)$$

$$\text{MAES}(\Delta) = \max_{p \in \Delta} |f(p) - g(p)| \sigma(\Delta), \quad (7)$$

$$\text{AVSS}(\Delta) = \sum_{p \in \Delta} |f(p) - g(p)| \sigma(\Delta), \quad (8)$$

$$\text{SVSAS}(\Delta) = \sum_{p \in \Delta} [f(p) - g(p)]^2 A(\Delta) \sigma(\Delta), \quad (9)$$

$$\text{JC}(\Delta) = \text{MAE}(\Delta) \left(1 - \frac{\sum_{p \in \Delta} \begin{cases} 1, & \text{if } f(p) = g(p) \\ 0, & \text{otherwise} \end{cases}}{A(\Delta)} \right), \quad (10)$$

$$\text{CZD}(\Delta) = \frac{\text{MAE}(\Delta) A(\Delta)}{\sum_{p \in \Delta} \left(1 - \frac{2 \min(f(p), g(p))}{f(p) + g(p)} \right)}, \quad (11)$$

$$\text{SVSAM}(\Delta) = \frac{\sum_{p \in \Delta} [f(p) - g(p)]^2 A(\Delta)}{\bar{f}(p)}, \quad (12)$$

where $k(p) > 0$ is the curvature at point p calculated with a Laplacian filter, $\sigma(\Delta)$ is the standard deviation of the errors calculated in triangle Δ , $A(\Delta)$ denotes the area of triangle Δ , and $\bar{f}(p)$ is the mean of the values $f(p)$ in triangle Δ . Metrics MAEA, MAES, AVSS, SVSAS and SVSAM are proposed in this work.

Algorithm 1 shows the main steps of the mesh refinement process for a set of points chosen from an input image. Initially, a coarse triangular mesh is constructed with the four corners of the image (Lines 1-2). At each iteration, the Delaunay triangulation is refined by inserting the point with highest error among all triangles according to the metric under consideration, that is, one of the metrics defined in Equation 1 to 12. The intensity value in each triangle point is then approximated through an interpolation process, which is performed by computing the plane defined by the three vertices of the triangle and evaluating the plane equation at the point under consideration. This process is repeated until either a specified error tolerance is satisfied or a given number of points is achieved (Lines 3-7). The selected subset

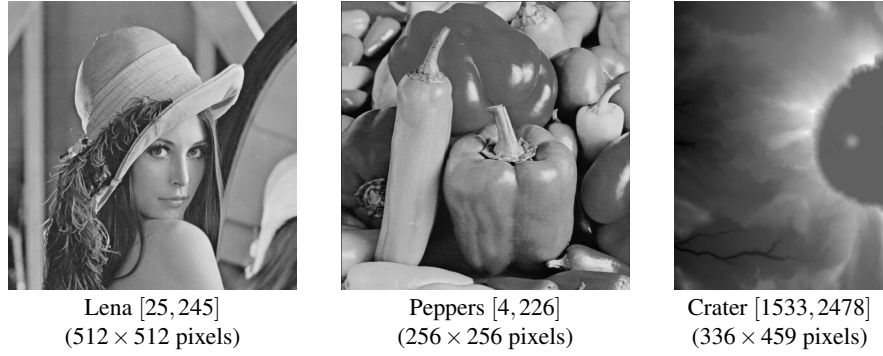


Fig. 2: Some data sets used in the experiments.

of points and its triangulation are returned as results of this mesh refinement process (line 8).

Algorithm 1: Refinement of the triangular mesh

input : image f
 point insertion metric m_p
 maximum number n of points
 tolerance error ε

output: subset P of points
 triangulation T

- 1 $P =$ four corners of f
- 2 $T =$ Delaunay triangulation(P)
- 3 **while** (highest error in $T > \varepsilon$) and (number of points $< n$) **do**
- 4 select point p with highest error according to chosen metric m_p
- 5 $P = P \cup \{p\}$
- 6 $T =$ Delaunay triangulation(P)
- 7 **end**
- 8 return P and T

The evaluation of the resulting surface approximation is performed through two common image quality measures, the root mean squared error (RMSE) and peak signal to noise ratio (PSNR), expressed as:

$$\text{RMSE} = \sqrt{\frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [f(p) - g(p)]^2}, \quad (13)$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{MNL^2}{\sum_{x=1}^M \sum_{y=1}^N [f(p) - g(p)]^2} \right), \quad (14)$$

where M and N correspond to the image dimensions and L is the maximum intensity value of the image.

RESULTS

Experiments were conducted on several data sets to demonstrate the effectiveness of the evaluated metrics for mesh refinement. To allow a further quantitative and qualitative analysis of the results, Fig. 2 shows three tested images with their respective dimensions in pixels, as well as minimum and maximum intensity (elevation) values.

