
***N*-dimensional Data-Dependent Reconstruction Using Topological Changes**

Z. Tóth¹, I. Viola², A. Ferko¹, and M. E. Gröller²

¹ Faculty of Mathematics, Physics, and Informatics, Comenius University, Bratislava, Slovakia {toth|ferko}@sccg.sk

² Institute of Computer Graphics and Algorithms, Vienna University of Technology, Austria {viola|meister}@cg.tuwien.ac.at

Abstract

We introduce a new concept for a geometrically based feature preserving reconstruction technique of n -dimensional scattered data. Our goal is to generate an n -dimensional triangulation, which preserves the high frequency regions via local topology changes. It is the generalization of a 2D reconstruction approach based on data-dependent triangulation and Lawson's optimization procedure. The definition of the mathematic optimum of the reconstruction is given. We discuss an original cost function and a generalization of known functions for the n -dimensional case.

1 Introduction

The continuous reconstruction from discretely sampled data is an important part of data processing. Reconstruction is necessary to determine values at arbitrary positions, not just those at which the data sample is available. Such discrete data sets can be acquired by digital photography in 2D or computed tomography (CT) and magnetic resonance imaging (MRI) in 3D. Another way of data acquisition is mathematical simulation of certain phenomena using for example finite element methods.

In this work we are introducing a reconstruction technique based on topological changes of triangulations. The topology of the resulting mesh is driven by the features represented in the data. Our optimization process turns an arbitrary triangulation of discretely sampled data into a feature-preserving mesh.

The most common reconstruction technique for regularly placed data is convolution-based resampling using reconstruction filters. The drawback is that the features having other directions than the directions of the applied 1D basis functions are not reconstructed very well. This results in blurry artifacts at the border between different features. Geometrically based reconstruction,

denoted as data-dependent triangulation (DDT) was introduced by Dyn *et al.* [DLR90]. In Figure 1 the output from convolution and triangulation based techniques is shown, at 1000% magnification. This kind of triangulation allows the generation of inevitable long and tiny triangles to preserve high-frequency features, and can be applied for arbitrary scattered data. The original DDT is limited to two dimensions and its extension to higher dimensions is not trivial. Its generalization into arbitrary dimension is the main scope of our work. In volume graphics this method can be applied for reconstruction using both regular and irregular grids. Therefore the target applications are tasks, where the reconstruction of sharp features is crucial.

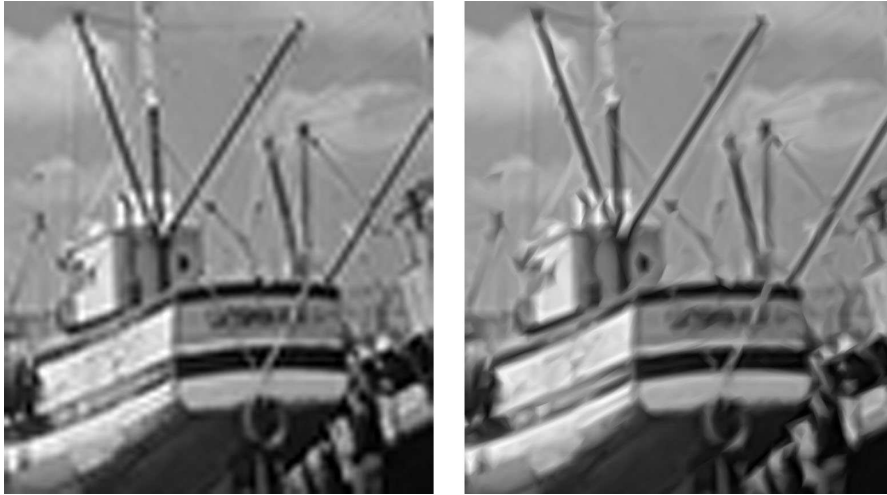


Fig. 1. Reconstruction result with bicubic filtering (left), data-dependent triangulation (right) at 1000% magnification.

The contributions of this paper are the following:

- n -dimensional reconstruction using a data dependent triangulation approach
- a mathematical definition of the optimal reconstruction using triangulations
- a new cost function and generalization of existing functions

This paper is structured as follows. In Section 2 we briefly survey previous work done on DDT. Section 3 defines basic concepts of n -dimensional triangulations and the problem of the extension into higher dimensions is highlighted. In the same section the n -dimensional DDT is introduced. Section 4 presents a feature preserving triangulation algorithm and estimates the usability of the approach. In Section 5 we show a number of examples that demonstrate the

effectiveness of our concept. Finally in Section 6 we draw conclusions from the results and we outline future work possibilities.

