An Improved Error-Diffusion Approach for Generating Mesh Models of Images

Xiao (Brian) Ma and Michael D. Adams*

Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada

Abstract

In earlier work, Yang et al. proposed a highly-effective technique for generating triangle-mesh models of images, known as the error diffusion (ED) method. Unfortunately, the ED method, which chooses triangulation connectivity via a Delaunay triangulation, typically yields triangulations in which many triangulation edges crosscut image edges, leading to increased approximation error. In this paper, we propose a computational framework for mesh generation that modifies the ED method to use data-dependent triangulations (DDTs) in conjunction with the Lawson local optimization procedure (LOP) and has several free parameters. Based on experimentation, we recommend two particular choices for these parameters, yielding two specific mesh-generation methods, known as MED1 and MED2, which make different tradeoffs between approximation quality and computational cost. Through the use of DDTs and the LOP, triangulation connectivity can be chosen optimally so as to minimize approximation error. As part of our work, two novel optimality criteria for the LOP are proposed, both of which are shown to outperform other well known criteria from the literature. Through experimental results, our MED1 and MED2 methods are shown to yield image approximations of substantially higher quality than those obtained with the ED method, at a relatively modest computational cost.

Keywords: Image representations; nonuniform sampling; triangle meshes; data-dependent triangulations; error diffusion.

1. Introduction

In real-world applications, images are typically nonstationary. Consequently, uniform sampling of images (such as with a truncated lattice) is usually far from optimal, with the sampling density inevitably being

*Corresponding author. Mailing address: Department of Electrical and Computer Engineering, University of Victoria, PO Box 1700 STN CSC, Victoria, BC, V8W 2Y2, Canada; tel.: +1 250 721 6025; fax: +1 250 721 6052; e-mail: mdadams@ece.uvic.ca.

too high in some regions while too low in others. This has led to an interest in image representations based on nonuniform (i.e., content-adaptive) sampling. By choosing the sample points in a manner dependent on the image content, the number of samples can be greatly reduced. This smaller sample count can often be exploited in applications in order to reduce computational cost. Moreover, the sample data can often better capture the geometric structure inherent in images (such as image edges). In some applications, this can be exploited in order to obtain better quality results. Some applications in which nonuniform sampling has proven useful include: feature detection [1], pattern recognition [2], computer vision [3], restoration [4], tomographic reconstruction [5], filtering [6], interpolation [7, 8], and image/video coding [9–15].

Many general approaches to nonuniform sampling have been proposed to date. Some of the more popular approaches include: inverse distance weighted methods [16, 17]; radial basis function methods [16, 17]; Voronoi and natural-neighbor interpolation methods [16]; and finite-element methods [16, 17], including triangle meshes based on Delaunay triangulations [18–21], constrained Delaunay triangulations [22], data-dependent triangulations [23–28, 18, 29, 30], and geodesic triangulations [31]. Two excellent survey papers [16] and [17] present a good overview of the numerous general approaches to nonuniform sampling.

One particularly effective approach to nonuniform sampling is offered by triangle meshes. In this approach, the (nonuniformly chosen) sample points are triangulated, partitioning the image domain into triangular faces, and then an approximating function is constructed over each face of the triangulation. One key difference between the various triangle-mesh-based approaches is in how they select the triangulation connectivity (i.e., how the vertices of the triangulation are connected by edges). The most common approach is to choose the connectivity by using a Delaunay triangulation [32]. In such a case, the connectivity is determined solely by the set of sample points being triangulated. By choosing an appropriate technique for handling degeneracies, such as preferred directions [33], a unique triangulation can be obtained. Examples of mesh-generation methods that are based on Delaunay triangulations are plentiful in literature, a few examples of which are [18–21]. Another approach to choosing the triangulation connectivity is offered by **data-dependent triangulations (DDTs)**. With a DDT, the triangulation connectivity can be chosen in an arbitrary manner, using information in the dataset from which the points to be triangulated were chosen. Since, unlike the Delaunay case, the connectivity of a DDT may be chosen arbitrarily, DDTs offer much greater flexibility than Delaunay triangulations. This said, however, connectivity selection is often a

challenging task. Typically, optimization techniques are employed for this purpose, with the most common such technique, by far, being the **local optimization procedure (LOP)** of Lawson [34]. Examples of mesh-generation methods based on DDTs include [23–28, 18, 29, 30]. These approaches make heavy use of the LOP or variants thereof, such as the **look-ahead LOP (LLOP)** [29].

In [19], Yang et al. proposed a simple technique for generating triangle-mesh models of images, known as the **error-diffusion (ED)** method. Although this method has proven highly effective, it has the weakness that it often yields triangulations in which a significant number of (triangulation) edges crosscut image edges (i.e., discontinuities in the image), leading to a degradation in approximation quality. This weakness can be attributed to the fact that the ED method employs a Delaunay triangulation for choosing triangulation connectivity. In this paper, we propose a computational framework for mesh generation that modifies the ED method to use DDTs in conjunction with the LOP. By using DDTs instead of Delaunay triangulations, we are able to better exploit triangulation connectivity in order to obtain higher quality approximations. Using our framework, we derive two specific mesh-generation methods known as MED1 and MED2, which make different tradeoffs between approximation quality and computational cost. As we will show later, our MED1 and MED2 methods yield image approximations of substantially higher quality than those obtained with the ED method in terms of both **peak-signal-to-noise ratio (PSNR)** and subjective quality, at a relatively modest computational cost. For example, in terms of PSNR, our MED1 and MED2 methods outperform the ED method, on average, by 3.26 and 3.81 dB, respectively. As part of our work, we propose two novel optimality criteria for use with the LOP. Both of these criteria are shown to outperform numerous other well known criteria from the literature. In passing, we note that the work described herein has been partially presented in our conference paper [35]. The work herein, however, adds a number of new elements beyond our conference paper, such as: considering the use of the LLOP, proposing two new highly-effective optimality criteria for the LOP, and exploiting one of these criteria in order to obtain more effective mesh-generation schemes. Furthermore, as will be demonstrated later, the mesh-generation method proposed in our conference paper, henceforth referred to as the CCCG method, produces meshes of significantly lower quality than those obtained with the MED1 and MED2 methods proposed herein.

The remainder of this paper is organized as follows. To begin, Section 2 provides some background information on triangle meshes for image representation and introduces some key methods related to our

work. In Section 3, we begin by introducing our computational framework for mesh generation. Then, we consider how the free parameters of this framework should be chosen in order to achieve the best performance, leading to the proposal of our mesh-generation methods MED1 and MED2. In Section 4, the performance of our mesh-generation methods are evaluated. Finally, Section 5 concludes with a brief summary of our work and some closing remarks.

2. Background

Before proceeding further, a brief digression is necessary in order to introduce some basic notation and terminology employed herein. The cardinality of a set S is denoted $|S|$, and the 2-norm of a vector v is denoted $||v||$. The ***triangulation*** of a set P of points is a set T of (nondegenerate) triangles such that: 1) the union of the vertices of all triangles in T is P; 2) the interiors of any two triangles in T are disjoint; and 3) the union of the triangles in T is the convex hull of P.

In the context of our work, an image is an integer-valued function ϕ defined on the domain $I = [0, W - 1] \times [0, H - 1]$ and sampled on the truncated two-dimensional integer lattice $\Lambda = \{0, 1, \ldots, W - 1\} \times \{0, 1, \ldots, H - 1\}$ (i.e., a rectangular grid of width W and height H). A (triangle) mesh model of ϕ consists of: 1) a set $P = \{p_i\}$ of ***sample points***, where $P \subset \Lambda$; 2) a triangulation T of P; and 3) the function values $\{z_i = \phi(p_i)\}$ for $p_i \in P$. In order to ensure that the triangulation T covers all points in Λ, P must always be chosen to include all of the extreme convex hull points of I (i.e., the four corner points of the image bounding box). As a matter of terminology, the ***size*** and ***sampling density*** of the model are defined as $|P|$ and $|P| / |\Lambda|$, respectively.

The above mesh model is associated with a function $\hat{\phi}$ that approximates ϕ, where $\hat{\phi}$ is determined as follows. First, we construct a continuous piecewise linear function $\tilde{\phi}$ that interpolates ϕ at each point $p_i \in P$. More specifically, for each face f in the triangulation T, $\tilde{\phi}$ is defined to be the unique linear function that interpolates ϕ at the three vertices of f. Since ϕ is integer valued, we wish for its approximation $\hat{\phi}$ to be integer valued as well. Thus, we define the approximation $\hat{\phi}$ in terms of $\tilde{\phi}$ as $\hat{\phi}(p) = \text{round}(\tilde{\phi}(p))$, where round denotes an operator that rounds to the nearest integer.

In our work, for a given model size (i.e., number of sample points), we want to find a model to minimize

ϵ, the difference between $\hat{\phi}$ and ϕ as measured by the **mean squared error (MSE)**, where

$$\epsilon = |\Lambda|^{-1} \sum_{p \in \Lambda} \left(\hat{\phi}(p) - \phi(p) \right)^2 . \tag{1}$$

For convenience, we will express the MSE in terms of the PSNR, which is defined as $\mathsf{PSNR} = 20 \log_{10}[(2^{\rho} - 1)/\sqrt{\epsilon}]$, where ρ is the number of bits per sample used by the (integer-valued) image ϕ. Finding computationally efficient methods to solve the above problem is extremely challenging, as problems like this are known to be NP-hard [36].

ED Method. As mentioned earlier, one highly effective method for generating mesh models of images is the ED method [19]. Since our work builds on the ED method, it is helpful to briefly introduce this method here. Given an image ϕ and a desired mesh size N, the ED method constructs a mesh model of ϕ with the set P of sample points, as follows:

1. Sample-point selection. Select P, with $|P| = N$, using Floyd-Steinberg error diffusion [37]. This is done in such a way as to ensure that the points in P are distributed with a density approximately proportional to the maximum-magnitude second-order directional derivative of ϕ.
2. Triangulation. Triangulate P using a Delaunay triangulation.

