HYDROACOUSTICS
4, 37-42 (2001)

3^D IMAGING OF SEABED FROM ELECTRONIC CHART BATHYMETRIC DATA

J. Demkowicz[1] and A. Stepnowski[2]
[1]C-MAP Poland Ltd., Narutowicza 11/12, 80-952 Gdańsk
[2]Technical University of Gdańsk, Remote Monitoring Systems Department, 80-952 Gdańsk
e-mail: jdemkowicz@c-map.pl, astep@pg.gda.pl

Application of 3^D imaging and mapping of seabed is very attractive in many areas including marine cartography, navigation, hydrography, GIS systems etc. There are two competing approches to constructing 3^D images of seabed. The first, off-line method, utilizes the bathymetric information extracted from vectorized digital navigational charts. The second one, on-line technique, employs multibeam sonar seabed echo records processing. The paper describes procedures of bathymetric data analysis for the first method along with newly proposed rendering schemes.

INTRODUCTION

A three dimensional (3^D) graphics has been finding its place in many fields of marine science and technology and it has become almost indispensable tool in such applications like sea floor mapping and characterization, marine GIS, ECDIS and related systems. Many European and US marine institutions and companies as C-MAP, Simrad, Spectra Precision Software, Reson, Elac, Roxar etc. developed and implemented various 3^D mapping/imaging systems for hydrography, navigation, remote sensing and visualisation of marine environment and marine ecosystems.

3^D graphics has found its place in the mentioned areas for the several reasons, viz.:

- 3^D graphics, when presented in a very realistic manner (after rendering), constitutes a very precise language of a human intercommunication.
- 3^D space can be projected in a natural way utilizing symbols, images or maps.
- relations between the objects and their spatial relations are already contained in 3D graphics representation, and redundancy of information is drastically reduced in that way.
- last but not least are psychological reasons, which have a great impact on a way of communication; there is also related topic, though not strictly referred to the technical aspects, but important in the context of ergonomy, management, safety on the sea etc.

A very attractive tool in many applications, and particularly in marine information systems and hydrography is a rendering process, which is responsible for 3^D graphic visualization. Rendering stands for imaging process, which, after applying a shading, texturing and other image processing operations, generates a 3^D object model which seems almost like the actual objects in the reality [1].

1. RECONSTRUCTION METHODS

A vectorized World Wide Electronic Chart Database CM-93 from C-MAP [4] constitutes the main source of input bathymetric data (soundings) for 3^D rendering process and eventual 3^D seabed imaging. To generate a set of sounding from the carthographic library C-MAP CM-93/3 the Software Development Kit 3.2 was used [3]. The set of the output data included: depth of every sounding and its position. So, there are three coordinates prescribed to every soundig (see Fig. 1.)

The second stage was the generation of a discrete 2^D representation of the sea bed, with the maximum attainable accuracy. It means, that a proper triangulation procedure should be applied. In the experiment, a fast Delaunay method was choosen [1]. The triangulation was carried out in a 2^D space over soundings casted perpendiculary at a sea level plane. Following the triangulation, the relative sounding depth was added for each triangle vertex (as each soundig has three coordinates).