2 Related Work

A general notion for interpolating given scalar data over an n -dimensional domain is called *scattered data interpolation*, and mathematically can be expressed as:

$$F(x_{i,1}, x_{i,2}, \dots, x_{i,n}) = z_i, i = 1, \dots, m,$$

where $V = \{V_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})\} \in \mathbf{R}^n$ is a set of not all cohyperplanar, distinct points in the domain space, and z_i are the measured data values. The goal of the reconstruction is to evaluate the function value $F(\mathbf{x})$ for an arbitrary point of the domain. In the rest of this section we give a short review of the related research work done in this field.

Data-dependent triangulation (DDT) is a geometrically based reconstruction technique developed by Dyn *et al.* [DLR90]. It is a special case of optimal triangulations. A general survey on optimal triangulations was done by Bern and Eppstein [BE95]. The quality of the resulting triangulation is defined through a special function called *cost function* and the optimization algorithm. It fits the measured data values with a triangle mesh, creating a piecewise linear interpolation. In contrary to other mesh generation methods DDT maps the alignment of edges to the underlying data and organizes the simplicial structure into a feature preserving mesh. The application of this idea to image reconstruction has been done by Yu *et al.* [YBS01]. Improving the image reconstruction quality has been the scope of our previous work [Tot04]. Kreylos *et al.* [KH01] used it for image compression with a decimation process, based on a simulated annealing optimization technique.

Several algorithms have been developed and different optimization techniques have been applied to generate data-dependent triangulation in the 2D case. Genetic optimization technique was introduced by Kolingerová [Kol99]. The data dependent triangulation technique combined with simulated annealing was first introduced by Schumaker [Sch93]. A real-time version was presented by Su and Willis [SW04]. This pixel-level data-dependent triangulation is limited to regularly placed image data. The results from this simplified version are not as convincing as from other triangulation based techniques. Battiato *et al.* [BGM04] used the concept of a triangulation based technique for creating vector format images from raster data.

Typically, the above mentioned approaches assign the cost function values to the edges in the triangulations. Brown [Bro91] came up with cost assigning to vertices. This idea is called vertex-based data dependent triangulation, and it is a useful improvement of the basic approach.

In the 2D case there is a relation between the Delaunay triangulation of the terrain surface and the data-dependent triangulations. Both can be

constructed with Lawson’s optimization process, also called flipping algorithm [Law77]. The difference is only in the criterion which decides when to take optimization steps. In higher dimensions this optimization technique fails in the creation of the Delaunay triangulation [Joe91]. Only its modification with a special incremental insertion produces the Delaunay n -dimensional triangulation.

Feature preserving triangulations for volume data sets with a mesh refinement technique was studied by Marchesin *et al.* [MDM04] and Roxborough and Nielson [RN00]. Both approaches are based on the longest edge split method.

Other reconstruction techniques are numerical methods applicable for scattered data reconstruction, like C^1 methods in the 3D case by Nielson [Nie93]. These techniques are not as good as Polygonal representation has better readability and easier handling compared to analytical description.

An initial approach to the extend data-dependent triangulation approach into 3D has been proposed by Lee [Lee00]. Lee used the simulated annealing technique to get an optimal data-dependent mesh. This may result in artifacts, as we will show later in Section 5. In the following section we introduce the general approach to n -dimensional triangulation.

3 N - dimensional triangulation

A correct description of triangulation in arbitrary dimension requires to use some definitions from simplex theory. We selected Edelsbrunner’s and Shah’s terminology [ES92].

The convex hull (*conv*) of $k+1$ affinely independent points in \mathbf{R}^n , marked as set T is a k -simplex, denoted by σ_T , where $0 \leq k \leq n$. For subsets $U \subseteq T$, simplices σ_U are called *faces* of σ_T . A collection of simplices, \mathcal{K} , is a *simplicial complex* if:

- (i) The faces of every simplex in \mathcal{K} are also in \mathcal{K} .
- (ii) If $\sigma_T, \sigma_{T'} \in \mathcal{K}$, then $\sigma_T \cap \sigma_{T'} = \sigma_{T \cap T'}$

Let V be a finite point set in \mathbf{R}^n . Usually a triangulation of V is defined as a simplicial complex so that V is the set of 0-simplices (vertices) and the underlying space of the complex is the convex hull of V . A simplicial complex \mathcal{K} is a *triangulation* of V if:

- (i) Each vertex of \mathcal{K} is a point in V .
- (ii) The underlying space of \mathcal{K} is $\text{conv}(V)$.

The *content* of the n -simplex is its generalized volume (in 2D area, in 3D volume, etc.).

From the definition of the triangulation above we can see that it is a C^0 continuous reconstruction. The number of possible valid mesh configurations is exponential in the number of elements in the discrete data set.