In step 1, the set P is always chosen to include all extreme convex-hull points of the image domain. This ensures that the triangulation produced in step 2 covers the entire image domain. Since several variants of the ED scheme are presented in [19], it is worth noting, for the sake of completeness, that we consider the variant with the following characteristics herein: 1) a third-order binomial filter is used for smoothing; 2) non-leaky error diffusion is used with a serpentine scan order; 3) the sensitivity parameter γ is chosen as 1; and 4) the error diffusion algorithm is performed iteratively in order to achieve exactly the desired number of sample points. Since, in our work herein, we require that the approximating function (i.e., $\hat{\phi}$) interpolate the original (i.e., ϕ), we consider only the variant of the ED method that satisfies this interpolating condition. (That is, the variant that employs a least-squares fit is not considered.)

LOP. Before proceeding further, it is necessary to interject some additional background related to triangulations. An edge e of a triangulation is said to be ***flippable*** if e has two incident faces (i.e., is not on the triangulation boundary) and the union of these two faces is a strictly convex quadrilateral q. For a flippable edge e, an ***edge flip*** is an operation that replaces the edge e in the triangulation by the other diagonal e' of

q, as shown in Figure 1.

[Figure 1 about here.]

The fact that every triangulation of a set of points is reachable from every other triangulation of the same set of points via a finite sequence of edge flips [38] motivated Lawson to propose the so called LOP [34].

The LOP [34] is an optimization technique, based on edge flips, that is used to select the connectivity of a triangulation so as to be optimal in some sense. In practice, the LOP is frequently used to choose triangulation connectivity in the case of DDTs. As a matter of terminology, a flippable edge e is said to be ***optimal*** if it satisfies some prescribed edge-optimality criterion. In turn, a triangulation T is said to be ***optimal*** if every flippable edge in T is optimal. In order to produce an optimal triangulation, the LOP simply applies edge flips to flippable edges that are not optimal, until the triangulation is optimal (i.e., all flippable edges are optimal).

Cost-Based Criteria. Most frequently, the edge-optimality criterion is specified indirectly through some measure of triangulation cost. Let $\text{triCost}(T)$ denote the cost of the triangulation T. A flippable edge e in the triangulation T is then said to be ***optimal*** if

$$\text{triCost}(T) \leq \text{triCost}(T'), \tag{2}$$

where T' is the new triangulation obtained by applying an edge flip to e (in the triangulation T). That is, the flippable edge e is deemed optimal if applying an edge flip to e would not result in a strict decrease in the triangulation cost. In turn, the triangulation cost triCost is specified by defining a cost measure for all edges in the triangulation. Let $\text{edgeCost}(T, e)$ denote the cost of the edge e in the triangulation T. Then, triCost is defined as

$$\text{triCost}(T) = \sum_{e \in \mathcal{E}(T)} \text{edgeCost}(T, e), \tag{3}$$

where $\mathcal{E}(T)$ denotes the set of edges in T. That is, the cost of a triangulation is simply the sum of its corresponding edge costs. As a matter of terminology, we refer to a triangulation optimality criterion employing (2) (where triCost is of the form of (3)) as ***cost based***. By far, cost-based criteria are most commonly used in conjunction with the LOP, several examples of which can be found in [23, 39, 25, 26]. A

particularly important criterion of this type is **squared error (SE)** [25, 26]. With the SE criterion, the edge e is deemed optimal if applying an edge flip to e would not cause a strict decrease in the MSE as defined by (1).

Heuristic-Based Criteria. More recently, the work [30] introduced a type of triangulation optimality criterion that is not associated with any underlying triangulation cost function (i.e., a function of the form of (3)). With this type of criterion, a cost is assigned to each flippable edge. Let edgeCost(T, e) denote the cost of the edge e in the triangulation T. The flippable edge e is said to be ***optimal*** if

$$\text{edgeCost}(T, e) \leq \text{edgeCost}(T', e'), \tag{4}$$

where e' is the new edge produced by applying an edge flip to e and T' is the corresponding new triangulation (with e'). As a matter of terminology, we refer to a triangulation optimality criterion using (4) as ***heuristic based***.

Additional Remarks on the LOP. At this point, it is worthwhile to make a few additional remarks about the LOP. The first comment to be made is with respect to algorithm termination. If a cost-based optimality criterion is employed, the LOP must terminate after a finite number of steps (assuming the algorithm is implemented in a numerically robust manner). This is an indirect consequence of the fact that the LOP only flips an edge if doing so would result in a *strict* decrease in the triangulation cost. In contrast, if a heuristic-based optimality criterion is used (regardless of whether the implementation is numerically robust), the LOP can potentially become trapped in a cycle, repeating the same sequence of edge flips indefinitely. This is due to the fact that, in the absence of a well-defined triangulation cost function, it is possible to make inconsistent decisions about the optimality of an edge. Such inconsistent decisions can result in cycles. From a practical standpoint, this potential cycling issue does not pose any significant problems for two reasons. First, when performing the LOP, it is easy to avoid being trapped in a cycle by simply tracking how many times each edge is tested for optimality and if the count for an edge exceeds a particular threshold some special action can be taken, such as ignoring the edge for the remainder of the LOP or terminating the LOP early. Second, the more effective heuristic-based criteria only rarely result in cycles. Therefore, breaking cycles when they do occur has little impact on the result produced by the LOP. In the implementation employed in our work, in the case of heuristic-based criteria, we limit the number of

times an edge may be tested for optimality to 15. If this count is exceeded, the edge in question is simply ignored for the remainder of the LOP.

The second remark to make about the LOP concerns the optimal triangulation that it produces. For any optimality criterion of practical interest (other than the Delaunay criterion [32, 33]), the optimal solution produced by the LOP is almost never uniquely determined. The nonuniqueness of the solution is important because it implies that some optimal solutions may be (and, in practice, are) much better than others. The optimum produced will typically depend (often very heavily) on the initial triangulation to which the LOP is applied.

LLOP. Suppose that the LOP is used in conjunction with a cost-based optimality criterion. In this case, if a triangulation T is optimal, then no single edge flip can result in a new triangulation with strictly lower cost than T. If, however, more than one edge flip is allowed, it can no longer be guaranteed that the triangulation cost will not strictly decrease. In this sense, the LOP only guarantees a locally (but not necessarily globally) optimal triangulation. Since some local minima will, in practice, have a much lower cost than others, it would be advantageous to have some means to reduce the likelihood of converging to a poor local minimum. This observation motivated Yu et al. to propose the so called LLOP [29]. The LLOP is similar to the LOP in that the LLOP applies edge-flip-based transformations to a triangulation until the triangulation is optimal. The LLOP, however, differs from the LOP in two key respects. The first difference is that, instead of only allowing the triangulation to be transformed by a single edge flip in each step, the triangulation can be transformed by: 1) a single edge flip; or 2) a sequence of two edge flips, where the two edges involved share a common face. The second difference is that the definition of triangulation optimality is changed to the following: A triangulation T is said to be ***optimal*** if the application of a single transformation of one of the two above types cannot produce a new triangulation whose cost is strictly less than that of T. By being allowed to apply sequences of two edge flips (instead of just individual edge flips), the LLOP is able to reduce the likelihood of converging to a very poor local minimum. In effect, when trying to minimize the triangulation cost, the LLOP considers the effect of not just single edge flips (like the LOP) but also sequences of two edge flips. In practice, the LLOP usually produces a better local optimum (i.e., a triangulation with lower cost) than the LOP. The disadvantage of the LLOP is that it typically requires more computation time and can be quite difficult to implement in a numerically robust manner. Since the

LLOP fundamentally relies on the existence of a triangulation cost function, the LLOP can only be used in conjunction with optimality criteria that are cost based. In other words, the LLOP cannot be used with heuristic-based optimality criteria.

3. Proposed Approach and Its Development

Having introduced the necessary background, we now turn our attention to introducing the two mesh-generation methods proposed in this paper. As explained earlier, the ED method chooses triangulation connectivity using a Delaunay triangulation. Experimentally, however, we have observed that selecting the connectivity in this way results in a mesh in which triangulation edges often crosscut image edges (i.e., discontinuities in the image), leading to a degradation in approximation quality. This motivated us to consider choosing triangulation connectivity in a more flexible manner, using a DDT instead of a Delaunay triangulation.

In what follows, we will first introduce our general computational framework for mesh generation, which has several free parameters. Then, by advocating two particular choices for these parameters, we will arrive at the two specific mesh-generation methods proposed herein, namely MED1 and MED2. Since it is helpful for the reader to see how we arrived at these choices, we provide significant detail in this regard, including some experimental results.

3.1. Computational Framework for Mesh Generation

Given an image ϕ and a desired mesh size N as input, our general computational framework for mesh generation produces a mesh model of ϕ having the set P of sample points, with $|P| = N$, and the associated triangulation T. To accomplish this objective, our framework performs the following (in order):

1. Sample-point selection. Select P using the same sample-point selection strategy in step 1 of the ED method (as introduced earlier in Section 2).
2. Initial mesh construction. For each point $p \in P$ using the order specified by insOrder, where insOrder is a free parameter of the framework:
   (a) Insert p in the triangulation T. This is accomplished by deleting any faces containing p and retriangulating the resulting hole. This point-insertion process is illustrated in Figure 2.

(b) Adjust the connectivity of T by applying the LOP (as described in Section 2) with the triangulation optimality criterion chosen as insOptCriterion, where insOptCriterion is a free parameter of our framework.

3. Final connectivity adjustment. Adjust the connectivity of T by applying either the LOP or LLOP, as specified by the parameter fcaMethod, with the optimality criterion chosen as SE (i.e., squared error). If fcaMethod is LOP, the LOP is employed in this step; otherwise (i.e., if fcaMethod is LLOP), the LLOP is used.

[Figure 2 about here.]