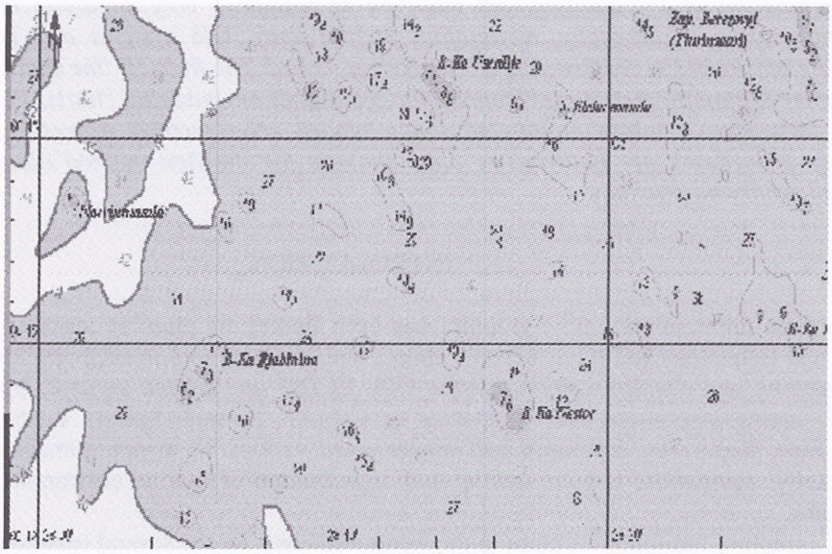

Fig.1. Electronic Navigational Chart in CM-93 format with soundings [North – East Baltic].

The algorithm used to construct the Delaunay triangulation is due to Strain and Russo [2]. In the first step the adaptive quad tree is constructed, which partitions the data points $\boldsymbol{Z}$ set into cells of varying sizes, each of which contains no more than $\boldsymbol{m}$ points. If the points of $\boldsymbol{Z}$ set are in a rectangle $\mathbf{R} = [\mathbf{a_x},\mathbf{b_{x.}}] \times [\mathbf{a_y},\mathbf{b_y}]$. The first step of the quad tree construction includes dividing the rectangle containing the data points (positions of soundings), into four equally sized sub-cells and assigning each point to the cell which contains it. For the next level of the tree, one checks each of the four cells to see if the number of points inside is greater than $\boldsymbol{m}$; if not, one keeps the cell and go to the next one. Then, the cell is divided into four sub-cells, if the cell has too many points, and the point is assigned to the corresponding sub-cell, mark the cell for deletion. Then each of the remaining cells is checked at this stage and are divided or kept as appropriate. The algorithm descends level by level until either a maximum level is reached or all of the cells have no more than m points.

The next stage in the algorithm deals with the constructing of Delaunay triangulation, which procedure is as follows. One gets an arbitrary point z_i in the Z point set and then finds its closest neighbor. This point is called z_j. Now, one begins with this edge, which is guaranteed to be a part of the Delaunay triangulation, and finds a point to complete the triangle. Due to Delaunay triangulation principle, in order to have a valid triangulation of a set of points Z, the circumcircle of any triangle (i.e. the circle intersecting the three vertices of the triangle) can not contain other points of Z in its interior. So, the next point is added to minimize the size of the circumcircle. Let z_m be the midpoint of the edge $z_i \rightarrow z_j$, and let n be the unit normal to $z_i \rightarrow z_j$. One can express the circumcircle requirement in another way by saying that he wants to find a point z which minimizes the following function $f(z)$

$$\mathbf{f(z)} = \frac{\overrightarrow{(z - z_i)} \cdot \overrightarrow{(z - z_j)}}{2 \cdot \overrightarrow{(z - z_o)} \cdot \vec{n}}$$

Calling O the center of the circumcircle passing through the three points z, z_i and z_j, then $f(z)$ computes the signed distance from O to the line connecting z_i and z_j (see Fig.2).

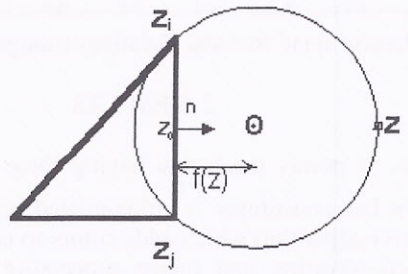

Fig. 2. The signed distance function $f(z)$.

Next, order three points so that the triangle is oriented counter-clockwise. The algorithm, from now includes going through the list of existing triangles and trying to add a triangle to each edge which is not already occupied. To do this, take a given edge $z_i \longrightarrow z_j$ and search the one or two cells q_i and q_j which contain the points of this edge for a point z which is on the correct side of $z_i \longrightarrow z_j$ and which minimizes $f(z)$.