Traditional (not data-dependent) triangulations usually avoid the generation of long, bad shaped k -simplices (triangles), because these simplices are not well suited for the finite element method used in simulations. However, such simplices are very suitable for reconstruction of areas where a function has high derivatives in one direction as compared to other directions. In images (2D domain case) such areas correspond to edges, in volume data (3D domain case) to surfaces of volumetric features. Our goal is to reconstruct feature boundaries sharply and correctly using a reconstruction grid adapting to the underlying data structures.

In Figure 2 a part of an image with a boundary between two constant regions is shown. The reconstruction process first converts the image into a height-field representation based on the underlying sample (pixel) values. The height represents the intensity value at a particular pixel position. The initial triangulation is iteratively optimized in order to preserve the feature boundaries. The differences to the initial mesh appear only close to the feature boundary. Figure 2 shows of the final height-field of the feature-preserving triangulation.

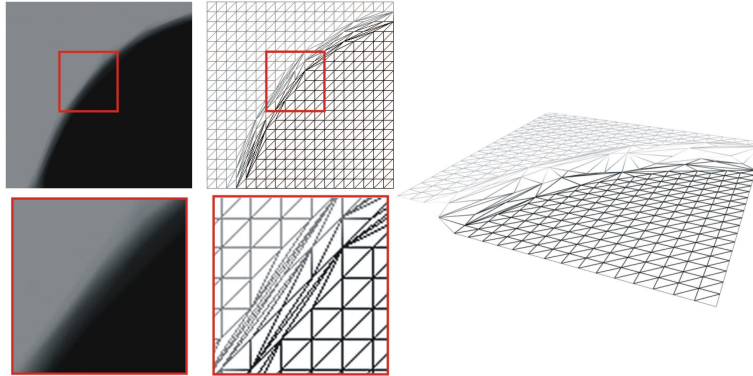


Fig. 2. Shape of the triangles at the border between different features.

In DDT applications cost functions are used to control the shape of the resulting mesh. The task of the optimization process is to improve the mesh via topological changes with regard to the cost function. Cost functions in 2D can be assigned to vertices (this technique is called *vertex based* DDT), to edges, or to triangles. Most of the existing functions are defined for edges. Generally, cost functions assign a cost to every k -simplex in the triangulation, according to some local (not strictly geometrically based) property, where $k = 0, \dots, n$ and n is the dimension of the domain space of the scattered data.

Different triangulations in 2D have the same number of edges. Thus, a topological change can not alter the number of edges in 2D. The goal of the

reconstruction is to minimize the sum of the cost function weights of particular simplices. We get a structurally identical problem to the minimum weight triangulation problem [FNP96], where the task is to find the triangulation with the minimum cost:

$$\text{cost}(\mathcal{K}_{\text{optimal}}) = \min_{\mathcal{K} \in \Omega} (\sum_{e \in \mathcal{K}} (\text{cost}(e))),$$

where $\mathcal{K}_{\text{optimal}}$ is the triangulation generating the minimum sum among the possible configurations marked as Ω and e are simplices assigned with the cost function. However, this problem belongs to the open problems of computational geometry and only its approximation can be computed in polynomial time. In previous work the global optimum was not defined and various heuristics were used for local improvements [YBS01, KH01].

A generalization to higher dimensions based on a constant number of simplices (edges in the 2D case) is not straightforward. The number of simplicial components of the n -dimensional triangulation in 3D and higher dimensions depends on the specific triangulation. For example if in 3D we decide to assign costs to faces (2-simplex) the sum changes not only due to the optimization steps but also depends on the number of faces in the triangulation (see Figure 5). It is necessary to find a solution independent of mesh connectivity changes. A vertex-based data-dependent triangulation satisfies this criterion, because the number of vertices (0-simplices) remains unchanged. However, designing of a cost function for the vertex-based approach is not an intuitive task even in the 2D case. In higher dimensions this becomes even more difficult.

Our solution is based on the observation that the convex hull of the scattered points also remains unchanged. That means that the volume is constant irrespective of the triangulation, and the task of reconstruction can be formulated as follows: the optimization process in the n -dimensional case should assign low weights to n -simplices which are well aligned with the underlying data. This means the minimizing of the weighted volumes of n -simplices summed over the entire space of the convex hull. The exact mathematical description looks as follows:

$$\text{cost}(\mathcal{K}_{\text{optimal}}) = \min_{\mathcal{K} \in \Omega} (\sum_{\sigma_k \in \mathcal{K}} (V(\sigma_k) \cdot w(\sigma_k))),$$