In step 2b of the above framework, the choice of the triangulation optimality criterion insOptCriterion is critical, as different choices of insOptCriterion will typically lead to vastly differing meshes. One of the optimality criteria considered in our work is the SE criterion introduced in Section 2. We also considered numerous other criteria, which we will introduce shortly. Before proceeding further, however, there is a very important comment that we must make regarding our above framework. Since our objective is to produce a mesh that minimizes the MSE (as given by (1)), this suggests the "obvious" solution of choosing the optimality criterion insOptCriterion as SE and simply skipping final connectivity adjustment (i.e., step 3) altogether. In other words, the obvious solution would be to simply optimize for squared error using the LOP after the insertion of each point in step 2. As it turns out, this obvious solution performs extremely poorly. This poor performance is due to an interplay between point insertion and the SE criterion in step 2b, which leads to triangulations with many poorly-chosen sliver (i.e., long thin) triangles, severely degrading approximation quality. In effect, this interplay causes the mesh-generation optimization process to converge to an extremely poor local optimum. To combat this problem, our framework allows the parameter insOptCriterion to be chosen differently from SE, and then adds a final-connectivity-adjustment step employing the SE criterion in order to reduce the squared error for the final mesh.

Insertion Order. Recall that step 2 of our framework (i.e., initial mesh construction) utilizes the parameter insOrder, which specifies the order in which points are to be inserted in the triangulation. In our work, we considered numerous possible choices for the insertion order insOrder, including:

1. randomized order: the extreme convex-hull points followed by the remaining points in randomized order;

2. xy-lexicographic order: the extreme convex-hull points followed by the remaining points in xy-lexicographic order;
3. farthest-point first order: the extreme convex-hull points followed by the remaining points prioritized such that the point most distant from the vertices in the triangulation is inserted first; and
4. closest-point first order: the extreme convex-hull points followed by the remaining points prioritized such that the point nearest another vertex in the triangulation is inserted first.

Detailed experiments showed randomized order (i.e., item 1 above) to be most effective. In particular, we found that, relative to randomized order, no one of the other insertion orders considered was able to consistently produce higher quality meshes at lower or comparable computational cost. Consequently, we advocate that insOrder always be chosen as randomized order, and we assume that this choice is always made for the remainder of this paper.

Optimality Criteria. Recall that step 2b of our framework (i.e., connectivity adjustment after point insertion) utilizes the parameter insOptCriterion, which determines the particular triangulation optimality criterion used for connectivity adjustment. In our work, we considered the following twelve possibilities for the choice of the optimality criterion insOptCriterion:

1. **squared error (SE)**, as given by Equation 1 in [25] and Section 2 in [26];
2. (preferred-direction) Delaunay, as specified in Section 2 in [33] and Section 11.2 in [34];
3. **angle between normals (ABN)**, as defined by Equation 3 in [23];
4. **jump in normal derivatives (JND)**, as specified in Section 3.1 in [23];
5. **deviations from linear polynomials (DLP)**, as given in Section 3.1 in [23];
6. **distances from planes (DP)**, as defined in Section 3.1 in [23];
7. **absolute mean curvature (AMC)**, as specified in Section 2.2 in [39];
8. **Garland-Heckbert hybrid (GHH)**, as described in Algorithm IV and Section 4.5.1 in [18] and Section III.B in [30];
9. **shape-quality-weighted SE (SQSE)**, as defined in Section III.B in [30];
10. **JND-weighted SE (JNDSE)**, as specified in Section III.B in [30];
11. **edge-length-weighted SE (ELSE)**, which is newly proposed herein; and
12. **minimum-angle-weighted SE (MASE)**, which is newly proposed herein.

The first ten of the above criteria are well known criteria taken from the literature, while the remaining two (namely, ELSE and MASE) are newly proposed in this paper. In the interest of brevity, we will only present herein the formal mathematical definitions of the two new criteria. The definitions of the other optimality criteria can be found in the references provided above. Of the old criteria (i.e., the first ten) the SE, Delaunay, ABN, JND, DLP, DP, and AMC criteria are all cost based (i.e., employ (2)), the SQSE and JNDSE criteria are heuristic based (i.e., employ (4)), and the GHH criterion is a hybrid of two cost-based criteria.

Before formally defining the ELSE and MASE criteria, we must first introduce some additional notation. For a triangulation T, let $\Gamma(T)$ denote the set of all integer lattice points falling inside or on the boundary of T. For a given triangulation T, let face_T denote a function that maps each point $p \in \Gamma(T)$ to *exactly one* face in T, where this function is defined as follows. If p is strictly inside a face f in T, $\text{face}_T(p) = f$ (i.e., p is mapped to f). Otherwise (i.e., if p is on an edge or a vertex in T), the method of [40] is used to uniquely map p to *exactly one* face in T. The set of all points $p \in \Gamma(T)$ satisfying $\text{face}_T(p) = f$ is denoted $\text{points}_T(f)$. With this notation in place, we can now proceed to present the ELSE and MASE criteria.

The ELSE and MASE criteria are both heuristic based (i.e., employ (4)). Therefore, each of these criteria is completely specified in terms of an edge-cost function. For a flippable edge e in the triangulation T, the edge-cost functions for the ELSE and MASE criteria are given, respectively, by

$$\text{edgeCost}_{\mathsf{ELSE}}(T, e) = \|e\| \left[\beta(T, f_i) + \beta(T, f_j)\right] \quad \text{and} \tag{5a}$$

$$\text{edgeCost}_{\mathsf{MASE}}(T, e) = \frac{\beta(T, f_i) + \beta(T, f_j)}{\min\left\{\theta(f_i), \theta(f_j)\right\}}, \tag{5b}$$

where

$$\beta(T, f) = \sum_{p \in \text{points}_T(f)} \left(\hat{\phi}(p) - \phi(p)\right)^2,$$

f_i and f_j denote the two faces incident to e, $\theta(f)$ denotes the minimum interior angle of the face f, and points_T is as defined earlier.

Final Connectivity Adjustment. In step 3 of our framework, the fcaMethod parameter is used to select whether the LOP or LLOP is used for final connectivity adjustment. Having the ability to choose

between the LOP and LLOP provides us with more flexibility to trade off between mesh quality and computational cost. In case the reader might be wondering why we did not allow similar flexibility to choose between the LOP and LLOP for connectivity adjustment after point insertion (i.e., in step 2b), we explain our rationale for this decision in what follows. The overriding reason for this decision was that, as we shall see later, all of the most effective triangulation optimality criteria during point insertion (i.e., in step 2b) are heuristic based, and such criteria cannot be used with the LLOP. Consequently, allowing the use of the LLOP during point insertion would not facilitate the development of a better mesh-generation method. To a much lesser extent, our decision was also influenced by computational cost considerations. In particular, much more time is typically spent performing connectivity adjustment in step 2b (in total) than in step 3. Thus, the increase in computational cost resulting from replacing the LOP with the LLOP in step 2b is much higher than that of replacing the LOP with the LLOP in step 3. Due to these (as well as other) factors, our framework only accommodates the use of the LOP in step 2b.

3.2. Selection of Free Parameters

As seen above, our computational framework for mesh generation has three free parameters, namely, 1) the insertion order insOrder, 2) the triangulation optimality criterion insOptCriterion, and 3) the method fcaMethod used for final connectivity adjustment. For the reasons presented earlier, we advocate choosing insOrder as randomized order. In what follows, we study the effects of making various choices for the two remaining parameters (namely, insOptCriterion and fcaMethod). Based on this analysis, we ultimately recommend two particular choices for these parameters, leading to our two proposed mesh-generation methods.

Test Data. Shortly, we will have the need to present some experimental results obtained with various test images. So, before proceeding further, a brief digression is necessary in order to introduce the test images that we employed. In our work, we have used 40 images, taken mostly from standard test sets such as [41], [42], and [43]. For the most part, the results that we present herein focus on the representative subset of these images listed in Table 1. This particular subset was chosen to contain a variety of image types (i.e., photographic, medical, and computer-generated imagery).

[Table 1 about here.]

Triangulation Optimality Criterion During Point Insertion. To begin, we study how the choice of triangulation optimality criterion insOptCriterion in step 2b of our framework affects mesh quality. Since the best choice of optimality criterion might possibly be dependent on whether the final-connectivity-adjustment method fcaMethod is chosen as LOP or LLOP, we treat these two cases separately. For fcaMethod being chosen as each of LOP and LLOP, we proceeded as follows. For each of the 40 images in our test set and five sampling densities per image (for a total of $40 \cdot 5 = 200$ test cases), we generated a mesh using each of the twelve choices for insOptCriterion under consideration, and measured the resulting approximation error in terms of PSNR. In each of the test cases, the results obtained with the twelve methods were ranked from 1 (best) to 12 (worst). Then, the average and standard deviation of these ranks were computed across each sampling density as well as overall. These ranking results are given in Tables 2(b) and 3(b) for the cases of fcaMethod being chosen as LOP and LLOP, respectively. Individual results for three specific images (namely, the ones listed in Table 1) are provided in Tables 2(a) and 3(a) for fcaMethod being chosen as LOP and LLOP, respectively. In each of these tables, the best result in each row is shown in bold font.

First, let us examine the results for the case that fcaMethod is chosen as LOP. From the ranking results in Table 2(b), we can make several observations: 1) the ELSE criterion is the clear winner (with an overall rank of 1.16), followed by the MASE and JNDSE criteria (with overall ranks of 2.50 and 2.85, respectively); 2) the MASE criterion yields better results than the JNDSE criterion, except at high sampling densities where the two criteria are comparable; and 3) the worst performers are the SE and DP criteria (with overall ranks of 10.88 and 11.74, respectively). Observation 3 supports our earlier claim that the SE criterion leads to extremely poor results (when used during point insertion). To add to observation 1, it is worth noting that a more detailed examination of the results shows that the ELSE criterion performs best and second best in 187/200 (94%) and 4/200 (2%) of the test cases, respectively. This observation is in agreement with the fact that the standard deviations for the rankings for the ELSE criterion are quite small (e.g., 0.89 or less). For that matter, most of the standard deviations in the table are relatively small, indicating that the actual ranking results tend to be reasonably close to the average rank. The results for the individual test cases, shown in Table 2(a), are consistent with the ranking results. For example, the ELSE criterion is the best, outperforming the second and third best criteria, MASE and JNDSE, in all 15 test cases by 0.01 to 1.77 dB and 0.04 to 4.15 dB, respectively. Moreover, the MASE criterion outperforms the JNDSE criterion in 12/15

of the test cases by 0.01 to 2.38 dB. In the preceding results, PSNR was found to correlate reasonably well with subjective quality. It is worthwhile to note that the two best performing criteria ELSE and MASE are newly proposed herein. This shows that our ELSE and MASE criteria, especially the former, make an important contribution beyond well-known criteria from the existing literature.