If we have now found such a point z, and if the circumcircle of these three points is inside $q_i \, U \, q_j$, then this is right point. If we have not found a point or the circumcircle is outside of $q_i \, U \, q_j$, we check the nearest neighbor cells of q_i and q_j. After searching the nearest neighbors, there are two possibilities: if there is still no a proper point, check the remaining cells to see if the current edge is on the convex hull. If a point has been found (so that now we have the edge $z_i \rightarrow z_j$ as well as a point z which is in a neighboring cell of q_i or q_j).

A sample plot of the adaptive quad tree and the triangulation for n=40, m=3 is shown in Fig.3.

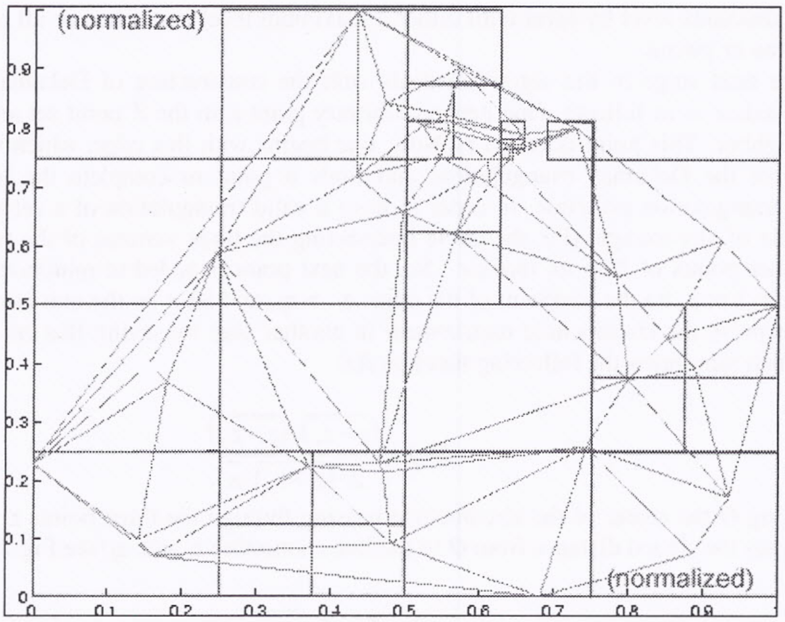

Fig. 3: Adaptive quad tree and Delaunay triangulation for n=40, s=3.

2. RESULTS

Such prepared triplets of points (each one having three coordinates make triangles and eventually a mesh, which in fact constitutes 3^D triangulated irregular network (TIN) - see Fig. 5. The triangles, in a recursive algorithm which adds colors to each vertex in a gradient way, are further processed in the visualization and image processing module, where are subject of shadowing and a blending etc. [4]. It leads eventually to a 3^D image seabed reconstruction, sample of which is presented in Fig 5.

In the reconstruction process the TerraVista Lite 3^D imaging module was used, which has following functionality: orthographic, perspective and plan camera view, dynamic rotation, smoothing, color coding and wire frame color displaying [4].

As all these operations take a negligible amount of time, so all can be performed nearly in a real time processing mode.

3. CONCLUSION

The applied 3^D reconstruction of seafloor images was not affected by a soundings redundancy, as generally there are not to many soundings at any electronic navigational chart of a given region. Such an automatic generation of 3^D images is not excessively time-consuming when using typical PC like Pentium 300MHz (64 M RAM). However, the system was not tested neither with a larger amount of soundings, nor on charts covering larger areas.