where σ_k is an n -simplex, $V(\sigma_k)$ is its volume, and the assigned weight function $w(\sigma_k)$. $\mathcal{K}_{\text{optimal}}$ is the simplicial complex with minimal weighted volume in the set of possible configurations denoted as Ω . This observation is useful only if we can easily and intuitively find feature-preserving weight functions. A simple approach can use the variance property for weight assignment. Such an idea can be based on the preservation of low variance regions. Therefore the generation of $(n-1)$ -simplices (faces) with low variance is preferred. This implies a weight function based on the variance of the function values in the n -simplex

$$w(\sigma_k) = \text{Variance}(z_{k_1}, z_{k_2}, \dots, z_{k_{n+1}}),$$

where $w(\sigma_k)$ is the weight function for the σ_k n -simplex and z_{k_i} , $i = 1, 2, \dots, n + 1$ are the known function values in the simplex vertices. Figure 3 illustrates the example of a 3D domain case of a triangulation on a given surface. If the second type of triangulation will be chosen (see Figure 5 image on the right), the weighted volume value will be higher.

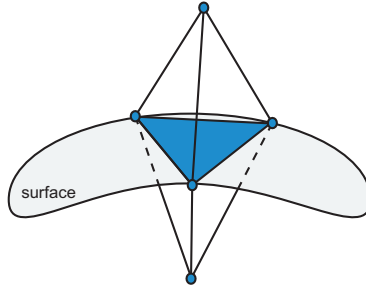


Fig. 3. Example of correct tetrahedralization according to the face variance.

Another useful property of triangulations in arbitrary dimensions is that $(n - 1)$ -simplices not lying at the boundary of the convex hull are exactly shared with two n -simplices. In other words, in 2D triangulations each interior edge is shared exactly by two triangles, in 3D each interior triangular face is shared exactly by two tetrahedrons, etc. This allows us to generalize most of the known cost functions from the 2D case for the $(n - 1)$ -simplices. Each n -simplex contains $(n + 1)$ of $(n - 1)$ -simplices, for which the generalized feature-preserving cost can be evaluated. Averaging of these values gives a feature fitting weight function for our technique. We illustrate this on a concrete cost function generalization into arbitrary dimensions. We have chosen a cost function from the 2D case with the most convincing result, i.e. Sederberg’s cost function [YBS01]. In Figure 4 the cost function dependency for data-dependent triangulation is shown over a planar domain. T_1 and T_2 are triangles at the generated piecewise linear surface, they share a common edge, denoted as e . Sederberg’s cost function is based on the angle α between the gradients of the planes containing triangles T_1, T_2 , weighted by the magnitude of the gradients:

$$cost(e) = \|\nabla P_1\| \cdot \|\nabla P_2\| \cdot (1 - \cos(\alpha)) = \|\nabla P_1\| \cdot \|\nabla P_2\| - \nabla P_1 \cdot \nabla P_2,$$

$\nabla P_1, \nabla P_2$ are the gradients of the planes containing T_1, T_2 , and $\|P_1\|, \|P_2\|$ are their magnitudes. Angle α is the angle between these gradients.

Its straightforward generalization into higher dimension looks as follows:

$$cost(\sigma_{n-1}) = \|\nabla P_1\| \cdot \|\nabla P_2\| - \nabla P_1 \cdot \nabla P_2,$$

where linear polynomials $P_1(\mathbf{x})$ and $P_2(\mathbf{x})$ represent the hyperplanes. These hyperplanes are formulated from the n -dimensional domain and the function values of the data function:

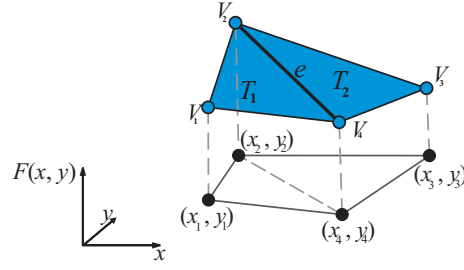


Fig. 4. Illustration of geometric dependencies in 2D data dependent triangulation.

$$P_i(\mathbf{x}) = a_{i,1}x_1 + a_{i,2}x_2 + \dots + a_{i,n}x_n + a_{i,n+1}, i = \{1, 2\}.$$

∇P_i is the gradient on the hyperplane, and $\|\nabla P_i\|$ is the magnitude of the gradient:

$$\begin{aligned} \nabla P_i &= (a_{i,1}, a_{i,2}, \dots, a_{i,n}), i = \{1, 2\}, \\ \|\nabla P_i\| &= \sqrt{a_{i,1}^2 + a_{i,2}^2 + \dots + a_{i,n}^2}, i = \{1, 2\}. \end{aligned}$$

With the help of these generalized cost functions we can describe a feature-preserving mesh, as follows:

$$w(\sigma_k) = \frac{\sum_{\sigma_j \in \sigma_k} \text{cost}(\sigma_j)}{(n+1)},$$

where σ_k is an n -simplex and $w(\sigma_k)$ is its weight function. The generalized cost functions of the $(n-1)$ -dimensional subsets (faces) of this simplex are averaged. The $(n-1)$ -simplices are marked in the sum as σ_j .