[Table 2 about here.]

Now, let us consider the results for the case that fcaMethod is chosen as LLOP. As we will see momentarily, the trends in this case are, for the most part, similar to those for the case just studied above. Examining Table 3(b), we observe that: 1) the ELSE criteria is the clear winner (with an overall rank of 1.44) followed by the MASE and JNDSE criteria (with overall ranks of 2.41 and 3.46, respectively); and 2) the SE and DP criteria are the worst performers (with overall ranks of 11.31 and 11.47, respectively). To add to observation 1, a more detailed analysis of the results shows the ELSE criterion to perform best and second best 166/200 (83%) and 12/200 (6%) of the test cases, respectively. This observation is in agreement with the fact that the standard deviations for the rankings for the ELSE criterion are quite small (e.g., 1.19 in the overall case). For that matter, most of the standard deviations in the table are relatively small, indicating that the actual ranking results tend to be reasonably close to the average rank. Compared to the case when fcaMethod is chosen as LOP, we observe that the MASE criterion outperforms the JNDSE criterion even more consistently (i.e., the two criteria differ more in terms of their overall rankings). The results for individual test cases shown in Table 3(a) are consistent with the preceding ranking results. For example, the ELSE criterion is the best, outperforming the second and third best criteria, MASE and JNDSE, in 13/15 of the test cases by 0.01 to 2.22 dB and 0.01 to 2.61 dB, respectively, and the MASE criterion outperforms the JNDSE criterion in 12/15 of the test cases by 0.01 to 0.84 dB. Again, in the preceding results, PSNR was found to correlate reasonably well with subjective quality.

[Table 3 about here.]

As the above experimental results demonstrate, regardless of whether the final-connectivity-adjustment method fcaMethod is chosen as LOP or LLOP, the best performance in terms of approximation quality is obtained by choosing the triangulation optimality criterion insOptCriterion as ELSE. Therefore, we advocate this particular choice for insOptCriterion in our framework.

In the experimental results above, we also saw that, regardless of whether the final-connectivity-adjustment method fcaMethod is chosen as LOP or LLOP, selecting the triangulation optimality criterion insOptCriterion as SE leads to meshes of extremely poor quality. Earlier, we indicated that this behavior is due to an interplay between point insertion and the SE criterion, which leads to triangulations with many poorly-chosen sliver triangles. To illustrate this phenomenon, we present two examples, one for the parameter fcaMethod being chosen as each of LOP and LLOP. For consistency, the examples are taken from the results presented earlier in Tables 2 and 3, and correspond to the lena image at a sampling density of 2%. For the parameter fcaMethod being chosen as each of LOP and LLOP, the results obtained are shown in Figures 3 and 4, respectively. Each figure shows part of the image approximation and the corresponding image-domain triangulation obtained when insOptCriterion is chosen as SE. For comparison purposes, the result obtained with the ELSE criterion (which performs very well) is also shown. First, let us consider the example in Figure 3. Examining Figure 3(b), we can see that the image-domain triangulation obtained with the SE criterion has a large number of poorly-chosen sliver triangles, which leads to very high error in the corresponding image approximation in Figure 3(a). In contrast, viewing Figures 3(c) and (d), we observe that the ELSE criterion does not suffer from this problem. Now, moving our attention to the second example in Figure 4, we can see that a similar pattern of behavior is obtained as in the first example. Again, the SE criterion yields a triangulation with many poorly-chosen sliver triangles, which severely degrades approximation quality.

[Figure 3 about here.]

[Figure 4 about here.]

As for why the SE criterion typically yields triangulations with many poorly-chosen sliver triangles, this can be attributed to the combination of two factors. First, the SE criterion does not explicitly consider triangle shape and, therefore, does not have any direct mechanism for preventing the creation of bad sliver triangles or eliminating such triangles once they are present. Second, the SE criterion is also unable to account for triangle shape in an indirect manner, due to the shortsightedness of the LOP and LLOP. (The shortsightedness of the LOP and LLOP follows from the fact that a decision made at any given step in each of these algorithms considers the impact of that decision only in the current step, not in *all subsequent* steps.) In practice, the above two factors conspire to produce a pattern of behavior with the SE criterion that

resembles the following. When a new point is inserted in the triangulation, a sliver triangle will sometimes result. In such a case, since the SE criterion does not directly consider triangle shape, the SE criterion will often be unable to eliminate the sliver triangle. Thus, as more points are inserted in the triangulation, the number of sliver triangles tends to grow significantly. In turn, as the number of sliver triangles grows, the number of unflippable edges also tends to increase. This leads to sliver triangles tending to have fewer flippable edges (on average). This, in turn, makes it more difficult to eliminate sliver triangles, once present. In this manner, a very large number of sliver triangles are obtained. Because the number of sliver triangles produced is so abnormally large, it is not surprising that the number of such triangles that are poorly chosen is also high.

In the experimental results above, we saw that the ELSE and MASE criteria perform best in terms of mesh quality. This excellent performance is made possible by the fact that each of these two criteria has a direct dependence on *both* triangle shape and squared error. The dependence on squared error is critical for achieving high mesh quality, while the dependence on triangle shape is important for avoiding large numbers of poorly-chosen sliver triangles. In the case of the ELSE criterion, triangle shape is implicitly considered by the criterion's dependence on edge length, which penalizes longer edges. In the case of the MASE criterion, triangle shape is considered by the criterion's dependence on minimum interior angle, which penalizes smaller interior angles. By accounting for triangle shape, the ELSE and MASE criteria are able to avoid the bad-sliver problem that plagues the SE criterion.

Method for Final Connectivity Adjustment. Next, we study how the choice of the final-connectivity-adjustment method fcaMethod in step 3 of our framework (which can be either LOP or LLOP) affects mesh quality. To do this, we fix the insOptCriterion parameter to be ELSE and proceed as follows. For each of the 40 images in our test set and five sampling densities per image (for a total of $40 \cdot 5 = 200$ test cases), we generated a mesh using each of the two choices for fcaMethod under consideration (namely, LOP and LLOP), and measured the resulting approximation error in terms of PSNR. In all of these 200 test cases, the LLOP outperformed the LOP by a margin of 0.09 to 2.30 dB, with the average margin being 0.56 dB. Individual results for three images (namely, the images listed in Table 1) are given in Table 4. Examining this table, we see that the LLOP outperforms the LOP in all cases by a margin of 0.30 to 0.93 dB. Although we have only shown results for one choice of the fixed parameter insOptCriterion (i.e., ELSE), we found

similar results with other choices. Thus, from above, we conclude that choosing the parameter fcaMethod as LLOP (as opposed to LOP) yields higher mesh quality. This said, however, we must point out that this choice entails a tradeoff in terms of computational cost. As noted earlier (in Section 2), the LLOP has a higher computational cost than the LOP. For example, for the test case of the lena image at a sampling density of 2%, we found the LOP and LLOP to have computation times of about 1.40 seconds and 2.30 seconds, respectively. More generally, we have found the LLOP to typically require a computation time that is about 1.4 to 1.7 times that of the LOP. Thus, the best choice for the parameter fcaMethod depends on the most appropriate tradeoff between mesh quality and computational cost for the application at hand.

[Table 4 about here.]

3.3. Proposed Methods

Above, we have considered how various choices for the free parameters in our computational framework for mesh generation (namely, the insertion order, triangulation optimality criterion, and final-connectivity-adjustment method) affect mesh quality. This led us to conclude that the triangulation optimality criterion insOptCriterion and the insertion order insOrder are best chosen as ELSE and randomized order, respectively. Whether the final-connectivity-adjustment method fcaMethod should be chosen as LOP or LLOP is less clear cut, due to a tradeoff between mesh quality and computational cost. As a result, we chose to propose two methods, known as MED1 and MED2, where the first method has a lower computational cost relative to the second. The MED1 and MED2 methods both employ the best choices for insOptCriterion and insOrder as identified above (i.e., ELSE and randomized order, respectively). For the final-connectivity-adjustment method fcaMethod, however, the MED1 method uses the LOP (which has lower computational cost), while the MED2 method uses the LLOP (which has higher computational cost). In passing, we note that the CCCG method (from our conference paper [35] discussed earlier in Section 1) is equivalent to using our framework with the insOptCriterion, insOrder, and fcaMethod parameters chosen as JNDSE, randomized order, and LOP, respectively. As we will see later, due to its use of the less effective JNDSE triangulation optimality criterion, our CCCG method produces poorer quality meshes than our MED1 and MED2 methods proposed herein.

4. Evaluation of Proposed Methods

Having introduced our MED1 and MED2 mesh-generation methods, we now compare their performance to that of the ED scheme in terms of mesh quality. To demonstrate the our MED1 and MED2 methods make a significant contribution beyond the CCCG scheme from our conference paper [35] (discussed in Section 1), we also consider the CCCG method in this evaluation. In addition, we make a few comments regarding the computational cost of our proposed methods. The software implementations of the methods used in this evaluation were developed by the authors of this paper and written in C++. For test data, we employ the same set of 40 images described earlier in Section 3.2 (under the heading "Test Data").

Mesh Quality. For all 40 images in our test set and five sampling densities per image (for a total of $40 \cdot 5 = 200$ test cases), we used each of the various methods under consideration to generate a mesh, and then measured the resulting approximation error in terms of PSNR. Individual results for three specific images (namely, the images listed in Table 1) are given in Table 5.