Figs. 3, 4 and 5 show the triangular meshes constructed for approximations of the three images used in our experiments with the evaluated measures. All triangulations satisfy the same error level (RMSE = 15 or PSNR = 25), and the results show the spatial distribution of points generated by each metric.

The metrics AVS, SVS, LAVS, AVSS, SVSAS and SVSAM generated triangulations with a smaller number of points, fitting to the image features adequately for a fixed error rate. Other metrics, such as MAE, LMAE, MAES and JC concentrated points on specific portions of the image, particularly regions with high curvature, requiring more points. Some metrics (AVS, LAVS, MAEA, CZD and SVSAM) produced meshes with points more regularly distributed across the image, although MAEA and CZD did not generate compact triangulations.

In this work, the common criteria for mesh refinement based on the vertical error between the original and approximated models were modified in order to make them more adaptable to the data. Metrics weighted by triangle area favor the partitioning of large triangles into smaller ones, which can be more suitable for high curvature regions. On the other hand, metrics weighted by the standard deviation of the errors can prevent the selection of points based on local errors in the presence of noise (outliers).

Fig. 6 shows the number of points required for each mesh refinement metric to achieve a fixed tolerance level (RMSE ou PSNR) for each tested

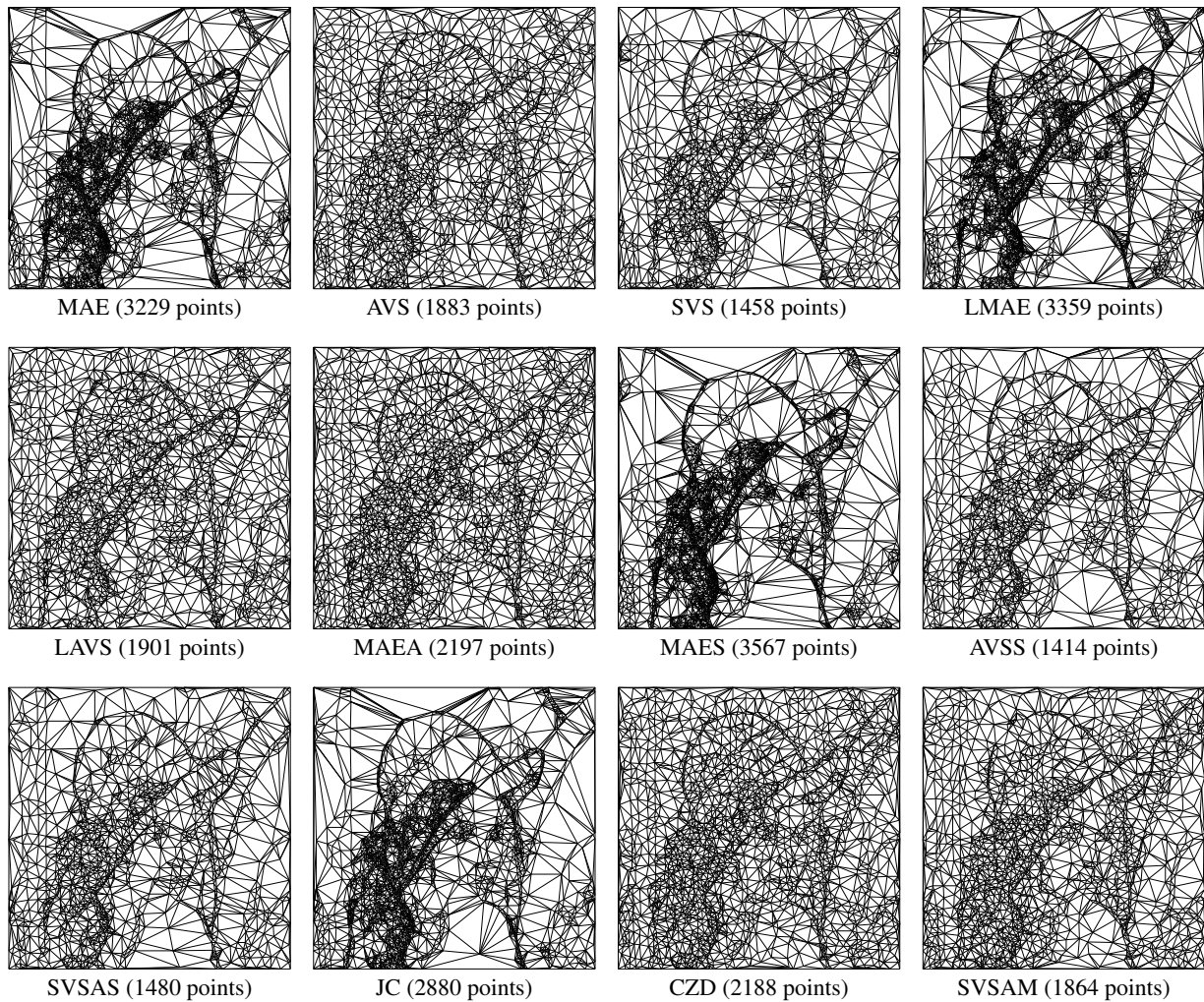


Fig. 3: Triangulations using different refinement metrics for approximations of Lena image with RMSE = 15 or PSNR = 25.

image. It is worth mentioning that the point selection criteria do not guarantee a monotonically convergence, however, the approximation errors show a decreasing behavior. From the results, it is possible to observe that new strategies for point selection can provide higher adaptability to the data when compared to the vertical error measure.