Other geometrical properties can also be involved in the weight function design. On the base of the shape of $(n-1)$ -simplices we can consider their weighted sum instead of the unweighted sum. Having feature fitting weight functions the task is to solve the construction of an optimized mesh. This problem is treated in the next section.

4 Content Based Data-Dependent Reconstruction

As we mentioned earlier, the feature preserving property of the reconstruction technique is achieved by specific quality-improving operations on the topology of the triangulation. In 2D this kind of topology transformation is called *edge flip* and it changes the topology as follows: In Figure 4 the edge e can be replaced with the edge $V_1 - V_3$ if the four vertices form a strictly convex quadrilateral. Via flip improvements the cost of the triangulation is decreased iteratively, and we are getting a feature-preserving mesh. The 2D triangulation algorithm based on this idea is called *Lawson's optimization process* [Law77].

In higher dimensions the notion of an edge flip generalizes to *bistellar flips*. Bistellar flips contain vertex removing and inserting into triangulations. We

are interested only in such transformations where the number of vertices does not change. This general topological operation is based on Radon's theorem from convex geometry [ES92]. Each transformation can be interpreted as a projection of simplex into a lower dimension. The views of the simplex from antipodal points of view in the direction of the projection introduces the two configurations, before and after the flip. For example if we project a tetrahedron into a plane we get an edge flip. Projections which cause degeneracies in the projected complex can be interpreted as bistellar flips of lower dimension in the simplex (of lower dimension) where the degeneracy occurred.

Let us describe the set of bistellar flips in 3D:

- A face can be changed into an edge as is illustrated in Figure 5, top part. This is called a 2-3 flip.
- An edge can be changed to a face as is illustrated in Figure 5, top part. This is called a 3-2 flip.
- A degenerate case occurs when six vertices form four adjacent tetrahedrons with one common edge as is illustrated in Figure 5, bottom part. If four of these vertices are coplanar and the remaining vertices are divided by this plane, there are two possible ways how to tetrahedralize this structure. This operation is called 4-4 flip.

In higher dimensions the situation is the following:

- Let us consider an n -dimensional space, and a simplicial complex \mathcal{M} of n -simplices which share a common edge. If \mathcal{M} has $n + 2$ vertices and forms a convex space then there exist exactly two triangulations of \mathcal{M} . One that includes the common edge and one that instead includes a hyperface formed by the vertices not related to the removed edge.
- All other possibilities are degeneracies and can be described as bistellar flips of lower dimension.

Unlike in the 2D case it is not proven that with these bistellar flips one can get from an arbitrary triangulation to any other possible configuration. It is proven up to our knowledge only in the 3D case for points in convex position [Bes01].

Another possibility for mesh improving is to investigate more complex topological transformations, like the *edge-removal* operation in the 3D case. It is a transformation that removes a single edge from the mesh along with all tetrahedra that contain it. This operation can be composed from a series of bistellar flips, but in that way the optimization can get stuck in local optima. Effective implementation was discussed in the work of Shewchuk [She02].

To make our concept more general we define the set $\mathcal{T}_{\mathcal{K}}$ of topological transformations of triangulation \mathcal{K} . The members of this set can be selected arbitrarily. Reconstruction will be done considering the members of this set.

Good reconstruction results can be expected from such a set of $\mathcal{T}_{\mathcal{K}}$, that generates all possible topological transformations at the given dimension. We are interested in those topological operations which remove a specific k -simplex $k = 1, \dots, n-1$ and change the topology of n -simplices containing the removed k -simplex.