To begin, we compare the MED1 and MED2 methods to the CCCG scheme. Examining the results for the individual test cases in Table 5, we see that the MED1 and MED2 methods both outperform the CCCG scheme in all 15 test cases by margins of 0.04 to 4.15 dB and 0.36 to 4.83 dB, respectively. Next, we comment on the full set of results for all 200 test cases (i.e., 40 images with five sampling densities per image). In the full set of results, we found that the MED1 and MED2 methods outperform the CCCG scheme in 191/200 (i.e., 96%) and 198/200 (i.e., 99%) of the test cases, respectively. Thus, our MED1 and MED2 methods are clearly superior to the CCCG scheme. This demonstrates that the MED1 and MED2 methods proposed herein represent a substantial contribution beyond the CCCG scheme from our conference paper [35]. Since the MED1 and MED2 methods are clearly superior to the CCCG scheme, we will not consider the CCCG scheme further in our evaluation.

Now, we compare the MED1 and MED2 methods to the ED scheme. Examining the results for the individual test cases in Table 5, we see that the MED1 and MED2 methods both outperform that ED scheme in all 15 test cases, by margins of 1.94 to 8.46 dB and 2.24 to 9.14 dB, respectively. Next, we consider the full set of results taken across all 200 test cases (i.e., 40 images with five sampling densities per image). In the full set of results, we found that the MED1 and MED2 methods both yield higher quality meshes than the ED scheme in all 200 test cases. More specifically, the MED1 method outperformed the ED scheme by

a margin of 1.93 to 8.46 dB with an average margin of 3.26 dB, while the MED2 method outperformed the ED scheme by a margin of 2.23 to 9.14 dB with an average margin of 3.81 dB. Thus, the MED1 and MED2 methods clearly yield meshes of very substantially higher quality, relative to the ED method.

Next, we compare the performance of the MED1 and MED2 methods. Examining the results from the individual test cases in Table 5, we can see that the MED2 method beats the MED1 method in all 15 test cases by a margin of 0.30 to 0.93 dB. In the full set of results, we found that the MED2 method yields higher quality meshes than the MED1 method in all 200 test cases, by a margin of 0.09 to 2.30 dB with an average margin of 0.56 dB. Therefore, the MED2 method consistently yields meshes of higher quality than the MED1 method. This behavior is due to the MED2 method using the more effective LLOP (instead of the LOP) for final connectivity adjustment.

[Table 5 about here.]

In the above results, PSNR was found to correlate reasonably well with subjective image quality. For the benefit of the reader, however, we include an illustrative example in what follows. For one of the test cases in Table 5 (namely, the lena image at a sampling density of 2%), part of the image approximation and the corresponding image-domain triangulation obtained for each of the various methods is shown in Figure 5. Examining this figure, we can see that the image approximations produced by our MED1 and MED2 methods (in Figures 5(a) and (b), respectively) are clearly of much higher quality than the one produced by the ED scheme (in Figure 5(c)), with image details such as image edges/contours being much better preserved in the MED1 and MED2 cases. In order to more clearly highlight some of the more subtle differences between the results for our MED1 and MED2 methods, we show (for the same test case) the results for a smaller region of interest under greater magnification in Figure 6. By carefully comparing the image approximations for our MED1 and MED2 methods in Figures 6(a) and (b), respectively, we can see that there are a few places where image details (such as edges and contours) are slightly better preserved by our MED2 method than our MED1 scheme, one example being the (image) edge along the top of the hat. The improved performance in the MED2 case is largely due to triangulation edges being better aligned with image edges/contours. So, in terms of subjective quality, our MED1 and MED2 methods are both vastly superior to the ED scheme, with our MED2 method yielding slightly better quality than our MED1 scheme.

[Figure 5 about here.]

[Figure 6 about here.]

Computational Cost. Next, we briefly consider the computational costs of our MED1 and MED2 methods. For the purposes of making timing measurements, we employed very modest hardware, namely an eight-year-old notebook computer with a 2.00 GHz Intel Core2 Duo T7250 CPU and 1.0 GB of RAM. On this machine, our MED1 and MED2 methods typically require only a few seconds of computation time for images like lena from Table 1. In particular, for the lena image and sampling densities in the range 0.5 to 4% (as in Table 5), the MED1 and MED2 methods required 0.95 to 2.06 seconds and 1.38 to 3.30 seconds, respectively. For cases like this, our MED1 and MED2 methods typically increase the computation time relative to the ED scheme by only 0.3 to 1.1 seconds and 0.7 to 2.4 seconds, respectively. In absolute terms, this incremental cost is very small when one considers the very substantial improvement in mesh quality obtained with our methods. Furthermore, when viewed in the broader context of the many mesh-generation techniques proposed to date in the literature, our MED1 and MED2 methods are quite low in terms of computational cost. For example, some other methods, which are based on techniques such as simulated annealing or simplification of very large meshes, can easily require computation times on the order of minutes or more.

Typically, the computation time for our MED2 method was found to be about 1.4 to 1.7 times that of our MED1 scheme. So, our MED2 method is more computationally expensive, with this higher cost coming from the use of the LLOP (instead of the LOP) during final connectivity adjustment. As we saw earlier, our MED2 method yields higher quality meshes than our MED1 scheme. So, whether our MED1 or MED2 method is more attractive for a particular application, depends on computational constraints. In applications that are sensitive even to small increases in computational cost, our MED1 method would be more appropriate, while our MED2 method would be preferred otherwise.

5. Conclusions

In this paper, we have proposed a computational framework for mesh generation that modifies the ED method to use DDTs in conjunction with the LOP. By using DDTs in conjunction with the LOP (instead of

Delaunay triangulations), triangulation connectivity can be chosen optimally so as to minimize approximation error. Using our computational framework, we derived two specific mesh-generation methods known as MED1 and MED2. Through experimental results, our MED1 and MED2 methods were shown to produce image approximations of much higher quality than the ED method, both in terms of PSNR and subjective quality, at a relatively modest computational cost. In particular, our MED1 and MED2 methods were shown to outperform the ED method by margins of 3.26 to 3.81 dB on average. Our two methods allow different tradeoffs to be made between computational cost and approximation quality, allowing our proposed mesh-generation approach to be useful over a broader range of applications with differing computational constraints. As part of our work, we proposed two novel optimality criteria to be used in conjunction with the LOP, namely the ELSE and MASE criteria. These two criteria were shown to outperform other well known criteria from the literature. Of our two newly proposed criteria, the ELSE criterion was found to perform best and was used as a key component of our MED1 and MED2 methods. Since the LOP is used in many different applications and MSE (as in (1)) is a frequently employed error metric, the ELSE and MASE criteria proposed herein have the potential to be useful in a much broader range of contexts than the particular mesh-generation methods proposed herein. By allowing higher quality meshes to be generated at relatively low computational cost, the MED1 and MED2 methods are of great utility to the many applications that employ mesh models of images. Furthermore, our new optimality criteria, ELSE and MASE, can be exploited by future mesh-generation schemes that employ the LOP in order to achieve improved results.

Lastly, we make one further comment regarding the applicability of our methods. In this manuscript, we have focused our attention on the generation of mesh models of luminance images (i.e., images for which the image function value corresponds to light intensity). It is worth noting, however, that our methods can be used to generate mesh models of other types of bivariate functions. Some of these other types of functions include: 1) digital elevation maps, which are employed in geographic information systems; and 2) range images, which are used in robotics, gaming, and other applications. Since our mesh-generation methods produce high-quality meshes while not requiring excessive amounts of computation time (like the minutes or tens of minutes needed by some schemes), our methods may have advantages over some of the previously-proposed approaches for these other types of data sets. Thus, our proposed mesh-generation methods are not only useful for traditional image-processing applications, but are of potential benefit in

other areas as well.

Acknowledgments

This work was supported, in part, by the Natural Sciences and Engineering Research Council of Canada.

References

[1] S. A. Coleman, B. W. Scotney, M. G. Herron, Image feature detection on content-based meshes, in: Proc. of IEEE International Conference on Image Processing, Vol. 1, 2002, pp. 844–847.

[2] M. Petrou, R. Piroddi, A. Talebpour, Texture recognition from sparsely and irregularly sampled data, Computer Vision and Image Understanding 102 (2006) 95–104.

[3] M. Sarkis, K. Diepold, A fast solution to the approximation of 3-D scattered point data from stereo images using triangular meshes, in: Proc. of IEEE-RAS International Conference on Humanoid Robots, Pittsburgh, PA, USA, 2007, pp. 235–241.

[4] J. G. Brankov, Y. Yang, N. P. Galatsanos, Image restoration using content-adaptive mesh modeling, in: Proc. of IEEE International Conference on Image Processing, Vol. 2, 2003, pp. 997–1000.

[5] J. G. Brankov, Y. Yang, M. N. Wernick, Tomographic image reconstruction based on a content-adaptive mesh model, IEEE Trans. on Medical Imaging 23 (2) (2004) 202–212.

[6] M. A. Garcia, B. X. Vintimilla, Acceleration of filtering and enhancement operations through geometric processing of gray-level images, in: Proc. of IEEE International Conference on Image Processing, Vol. 1, Vancouver, BC, Canada, 2000, pp. 97–100.

[7] D. Su, P. Willis, Demosaicing of colour images using pixel level data-dependent triangulation, in: Proc. of the Theory and Practice of Computer Graphics, 2003, pp. 16–23.

[8] D. Su, P. Willis, Image interpolation by pixel-level data-dependent triangulation, Computer Graphics Forum 23 (2) (2004) 189–201.

[9] M. D. Adams, Progressive lossy-to-lossless coding of arbitrarily-sampled image data using the modified scattered data coding method, in: Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, 2009, pp. 1017–1020.

[10] G. Ramponi, S. Carrato, An adaptive irregular sampling algorithm and its application to image coding, Image and Vision Computing 19 (2001) 451–460.

[11] P. Lechat, H. Sanson, L. Labelle, Image approximation by minimization of a geometric distance applied to a 3D finite elements based model, in: Proc. of IEEE International Conference on Image Processing, Vol. 2, 1997, pp. 724–727.

[12] Y. Wang, O. Lee, A. Vetro, Use of two-dimensional deformable mesh structures for video coding, part II–the analysis problem and a region-based coder employing an active mesh representation, IEEE Trans. on Circuits and Systems for Video Technology 6 (6) (1996) 647–659.