Triangulations and respective reconstructed images for the tested data sets with metric AVSS using only 1% of the original number of points are shown in Fig. 7. Despite the small number of points, the triangular meshes correspond to suitable approximations of the images, resulting in an efficient compression scheme. Error maps, computed as the difference between the intensity values of the original and approximated images, are also illustrated.

The proposed approach provides a good trade-off between performance and flexibility to construct triangular meshes from different data sets.

DISCUSSION

In this work, we evaluated several metrics for refining triangular meshes constructed to approximate surfaces from a set of sampled points. The main benefit is the generation of more compact and accurate meshes, which can be applied to a number of different practical problems.

As demonstrated in our experiments, a substantial reduction in the size of triangular meshes can be achieved through the employment of more effective meshes, improving storage requirements, transmission capacities, and rendering performance.

Directions for future work include the proposition of new heuristics for mesh refinement and evaluation of features present in the data, such as corners and lines, to guide the triangulation construction process.

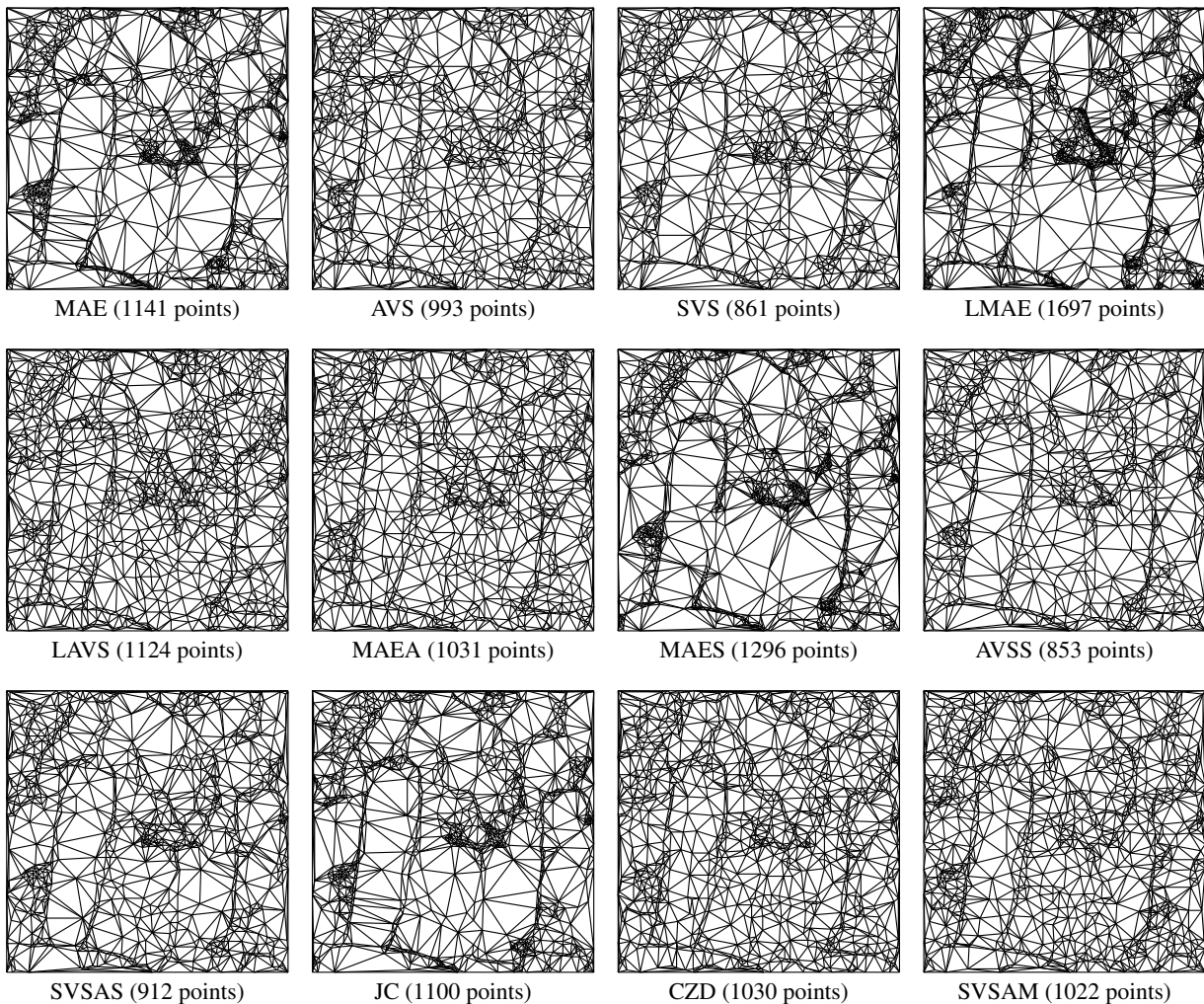


Fig. 4: Triangulations using different refinement metrics for approximations of Peppers image with RMSE = 15 or PSNR = 25.