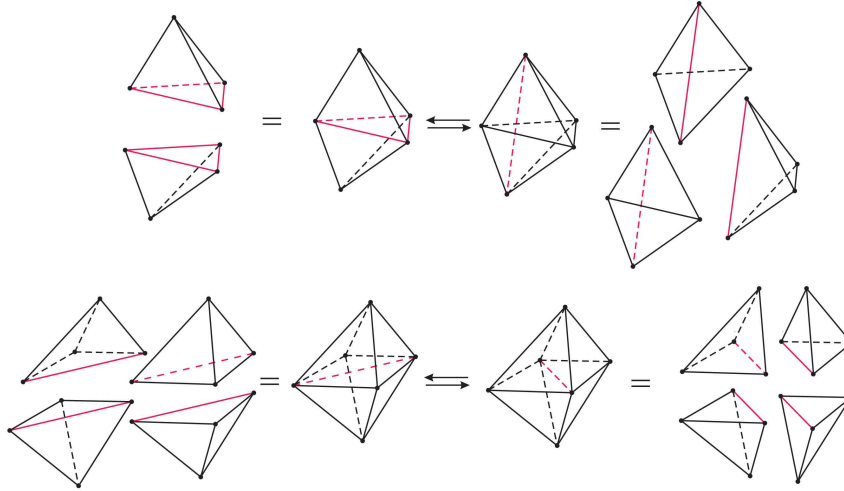


Fig. 5. Illustration of the 3 – 2 - flip and 2 – 3 - flip topological transformations (top image) and the 4 – 4 - flip topological transformation (bottom image).

Let us have a triangulation \mathcal{K} for the n -dimensional domain. The k -simplex $k = 1, \dots, (n - 1)$ is *locally optimal* with regard to $\mathcal{T}_{\mathcal{K}}$ and to a given cost function if:

- the k -simplex cannot be removed from the triangulation by applying one of the topological transformations from $\mathcal{T}_{\mathcal{K}}$
- there is a topological transformation in $\mathcal{T}_{\mathcal{K}}$ that removes the k -simplex (let us denote this k -simplex as *flippable*), but the triangulation cost doesn't decrease after the applied changes.

Triangulation \mathcal{K} is *locally optimal* if topological transformations from $\mathcal{T}_{\mathcal{K}}$ cannot improve its weighted content.

We are introducing the generalization of Lawson's algorithm which creates locally optimal data-dependent triangulation. The pseudocode of the algorithm is presented on Figure 6. At first an initial triangulation is generated, as it is written in *line2*. We suggest for this role the Delaunay triangulation. The initial triangulation should connect each vertex with its closest neighbors. Due to the duality with Voronoi diagrams, the Delaunay triangulation satisfies this criterion. Creation of good initial triangulations speeds up the

running time, as the algorithm needs less iterations to converge to a locally optimal solution. For image and medical volume data reconstruction this means the following: The optimization process at low-frequency areas will leave the initial triangulation unchanged, only at high frequency areas it will generate long feature-preserving simplices.

After creating the initial triangulation its weighted content is evaluated - *line3*. The *oldCost* variable is used to store the triangulation cost of the previous iteration step. Two lists are initialized in *line5* and *line6*. *List_active* contains all the simplices whose local optimality is tested in the given iteration step. In the beginning this list contains all $1, \dots, n - 1$ simplices of the triangulation \mathcal{K} . The second list *List_candidate* is the container of simplices whose local optimality could change due to the applied transformations from $\mathcal{T}_{\mathcal{K}}$. We denote these simplices with σ_k . At the beginning of the optimization process *List_candidate* is set to empty. In a given iteration local optimality of the members of *List_active* are tested. If there exists a cost reducing topological transformation from $\mathcal{T}_{\mathcal{K}}$ then it is applied. The tested simplex is removed from *List_active*. Into *List_candidate* those simplices are added whose local optimality could change. At the end of each iteration step *List_active* is empty and *List_candidate* contains all the simplices that will be tested in the next iteration step. The algorithm stops, if the triangulation cost was not improved in the last iteration step. This procedure results in a locally optimal triangulation \mathcal{K} with regard to a set of topological transformations $\mathcal{T}_{\mathcal{K}}$ and a given cost function.

It is evident that the described algorithm stops after a final number of iteration steps. In each step we decrease the overall cost of the triangulation and the number of possible triangulations is finite. The described idea of a weighted content based DDT can be used with other optimization techniques also. It is possible to construct a simulated annealing, look-ahead, or genetic optimization approach based on this idea.

5 Experiments and Results

DDT triangulations can be computed by stochastic processes. Such a technique is the simulated annealing optimization approach. Its usage can improve the approximation level of the algorithm to receive results which are closer to the global optimum than the described generalized Lawson's optimization process. However our experiences using simulated annealing does not create visually pleasant results. The initial stages of simulated annealing apply topological transformations on the mesh which increase the cost of the triangulation. This results in long, bad shaped simplices in areas without high-frequency features. Our goal was to create these bad shaped simplices only at high frequency areas. In Figure 7 a comparison between Lawson's optimization process and the simulated annealing technique is depicted, both using Sederberg's cost function. The reconstruction is displayed at 600% mag-