[13] F. Davoine, M. Antonini, J.-M. Chassery, M. Barlaud, Fractal image compression based on Delaunay triangulation and vector quantization, IEEE Trans. on Image Processing 5 (2) (1996) 338–346.

[14] K.-L. Hung, C.-C. Chang, New irregular sampling coding method for transmitting images progressively, IEE Proceedings Vision, Image and Signal Processing 150 (1) (2003) 44–50.

[15] M. D. Adams, An efficient progressive coding method for arbitrarily-sampled image data, IEEE Signal Processing Letters 15 (2008) 629–632.

[16] I. Amidror, Scattered data interpolation methods for electronic imaging systems: a survey, Journal of Electronic Imaging 11 (2) (2002) 157–176.

[17] R. Franke, G. M. Nielson, Scattered data interpolation and applications: A tutorial and survey, in: Geometric Modeling: Methods and Applications, Springer-Verlag, 1991, pp. 131–160.

[18] M. Garland, P. S. Heckbert, Fast polygonal approximation of terrains and height fields, Tech. Rep. CMU-CS-95-181, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA (Sep. 1995).

[19] Y. Yang, M. N. Wernick, J. G. Brankov, A fast approach for accurate content-adaptive mesh generation, IEEE Trans. on Image Processing 12 (8) (2003) 866–881.

[20] M. D. Adams, A flexible content-adaptive mesh-generation strategy for image representation, IEEE Trans. on Image Processing 20 (9) (2011) 2414–2427.

[21] M. D. Adams, A highly-effective incremental/decremental Delaunay mesh-generation strategy for image representation, Signal Processing 93 (4) (2013) 749–764.

[22] X. Tu, M. D. Adams, Image representation using triangle meshes with explicit discontinuities, in: Proc. of IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, BC, Canada, 2011, pp. 97–101.

[23] N. Dyn, D. Levin, S. Rippa, Data dependent triangulations for piecewise liner interpolations, IMA Journal of Numerical Analysis 10 (1) (1990) 137–154.

[24] N. Dyn, D. Levin, S. Rippa, Algorithms for the construction of data dependent triangulations, in: J. C. Mason, M. G. Cox (Eds.), Algorithms for Approximation II, Chapman and Hall, London, 1990, pp. 185–192.

[25] E. Quak, L. L. Schumaker, Lease squares fitting by linear splines on data dependent triangulations, in: Curves and surfaces, Academic Press Professional, Inc., 1991, pp. 387–390.

[26] S. Rippa, Adaptive approximation by piecewise linear polynomials on triangulations of subsets of scattered data, SIAM Journal on Scientific and Statistical Computing 13 (5) (1992) 1123–1141.

[27] N. Dyn, D. Levin, S. Rippa, Boundary correction for piecewise linear interpolation defined over data-dependent triangulations, Journal of Computational and Applied Mathematics 39 (1992) 179–192.

[28] N. Dyn, Data-dependent triangulations for scattered data interpolation and finite element approximation, Applied Numerical Mathematics 12 (1993) 89–105.

[29] X. Yu, B. S. Morse, T. W. Sederberg, Image reconstruction using data-dependent triangulation, IEEE Computer Graphics and Applications 21 (3) (2001) 62–68.

[30] P. Li, M. D. Adams, A tuned mesh-generation strategy for image representation based on data-dependent triangulation, IEEE

Trans. on Image Processing 22 (5) (2013) 2004–2018.

[31] S. Bougleux, G. Peyre, L. D. Cohen, Image compression with anisotropic triangulations, in: Proc. of IEEE International Conference on Computer Vision, 2009, pp. 2343–2348.

[32] B. Delaunay, Sur la sphere vide, Bulletin of the Academy of Sciences of the USSR, Classe des Sciences Mathematiques et Naturelle 7 (6) (1934) 793–800.

[33] C. Dyken, M. S. Floater, Preferred directions for resolving the non-uniqueness of Delaunay triangulations, Computational Geometry—Theory and Applications 34 (2006) 96–101.

[34] C. L. Lawson, Software for C^1 surface interpolation, in: J. R. Rice (Ed.), Mathematical Software III, Academic Press, New York, NY, USA, 1977, pp. 161–194.

[35] X. Ma, M. D. Adams, An improved method for generating triangle-mesh models of images, Halifax, NS, Canada, 2014, available online at `https://projects.cs.dal.ca/cccg2014/proceedings/papers/paper10.pdf`.

[36] P. K. Agarwal, S.Suri, Surface approximation and geometric partitions, Proc. of ACM-SIAM Symposium on Discrete Algorithms (1994) 24–33.

[37] R. W. Floyd, L. Steinberg, An adaptive algorithm for spatial greyscale, Proc. of the Society for Information Display 17 (2) (1976) 75–77.

[38] C. L. Lawson, Transforming triangulations, in: Discrete Math., Vol. 3, 1972, pp. 365–372.

[39] L. Alboul, G. Kloosterman, C. Traas, R. van Damme, Best data-dependent triangulations, Journal of Computational and Applied Mathematics 119 (2000) 1–12.

[40] K. Fleischer, D. Salesin, Accurate polygon scan conversion using half-open intervals, in: Graphics Gems III, 1992, pp. 362–365.

[41] JPEG-2000 test images, ISO/IEC JTC 1/SC 29/WG 1N 545 (July 1997).

[42] USC-SIPI image database, `http://sipi.usc.edu/database` (2014).

[43] Kodak lossless true color image suite, `http://r0k.us/graphics/kodak` (2014).

List of Figures

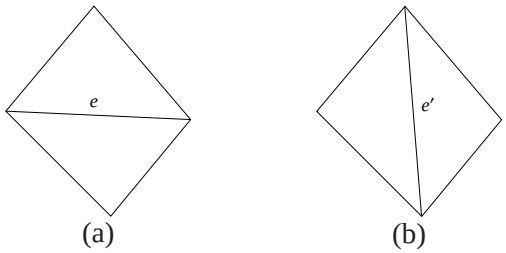

Figure 1: An edge flip. (a) Part of the triangulation containing a flippable edge e. (b) The same part of the triangulation after e has been flipped to yield the new edge e'.

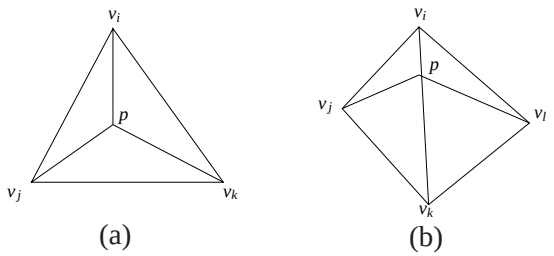

Figure 2: Point insertion examples. Part of a triangulation showing how the new vertex p is inserted (a) inside a triangle $v_i v_j v_k$ and (b) on an edge $v_i v_k$.

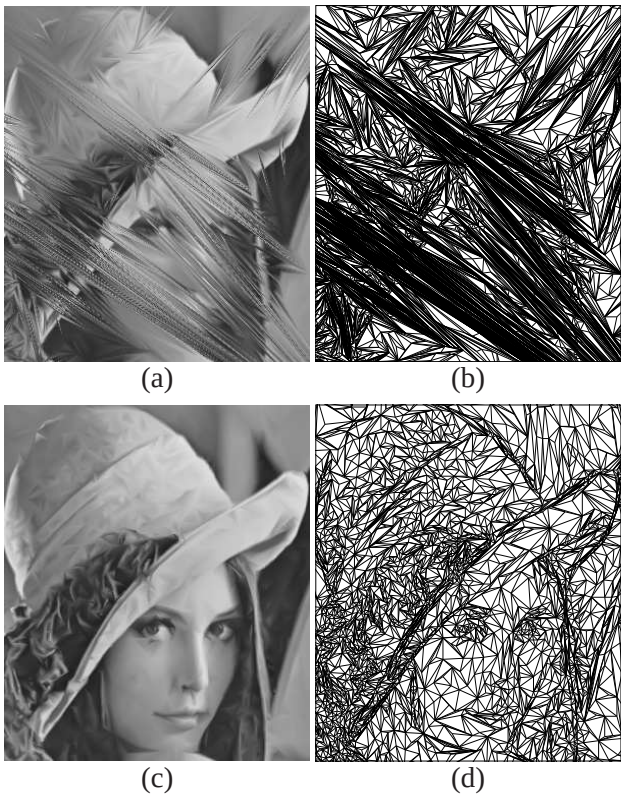

Figure 3: Comparison of the mesh quality obtained for the lena image at a sampling density of 2% in the case that the final-connectivity-adjustment method fcaMethod is LOP. Part of the image approximation obtained when the optimality criteria insOptCriterion is chosen as each of (a) SE (20.72 dB) and (c) ELSE (30.14 dB), and (b) and (d) the corresponding triangulations.

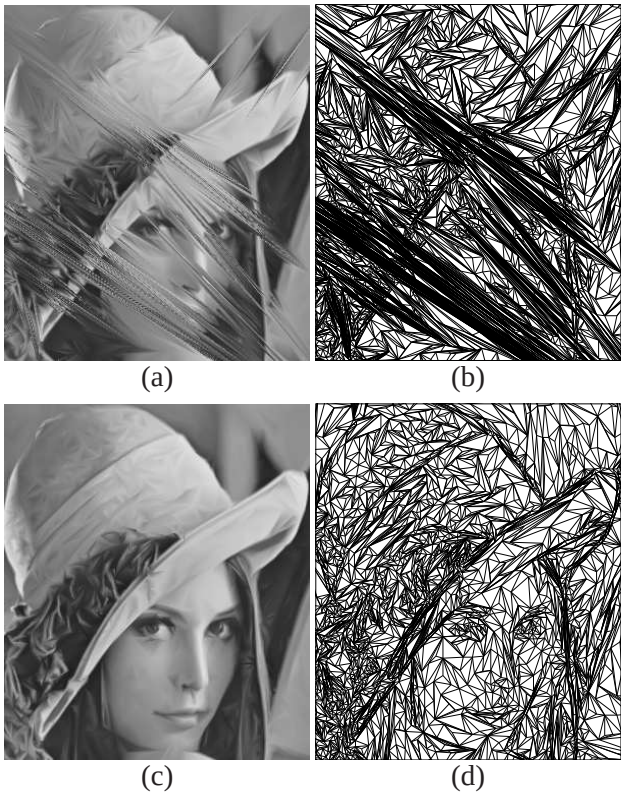

Figure 4: Comparison of the mesh quality obtained for the lena image at a sampling density of 2% in the case that the final-connectivity-adjustment method fcaMethod is LLOP. Part of the image approximation obtained when the optimality criteria insOptCriterion is chosen as each of (a) SE (22.64 dB) and (c) ELSE (30.68 dB), and (b) and (d) the corresponding triangulations.