ACKNOWLEDGEMENTS

The authors are grateful to São Paulo Research Foundation (FAPESP grant #2014/12236-1) and Brazilian National Council for Scientific and Technological Development (CNPq grant #305169/2015-7) for their financial support.

REFERENCES

- Alliez P, Desbrun M (2001). Progressive compression for lossless transmission of triangle meshes. In: Proc SIGGRAPH'01 28th Ann Conf Comp Graph 195–202.
- Araújo B, Lopes DS, Jepp P, Jorge JA, Wyvill B (2015). A survey on implicit surface polygonization. *ACM Comput Surv* 47(4):60 (39 pp).
- Barnsley MF, Hurd LP (1993). *Fractal image compression*. Wellesley, Massachusetts, USA: AK Peters.
- de Berg M, van Kreveld M, Overmars M, Schwarzkopf OC (2000). *Computational geometry*. Berlin, Heidelberg: Springer.
- Berger M, Levine JA, Nonato LG, Taubin G, Silva CT (2013). A benchmark for surface reconstruction. *ACM T Graph* 32(3):20 (17 pp).
- Boissonnat J, Pons J, Yvinec M (2009). From segmented images to good quality meshes using Delaunay refinement. In: Nielsen F, eds. *Emerging Trends in Visual Computing*. Lect Not Comput Sci 5416:13–7.
- Boissonnat JD, Shi KL, Tournois J, Yvinec M (2015). Anisotropic Delaunay meshes of surfaces. *ACM T Graph* 34(2):14.
- Bommes D, Lévy B, Pietroni N, Puppo E, Silva C, Tarini M, Zorin D (2013). Quad-mesh generation and processing: A survey. *Comput Graph Forum* 32:51–76.
- Botsch M, Kobbelt L, Pauly M, Alliez P, Lévy B (2010). *Polygon mesh processing*. New York: A K Peters/CRC Press.