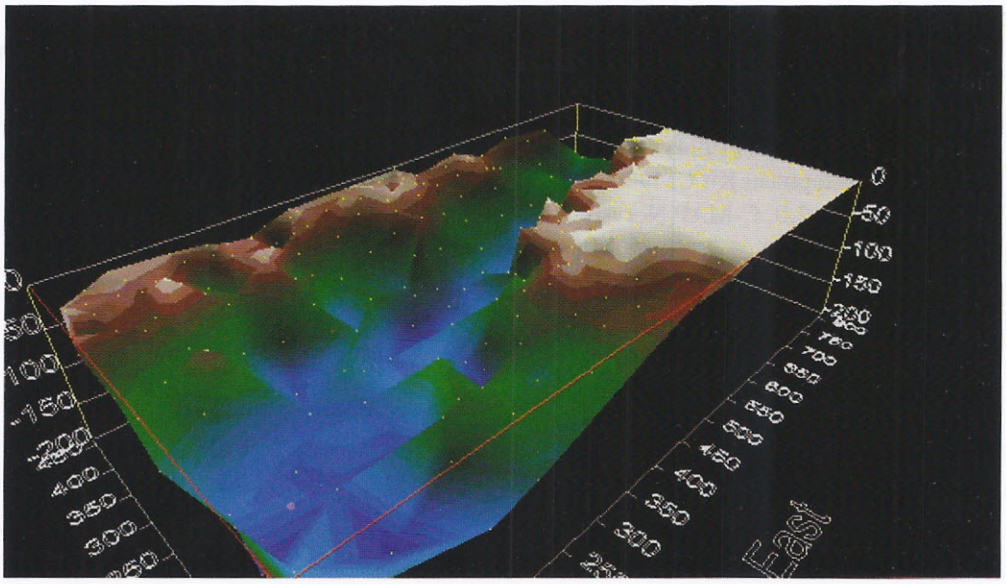

Fig. 4. Results of the reconstruction process; every dot represents the sounding retrieved from electronic chart

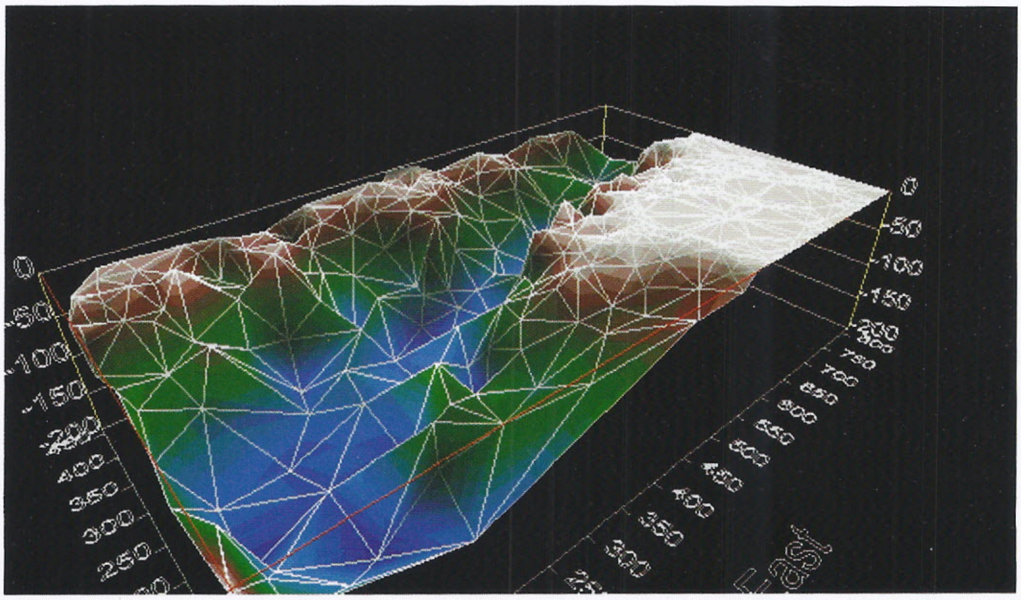

Fig. 5. Results of the reconstruction process after applying TIN

REFERENCES

1. R. Franklin, Terrain Elevation Data Structure Operation, New York, Rensselaer Politechnic Institute, (1999).
2. G. Russo and J. Strain, Fast Triangulated Vortex Methods for the 2D Euler Equations, Journ. Comp. Phys. 111, 291 (1994).
3. Software Development Kit CM93/3, Rel. 3.2, C-MAP Norway, (2000).
4. .Ocean View ™, 2.0, C-MAP Norway A/S, (2000).