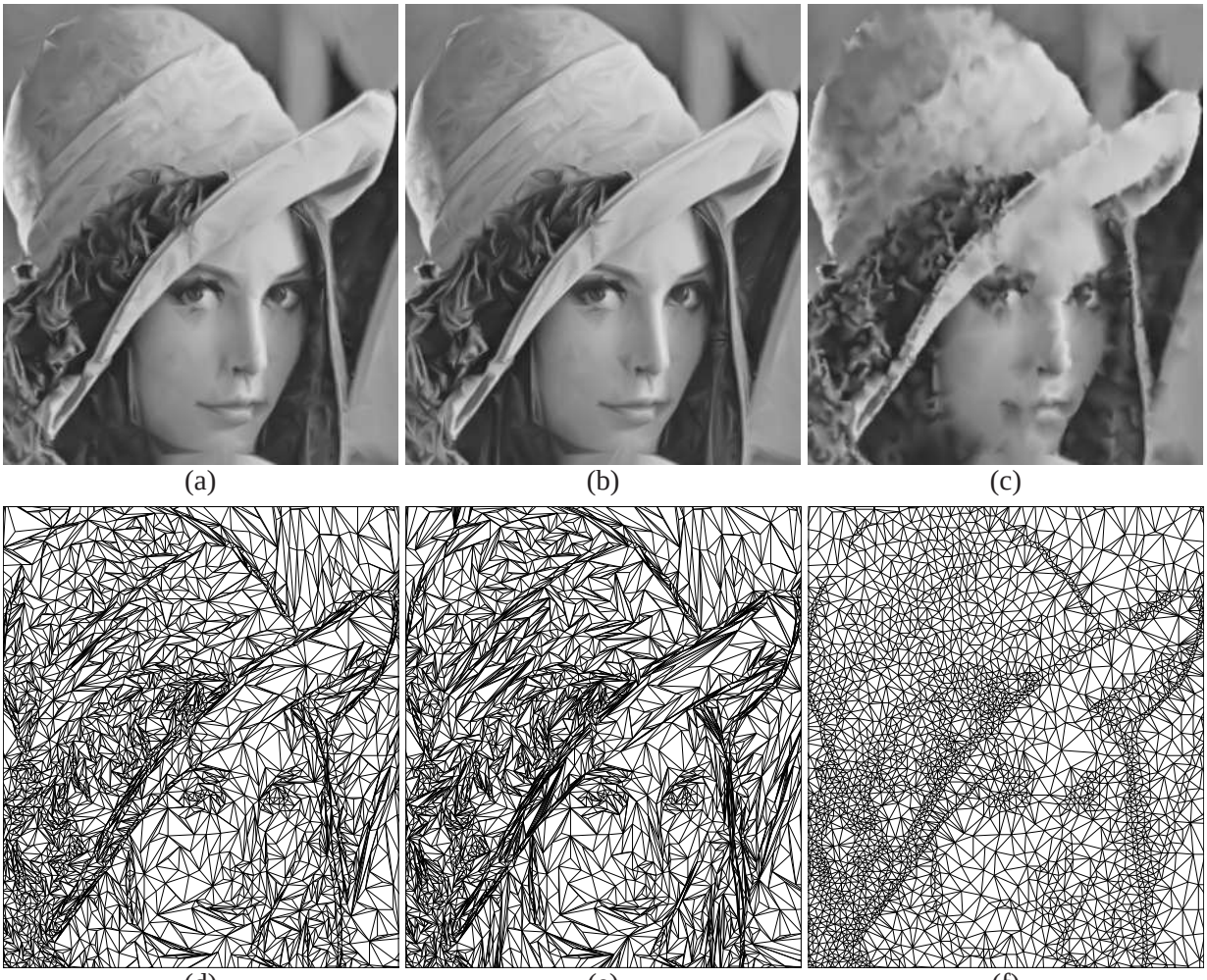

Figure 5: Part of the image approximation obtained for the lena image at a sampling density of 2% with each of the (a) MED1 (30.14 dB), (b) MED2 (30.68 dB), and (c) ED (26.38 dB) methods, and (d), (e), and (f) the corresponding triangulations.

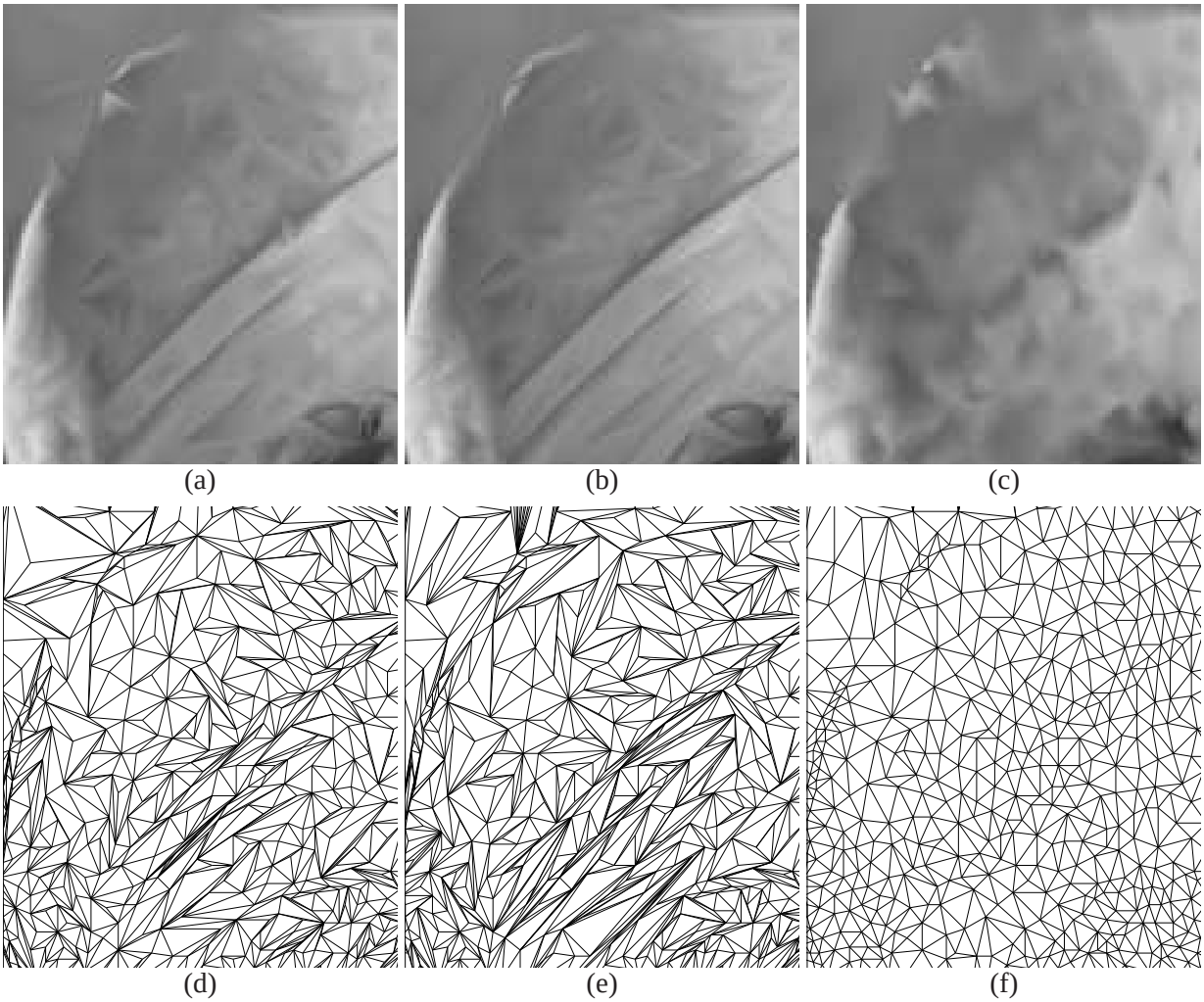

Figure 6: Part of the image approximation (under magnification) obtained for the lena image at a sampling density of 2% with each of the (a) MED1 (30.14 dB), (b) MED2 (30.68 dB), and (c) ED (26.38 dB) methods, and (d), (e), and (f) the corresponding triangulations.

List of Tables

Table 1: Test images

Image	Size, Bits/Sample	Description
bull	1024×768, 8	cartoon animal
cr	1744×2048, 10	x-ray [41]
lena	512×512, 8	woman [42]

Table 2: Comparison of the mesh quality obtained with the various choices of triangulation optimality criterion insOptCriterion in the case that fcaMethod is LOP. (a) PSNRs for three specific images. (b) Rankings averaged across 40 images.