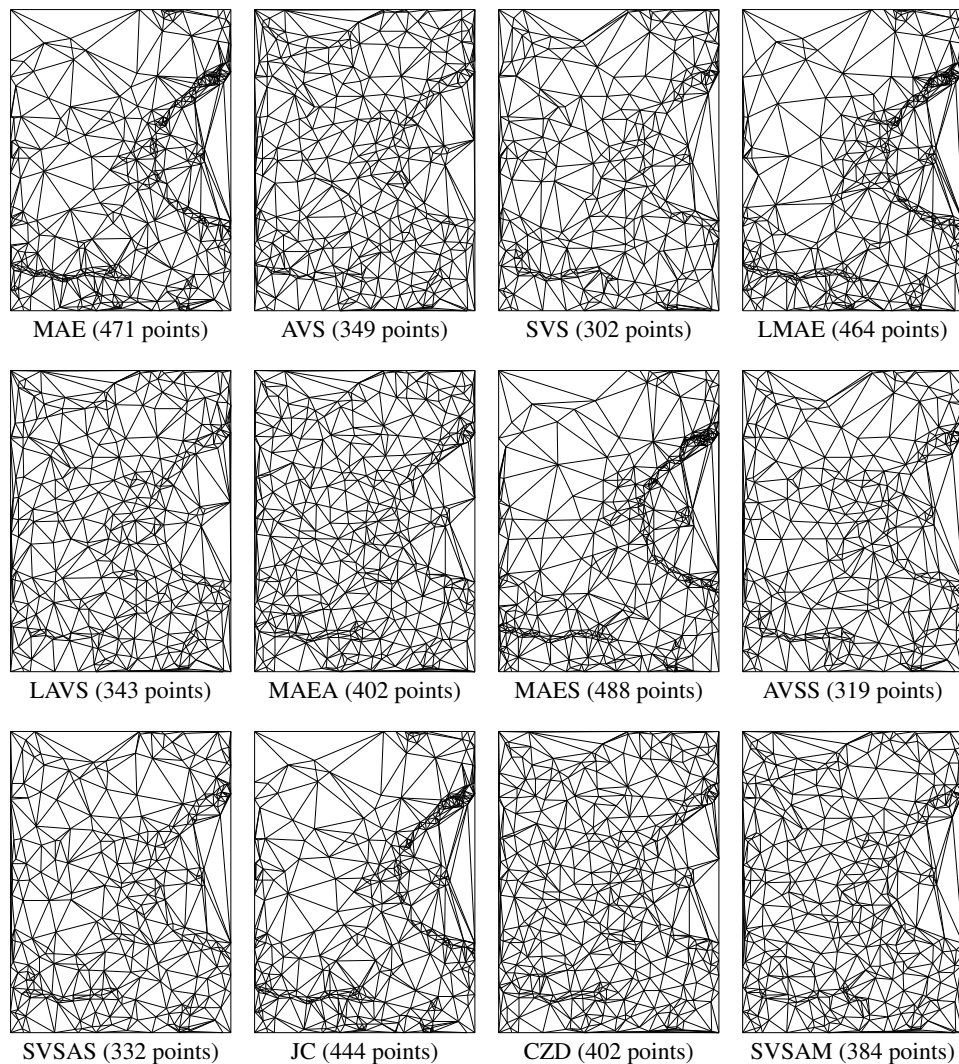


Fig. 5: Triangulations using different refinement metrics for approximations of Crater image with RMSE = 15 or PSNR = 25.

- Brown JL (1991). Vertex based data dependent triangulations. *Comput Aided Geom D* 8:239–51.
- Brown P (1997). A fast algorithm for selective refinement of terrain meshes. *Comput Networks ISDN* 29:1587–99.
- Burt P, Adelson E (1983). The Laplacian pyramid as a compact image code. *IEEE T Commun* 31:532–40.
- Chen H, Sun M, Steinbach E (2009). Compression of Bayer-pattern video sequences using adjusted chroma subsampling. *IEEE T Circ Syst Vid* 19:1891–6.
- Cheng SW, Dey TK, Shewchuk J (2012). *Delaunay mesh generation*. Boca Raton: CRC Press.
- Chow MM (1997). Optimized geometry compression for real-time rendering. In: *Proc Visualization'97*. Oct 19–24, Phoenix, AZ, USA 347–54.
- Cignoni P, Montani C, Scopigno R (1998). A comparison of mesh simplification algorithms. *Comput Graph* 22:37–54.
- Cohen-Or D, Levin D, Remez O (1999). Progressive compression of arbitrary triangular meshes. In: *Proc Conf Visualization'99*. San Francisco, CA, USA 67–72.
- de Berg M, Cheong O, van Kreveld M, Overmars M (2008). *Computational geometry: Algorithms and applications*. Berlin, Heidelberg: Springer.
- Deering M (1995). Geometry compression. In: *Proc SIGGRAPH'95 22nd Ann Conf Comp Graph* 13–20.
- DeVore RA, Jawerth B, Lucier BJ (1992). Image compression through wavelet transform coding. *IEEE T Inform Theory* 38:719–46.
- Dey T, Ray T (2010). Polygonal surface remeshing with Delaunay refinement. *Eng Comput* 26:289–301.
- Dyn N, Levin D, Rippa S (1990). Data dependent triangulations for piecewise linear interpolation. *IMA J*