Generalized Lawson's optimization process

Input: scattered data
Output: locally optimal triangulation

```

1  begin
2  create the initial triangulation  $\mathcal{K}$ ;
3   $Cost = cost(\mathcal{K})$ ;
4   $oldCost = Cost + 1$ ;
5   $List_{active} = \{\forall \sigma_k \in \mathcal{K}, k = 1, \dots, n - 1\}$ ;
6   $List_{candidate} = \emptyset$ ;
7  while ( $Cost < oldCost$ )
8  {
9    for ( $List_{active}$  members)
10   {
11     if (not locally optimal)
12     {
13       apply transformation from  $\mathcal{T}_{\mathcal{K}}$ ;
14       remove from  $List_{active}$ ;
15       for ( $\sigma_k$  whose local optimality could be changed)
16       {
17         if ( $\sigma_k$  is not member of  $List_{active}$  and  $List_{candidate}$ )
18         add  $\sigma_k$  to  $List_{candidate}$ ;
19       }
20     }
21     else
22       remove  $\sigma_k$  from  $List_{active}$ ;
23   }
24    $oldCost = Cost$ ;
25    $Cost = cost(\mathcal{K})$ ;
26    $List_{active} = List_{candidate}$ ;
27    $List_{candidate} = \emptyset$ ;
28 }
29 locally optimal  $\mathcal{K}$  created;
30 end

```

Fig. 6. The pseudocode of the n -dimensional DDT algorithm.

nification, the mesh of the resulting triangulation and the reconstructed 2.5D terrain are also included. In low frequency areas (e.g., cheek area below the eye) simulated annealing creates long and tiny triangles. Their generation is in this case undesired. From the mathematical point of view this observation for image and medical volume data reconstruction can be formulated as follows: Qualitative reconstruction should give a good approximation of the minimum weight triangulation, but should not change the topology in low-frequency regions.

In Section 3 two types of weight functions have been described, the variance-based approach and an approach based on the generalization of ex-

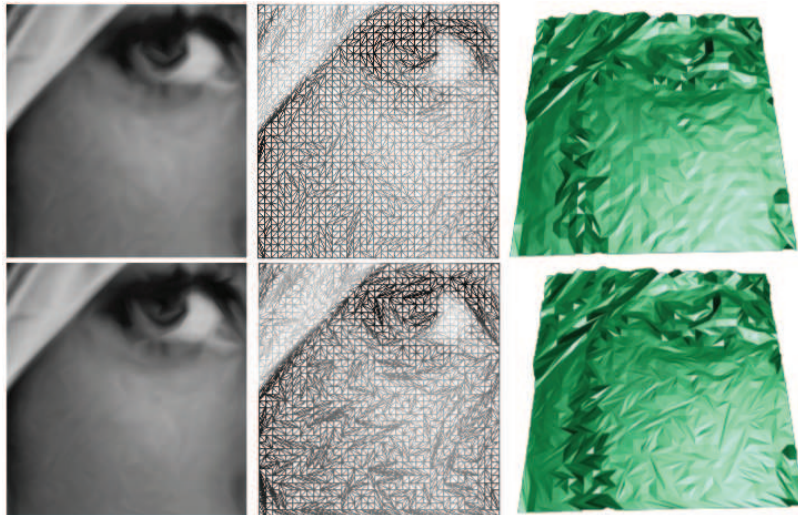


Fig. 7. Results from Lawson's optimization process (upper row), and from simulated annealing (lower row) at 600% magnification. The last column shows the triangulation of the height field.

isting cost functions into higher dimensions. We should notice that there is a difference between these techniques. The variance based weight function is clearly a C^0 continuous approach and depends only on the data values from the tested n -simplex vertices. The weight function based on generalized cost functions has different properties. It depends on the data values from the tested n -simplex vertices and on the vertices of n -simplices which share with the tested one a common $(n - 1)$ -simplex. For this reason these weight functions can be classified as *near* C^1 continuous weight functions. Because of this difference the running time of variance-based cost function DDT is lower than the running time of the DDT with generalized cost functions. The reason is that the weight function evaluation is computationally less complex, and the number of simplices whose local optimality could change is smaller.

We plan to include examples from 3D and 4D domain in the final version.

6 Conclusions and Future Work

We have introduced a new method using topological changes for n -dimensional data-dependent triangulation. This enables a better visualization of the local topology of sharp features both for with regular and sparse grids in arbitrary

dimension. Topology of the reconstruction grid is driven by the topology of the underlying data.

The potential of this new reconstruction technique results in many future work directions:

Our feature preserving triangulation can be used for data compression. If we assign a cost to vertices we have information about how important the vertices are. With mesh decimation techniques the non important vertices can be removed.

A straightforward use of the described technique for the reconstruction of time-varying volume data is possible. The reconstruction errors from known image processing methods are more disturbing than in static reconstruction.