(a)

Image	Samp. Density (%)	PSNR (dB)											
		SE	Del.	ABN	JND	DLP	DP	AMC	GHH	SQSE	JNDSE	MASE	ELSE
bull	0.5	24.73	31.44	28.44	30.49	29.72	25.96	31.42	30.22	31.59	31.37	33.75	**35.52**
	1.0	26.89	38.85	31.91	38.34	33.42	30.73	39.00	37.82	38.69	38.78	39.43	**39.99**
	2.0	30.53	42.12	34.87	42.26	39.52	25.93	41.94	41.36	42.36	42.36	42.49	**42.72**
	3.0	31.93	43.42	36.24	43.36	39.35	29.91	43.08	43.41	43.61	43.66	43.83	**43.97**
	4.0	31.23	44.34	34.97	44.17	41.40	31.92	44.07	44.23	44.49	44.53	44.62	**44.72**
cr	0.5	31.19	34.40	30.38	34.45	32.42	30.23	34.22	34.30	34.81	34.84	34.92	**35.11**
	1.0	32.41	36.33	33.01	36.35	34.42	31.16	36.37	36.48	37.13	37.16	37.22	**37.31**
	2.0	33.33	38.68	34.34	38.36	36.33	32.52	38.24	38.75	38.95	39.01	39.00	**39.10**
	3.0	34.12	39.57	34.95	39.32	36.96	33.78	39.17	39.62	39.76	39.82	39.80	**39.87**
	4.0	35.63	40.10	39.29	39.89	37.56	33.54	39.70	40.19	40.31	40.36	40.33	**40.42**
lena	0.5	17.61	21.17	19.22	20.55	19.61	18.07	20.51	21.20	21.75	21.82	21.83	**21.96**
	1.0	21.50	25.21	20.69	24.91	21.86	19.91	24.58	25.30	25.89	25.92	25.94	**26.13**
	2.0	20.72	29.48	24.36	29.09	26.25	21.04	27.67	29.26	29.91	29.99	30.06	**30.14**
	3.0	23.43	31.26	24.62	30.99	27.22	22.34	30.15	31.21	31.58	31.62	31.71	**31.72**
	4.0	23.67	32.39	26.30	32.17	29.13	24.06	31.45	32.47	32.78	32.84	32.87	**32.88**

(b)

Samp. Density (%)	Mean Rank[a]											
	SE	Del.	ABN	JND	DLP	DP	AMC	GHH	SQSE	JNDSE	MASE	ELSE
0.5	10.20	5.83	9.78	8.10	8.90	11.40	7.30	5.13	3.85	3.43	2.83	**1.28**
	(1.42)	(1.30)	(1.47)	(1.67)	(1.62)	(1.22)	(1.68)	(1.73)	(1.22)	(1.66)	(1.53)	(0.89)
1.0	10.95	6.05	10.03	7.60	9.10	11.75	6.95	5.35	3.80	2.98	2.28	**1.18**
	(0.80)	(1.18)	(0.57)	(0.80)	(0.49)	(0.49)	(1.26)	(0.79)	(0.64)	(0.79)	(0.63)	(0.67)
2.0	10.98	5.80	10.00	7.08	9.10	11.90	7.85	5.30	3.75	2.75	2.43	**1.08**
	(0.47)	(0.40)	(0.50)	(0.52)	(0.37)	(0.30)	(0.42)	(0.64)	(0.54)	(0.66)	(0.70)	(0.35)
3.0	11.18	5.90	10.00	6.95	9.03	11.80	8.00	5.13	3.88	2.58	2.43	**1.15**
	(0.44)	(0.37)	(0.22)	(0.22)	(0.16)	(0.40)	(0.00)	(0.46)	(0.40)	(0.67)	(0.54)	(0.69)
4.0	11.10	5.88	10.03	7.03	9.03	11.85	7.98	5.10	3.85	2.53	2.55	**1.10**
	(0.44)	(0.33)	(0.27)	(0.16)	(0.16)	(0.36)	(0.16)	(0.37)	(0.53)	(0.55)	(0.59)	(0.62)
All	10.88	5.89	9.97	7.35	9.03	11.74	7.62	5.20	3.83	2.85	2.50	**1.16**
	(0.88)	(0.84)	(0.76)	(0.97)	(0.79)	(0.67)	(1.05)	(0.94)	(0.72)	(1.01)	(0.90)	(0.67)

[a]The standard deviation is given in parentheses.

Table 3: Comparison of the mesh quality obtained with the various choices of triangulation optimality criterion insOptCriterion in the case that fcaMethod is LLOP. (a) PSNRs for three specific images. (b) Rankings averaged across 40 images.

(a)

Image	Samp. Density (%)	PSNR (dB)											
		SE	Del.	ABN	JND	DLP	DP	AMC	GHH	SQSE	JNDSE	MASE	ELSE
bull	0.5	26.68	34.31	31.70	35.24	31.83	29.98	34.72	33.10	33.75	33.88	34.72	**36.20**
	1.0	30.67	40.41	36.93	40.12	37.30	35.76	40.30	40.01	40.15	40.17	40.56	**42.78**
	2.0	33.73	43.27	39.05	43.37	41.25	31.54	43.39	42.69	43.23	43.23	43.38	**43.51**
	3.0	34.91	44.43	39.85	44.43	42.30	34.84	44.51	44.32	44.39	44.40	44.52	**44.58**
	4.0	33.76	45.20	40.25	45.15	44.34	36.72	45.22	45.05	45.19	45.21	45.29	**45.33**
cr	0.5	32.73	35.88	32.68	35.65	34.05	32.50	35.65	35.76	35.95	35.91	35.92	**36.04**
	1.0	34.13	37.71	35.10	37.18	35.92	33.19	37.55	37.72	37.77	37.78	37.78	**37.82**
	2.0	34.83	39.36	36.51	39.19	37.92	34.76	39.19	39.35	39.40	39.41	39.42	**39.43**
	3.0	35.68	40.14	37.16	40.01	38.59	35.99	39.99	40.12	40.17	40.18	40.18	**40.19**
	4.0	37.10	40.65	38.33	40.53	38.96	35.63	40.52	40.63	40.69	40.70	40.70	**40.72**
lena	0.5	18.80	22.51	20.68	21.76	20.86	22.03	22.03	22.47	22.28	22.43	22.45	**22.53**
	1.0	23.18	26.41	23.22	26.33	23.97	26.17	26.17	26.39	26.48	26.51	26.57	**26.63**
	2.0	22.64	30.55	27.23	30.30	28.77	30.24	30.24	30.08	30.57	30.58	30.60	**30.68**
	3.0	26.44	32.18	27.89	32.13	29.35	31.79	31.79	32.10	32.27	32.26	**32.29**	32.26
	4.0	25.94	33.38	29.82	33.23	31.03	33.09	33.09	33.30	33.39	33.42	**33.44**	33.42

(b)

Samp. Density (%)	Mean Rank[a]											
	SE	Del.	ABN	JND	DLP	DP	AMC	GHH	SQSE	JNDSE	MASE	ELSE
0.5	10.98	4.40	9.68	7.65	8.95	11.00	6.15	5.60	4.60	4.33	2.98	**1.70**
	(1.33)	(1.69)	(1.13)	(1.75)	(1.75)	(1.76)	(2.34)	(2.07)	(2.00)	(1.89)	(1.37)	(1.36)
1.0	11.48	4.55	9.88	7.78	9.00	11.43	6.60	5.78	3.90	3.60	2.33	**1.70**
	(0.59)	(1.18)	(0.60)	(0.79)	(0.89)	(0.54)	(1.61)	(1.25)	(1.04)	(1.43)	(0.96)	(1.65)
2.0	11.33	5.05	9.83	7.25	9.18	11.68	7.28	6.05	3.58	3.18	2.48	**1.15**
	(0.47)	(0.77)	(0.38)	(0.83)	(0.38)	(0.47)	(1.07)	(0.77)	(1.20)	(0.77)	(0.77)	(0.65)
3.0	11.38	5.05	9.93	7.05	9.08	11.63	7.68	5.98	3.70	3.15	2.10	**1.30**
	(0.48)	(0.38)	(0.26)	(0.67)	(0.26)	(0.48)	(0.85)	(0.57)	(0.81)	(0.91)	(0.54)	(1.03)
4.0	11.38	5.00	9.98	7.03	9.03	11.63	7.78	5.95	3.68	3.05	2.18	**1.35**
	(0.48)	(0.55)	(0.16)	(0.47)	(0.16)	(0.48)	(0.82)	(0.63)	(0.88)	(0.74)	(0.83)	(0.91)
All	11.31	4.81	9.86	7.35	9.05	11.47	7.10	5.87	3.89	3.46	2.41	**1.44**
	(0.77)	(1.06)	(0.62)	(1.05)	(0.91)	(0.94)	(1.59)	(1.21)	(1.31)	(1.32)	(0.99)	(1.19)

[a]The standard deviation is given in parentheses.

Table 4: Comparison of the mesh quality obtained with each of the two choices for the fcaMethod parameter

Image	Samp. Density (%)	PSNR (dB)	
		LOP	LLOP
bull	0.5	35.52	**36.20**
	1.0	29.99	**42.78**
	2.0	42.72	**43.51**
	3.0	43.97	**44.58**
	4.0	44.72	**45.33**
cr	0.5	35.11	**36.04**
	1.0	37.31	**37.82**
	2.0	39.10	**39.43**
	3.0	39.87	**40.19**
	4.0	40.42	**40.72**
lena	0.5	21.96	**22.53**
	1.0	26.13	**26.63**
	2.0	30.14	**30.68**
	3.0	31.72	**32.26**
	4.0	32.88	**33.42**

Table 5: Comparison of the mesh quality obtained with the various methods

Image	Samp. Density (%)	PSNR (dB)			
		MED1	MED2	ED	CCCG
bull	0.5	35.52	36.20	27.06	31.37
	1.0	39.99	40.78	34.46	38.78
	2.0	42.72	43.51	38.59	42.36
	3.0	43.97	44.58	40.47	43.66
	4.0	44.72	45.33	41.60	44.53
cr	0.5	35.11	36.04	31.96	34.84
	1.0	37.31	37.82	33.84	37.16
	2.0	39.10	39.43	35.72	39.01
	3.0	39.87	40.19	37.63	39.82
	4.0	40.42	40.72	38.48	40.36
lena	0.5	21.96	22.53	17.76	21.82
	1.0	26.13	26.63	21.50	25.92
	2.0	30.14	30.68	26.38	29.99
	3.0	31.72	32.26	28.50	31.62
	4.0	32.88	33.42	29.83	32.84