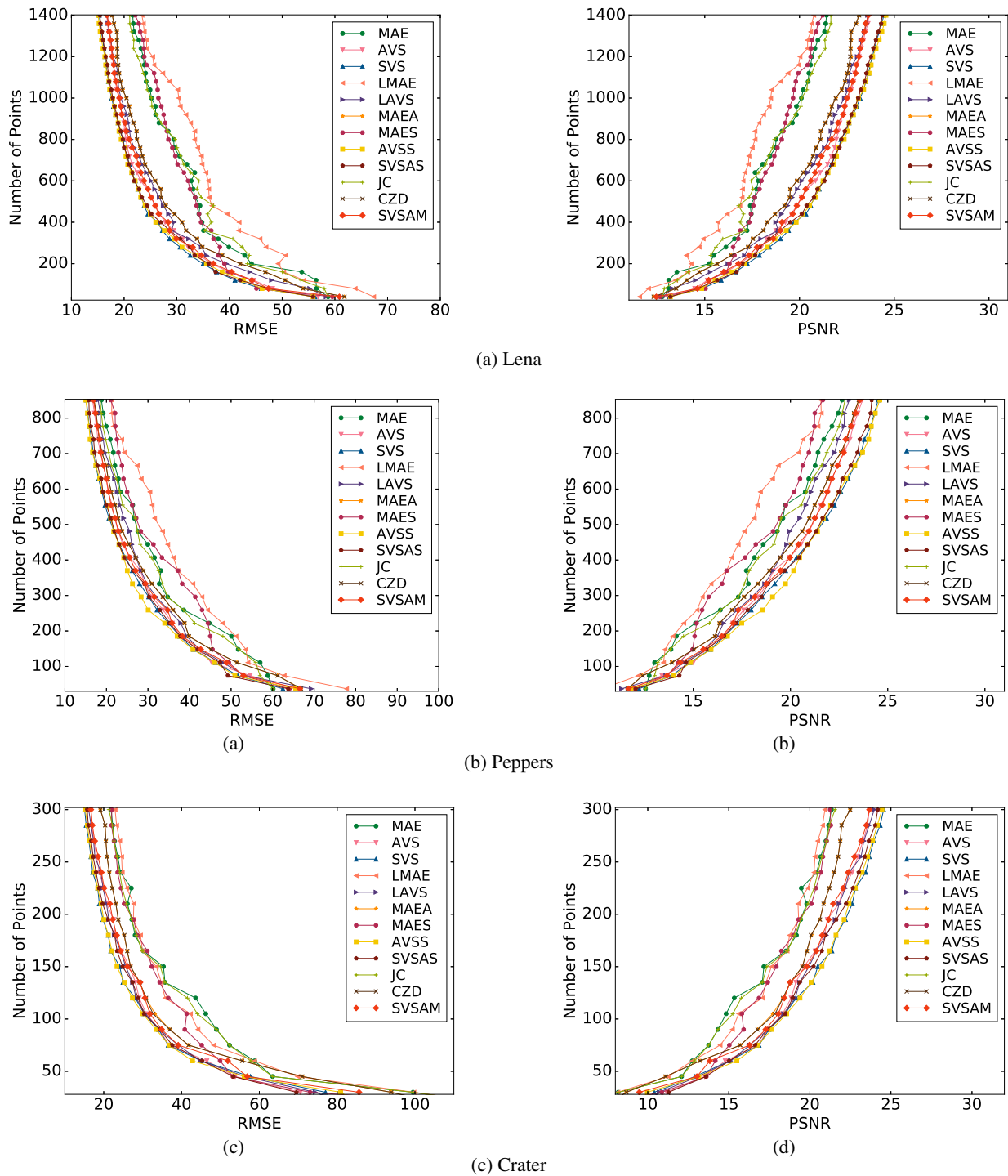


Fig. 6: Number of points required for each mesh refinement metric to generate image approximations with a fixed tolerance level.

Numer Anal 10:137–54.

Evans F, Skiena S, Varshney A (1996). Optimizing triangle strips for fast rendering. In: Proc Visualization'96. San Francisco, CA, USA. 319–26.

Fisher Y (2012). Fractal image compression: Theory and application. Berlin, Heidelberg: Springer.

Garland M, Heckbert PS (1997). Surface simplification

using quadric error metrics. In: Proc SIGGRAPH'97 24th Ann Conf Comp Graph 209–16.

Held G, Marshall TR (1996). Data and image compression: Tools and techniques. Chichester: Wiley.

Hoppe H (1996). Progressive meshes. In: Proc SIGGRAPH'96 23th Ann Conf Comp Graph 99–108.