Optimal triangulations in 2D are relatively well explored. Higher dimensional optimal triangulations is novel area for investigation. We hope that our contribution improves the theory in this research area and its application provides a new tool for practical scattered data reconstruction.

7 Acknowledgments

Funds for the support of this research have been allocated by the Rectorate of the Comenius University and the Ausseninstitut der Technischen Universität Wien. Part of the work presented in this publication is carried out within the **exvisation** project (www.cg.tuwien.ac.at/research/vis/exvisation) supported by the Austrian Science Fund (FWF) grant no. P18322.

References

- [BE95] BERN M., EPPSTEIN D.: Mesh generation and optimal triangulation. In *Computing in Euclidean Geometry*, second ed., no. 4. World Scientific, 1995, pp. 47–123.
- [Bes01] BESPAMYATNIK S. N.: Enumerating triangulations of convex polytopes. In *Proc. Conf. Discrete Models: Combinatorics, Computation, & Geometry (DM-CCG 2001)* (2001), Cori R., Mazoyer J., Morvan M., Mosseri R., (Eds.), no. AA in Discrete Mathematics & Theoretical Computer Science Proceedings, pp. 111–122.
- [BGM04] BATTIATO L., GALLO G., MESSINA G.: Svg rendering of real images using data dependent triangulation. In *In Proceedings of Spring Conference on Computer Graphics 2004* (2004), pp. 191–198.
- [Bro91] BROWN J.: Vertex based data dependent triangulations. *Computer Aided Geometric Design* 8, 3 (1991), 239–251.
- [DLR90] DYN N., LEVIN D., RIPPA S.: Data dependent triangulations for piecewise linear interpolation. *IMA Journal of Numerical Analysis*, 10 (1990), 137–154.
- [ES92] EDELSBRUNNER H., SHAH N. R.: Incremental topological flipping works for regular triangulations. In *SCG '92: Proceedings of the eighth annual symposium on Computational geometry* (1992), ACM Press, pp. 43–52.

- [FNP96] FERKO A., NIEPEL L., PLACHETKA T.: Criticism of hunting minimum weight triangulation edges. In *Proceedings of the 12th Spring Conference on Computer Graphics* (1996), pp. 259–264.
- [Joe91] JOE B.: Construction of three-dimensional delaunay triangulations using local transformations. *Comput. Aided Geom. Des.* 8, 2 (1991), 123–142.
- [KH01] KREYLOS O., HAMANN B.: On simulated annealing and the construction of linear spline approximations for scattered data. *IEEE Transactions on Visualization and Computer Graphics* 7, 1 (2001), 17–31.
- [Kol99] KOLINGEROVÁ I.: Genetic approach to data dependent triangulations. In *In Proceedings of Spring Conference on Computer Graphics 1999* (1999), pp. 229–238.
- [Law77] LAWSON C.: Software for c^0 interpolation. *Mathematical Software III.* (1977), 161–194.
- [Lee00] LEE K.: Three-dimensional medical image modeling of scattered data based on data-dependent criteria. In *Proc. SPIE Vol. 4117, Vision Geometry IX* (2000), pp. 91–99.
- [MDM04] MARCHESIN S., DISCHLER J. M., MONGENET C.: 3d roam for scalable volume visualization. In *2004 IEEE Symposium on Volume Visualization and Graphics (VV'04)* (2004), IEEE Computer Society, pp. 79–86.
- [Nie93] NIELSON G. M.: Scattered data modeling. *IEEE Comput. Graph. Appl.* 13, 1 (1993), 60–70.
- [RN00] ROXBOROUGH T., NIELSON G. M.: Tetrahedron based, least squares, progressive volume models with application to freehand ultrasound data. In *VISUALIZATION '00: Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000)* (2000), IEEE Computer Society.
- [Sch93] SCHUMAKER L. L.: Computing optimal triangulations using simulated annealing. In *Selected papers of the international symposium on Free-form curves and free-form surfaces* (1993), Elsevier Science Publishers B. V., pp. 329–345.
- [She02] SHEWCHUK J. R.: Two discrete optimization algorithms for the topological improvement of tetrahedral meshes. Unpublished manuscript, 2002.
- [SW04] SU D., WILLIS P.: Image interpolation by pixel-level data-dependent triangulation. *Comput. Graph. Forum* 23, 2 (2004), 189–202.
- [Tot04] TOTTH Z.: Towards an optimal texture reconstruction. In *Proceedings of the Central European Seminar on Computer Graphics '04* (2004).
- [YBS01] YU X., BRYAN B. S., SEDERBERG T. W.: Image reconstruction using data-dependent triangulation. *Computer Graphics and Applications* 21, 3 (2001), 62–68.