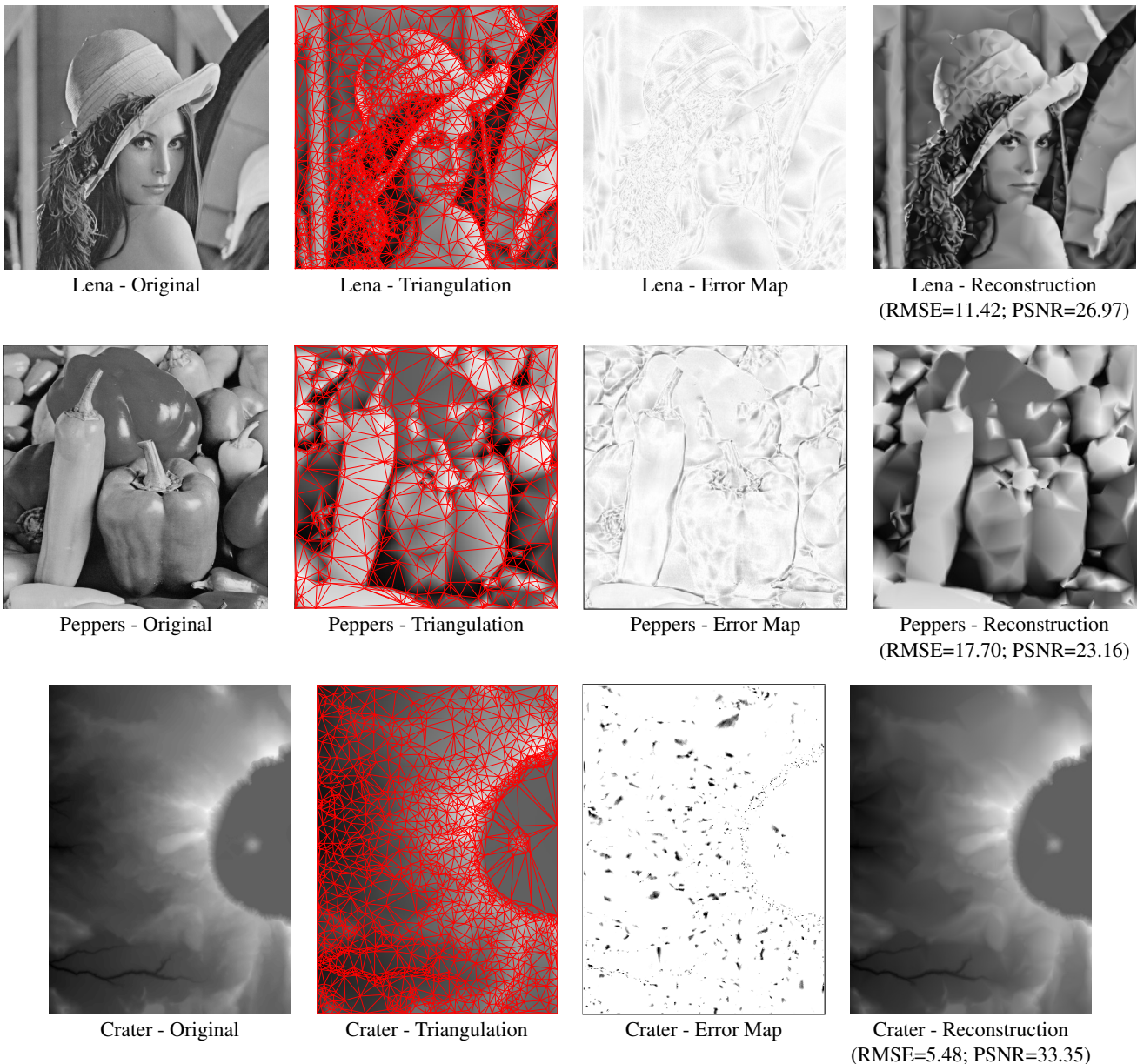


Fig. 7: Triangulations, corresponding image reconstructions, and error maps obtained with metric AVSS using only 1% of the original number of points.

Hu L, Sander PV, Hoppe H (2009). Parallel view-dependent refinement of progressive meshes. In: Proc I3D'09 Symp Interact 3D Graph Games 169–76.

Keeler K, Westbrook J (1995). Short encodings of planar graphs and maps. *Discrete Appl Math* 58:239–52.

Khodakovsky A, Schröder P, Sweldens W (2000). Progressive geometry compression. In: Proc SIGGRAPH '00 27th Ann Conf Comp Graph 271–8.

Kim SJ, Kim CH, Levin D (2002). Surface simplification using a discrete curvature norm. *Comput Graph* 26:657–63.

Kolingerová I, Kohout J, Rulf M, Uher V (2010). A proper

choice of vertices for triangulation representation of digital images. In: Bolc L, Tadeusiewicz R, Chmielewski LJ, Wojciechowski K, eds. *Computer Vision and Graphics. ICCVG 2010. Lect Not Comput Sci* 6375:41–8.

Lawson CL (1977). Software for C^1 surface interpolation. In: Rice JR, ed. *Mathematical Software III, Proc Symp* 161–94.

Lin T, Zhang P, Wang S, Zhou K, Chen X (2013). Mixed chroma sampling-rate high efficiency video coding for full-chroma screen content. *IEEE T Circ Syst Vid* 23:173–85.

- Lipman Y (2012). Bounded distortion mapping spaces for triangular meshes. *ACM T Graphic* 31(4):108.
- Luebke D, Watson B, Cohen JD, Reddy M, Varshney A (2002). *Level of detail for 3D graphics*. New York: Elsevier.
- Ma J, Feng HY, Wang L (2013). Normal vector estimation for point clouds via local Delaunay triangle mesh matching. *Comput Aided Design Appl* 10:399–411.
- Maglo A, Courbet C, Alliez P, Hudelot C (2012). Progressive compression of manifold polygon meshes. *Comput Graph* 36:349–59.
- Meyer M, Desbrun M, Schröder P, Barr AH (2003). Discrete differential-geometry operators for triangulated 2-manifolds. In: Hege HC, Polthier K, eds. *Visualization and Mathematics III*. Berlin: Springer, 35–57.
- Moraes TF, Amorim PH, da Silva JV, Pedrini H (2017). Out-of-core progressive web-based rendering of triangle meshes. In: Tavares J, Natal Jorge R, eds. *VipIMAGE 2017. ECCOMAS 2017. Lecture Notes in Computational Vision and Biomechanics*, vol 27 *Lect Not Comput Vision Biomech* 27:456–66.
- Morigi S, Rucci M (2013). Multilevel mesh simplification. *Visual Comput* 30:479–92.
- Mostafavian S, Adams MD (2015). An optimization-based mesh-generation method for image representation. In: *Proc 2015 IEEE Pacif* 234–9.
- Nivoliers V, Yan DM, Lévy B (2012). Fitting polynomial surfaces to triangular meshes with Voronoi squared distance minimization. *Eng Comput* 30:289–300.
- Pedrini H (2000). *An adaptive method for terrain surface approximation based on triangular meshes*. PhD Thesis. Troy, NY, USA: Rensselaer Polytechnic Institute.
- Peng J, Kim CS, Kuo CCJ (2005). Technologies for 3D mesh compression: A survey. *J Vis Commun Image R* 16:688–733.
- Polthier K, Schmies M (2006). Straightest geodesics on polyhedral surfaces. In: *Proc SIGGRAPH'06 Courses* 30–8.
- Preparata FP, Shamos M (1985). *Computational geometry: An introduction*. New York: Springer.
- Quak E, Schumaker LL (1990). Cubic spline fitting using data dependent triangulations. *Comput Aided Geom D* 7:293–301.
- Rabbani M, Jones PW (1991). *Digital image compression techniques*. Bellingham, USA: SPIE Press.
- Ren M, Yang J, Sun H (2002). Tracing boundary contours in a binary image. *Image Vision Comput* 20:125–31.
- Rigler S, Bishop W, Kennings A (2007). FPGA-based lossless data compression using Huffman and LZ77 algorithms. In: *Proc IEEE Can Conf Elec Comput Eng* 1235–8.
- Rippa S (1992a). Adaptive approximation by piecewise linear polynomials on triangulations of subsets of scattered data. *SIAM J Sci Stat Comp* 13:1123–41.
- Rippa S (1992b). Long and thin triangles can be good for linear interpolation. *IMA J Numer Anal* 29:257–70.
- Rossignac J (1999). Edgebreaker: Connectivity compression for triangle meshes. *IEEE T Vis Comput Gr* 5:47–61.
- Salinas D, Lafarge F, Alliez P (2015). Structure-aware mesh decimation. *Comput Graph Forum* 34:211–27.
- Samet H (1990). *The design and analysis of spatial data structures*. Boston: Addison-Wesley.
- Samet H (2006). *Foundations of multidimensional and metric data structures*. St Louis: Morgan Kaufmann.
- Sánchez-Cruz H, Bribiesca E, Rodríguez-Dagnino RM (2007). Efficiency of chain codes to represent binary objects. *Pattern Recogn* 40:1660–74.
- Scarlatos L (1992). Hierarchical triangulation using cartographic coherence. *Graph Model Im Proc* 54:147–61.
- Scholefield A, Dragotti PL (2014). Quadtree structured image approximation for denoising and interpolation. *IEEE T Image Proc* 23:1226–39.
- Shaffer E, Garland M (2001). Efficient adaptive simplification of massive meshes. In: *Proc IEEE Conf Visual VIS'01* 127–34.
- Silva MVG, van Kaick OM, Pedrini H (2002). Fast mesh rendering through efficient triangle strip generation. *Journal WSCG* 10:127–34.
- Skodras A, Christopoulos C, Ebrahimi T (2001). The JPEG 2000 still image compression standard. *IEEE Signal Proc Mag* 18:36–58.
- Sullivan GJ, Baker RL (1994). Efficient quadtree coding of images and video. *IEEE T Image Process* 3:327–31.
- Taubin G, Rossignac J (1998). Geometric compression through topological surgery. *ACM T Graphic* 17:84–115.
- Tóth Z, Viola I, Ferko A, Gröller E (2007). *N-dimensional data-dependent reconstruction using topological changes*. In: Hauser H, Hagen H, Theisel H, eds. *Topology-based Methods in Visualization*. Berlin, Heidelberg: Springer, 183–98.
- Tran A, Liu KM, Tzou KH, Vogel E (1987). An efficient pyramid image coding system. In: *Proc IEEE Int Conf Acoust Spee ICASSP'87* 744-7.
- Turán G (1984). On the succinct representation of graphs. *Discrete Appl Math* 8:289–94.
- Van Aerscht W, Jansen M, Bultheel A (2009). Normal mesh based geometrical image compression. *Image Vision Comput* 27:459–68.
- van Kaick OM, Pedrini H (2006). Assessment of image surface approximation accuracy given by triangular

- meshes. In: Wojciechowski K, Smolka B, Palus H, Kozera R, Skarbek W, Noakes L, eds. *Computer vision and graphics*. Warsaw: Springer 655–61.
- van Kaick OM, Silva MVG, Pedrini H (2004). Efficient generation of triangle strips from triangulated meshes. *Journal WSCG* 12:475–81.
- Wang K, Lo CP, Brook GA, Arabnia HR (2001). Comparison of existing triangulation methods for regularly and irregularly spaced height fields. *Int J Geogr Inf Sci* 15:743–62.
- Xie YH, Tang XA, Sun MY (2008). Image compression based on classification row by row and LZW encoding. In: *Proc Congr Image Signal Process* 617–21.