A-016

A Ridge Line Detection Method of Triangulated Surfaces Derived from Rectangular Dissections

Koushi Anzai† Takaaki Goto‡ Shinji Koka‡† Youzou Miyadera‡‡ Takeo Yaku‡†

Keywords and phrases: Ridge line detection, TIN, algorithms, data structures, terrain maps, image processing

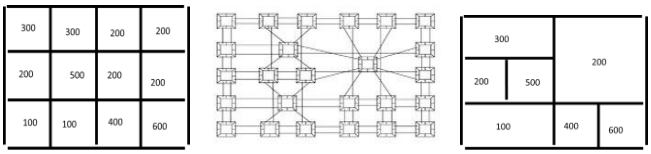
1. Introduction

Raster data is commonly used for surface CGs, since machine generated digital data is mostly created as raster data such as DEMs and digital images. However, raster data is not appropriate in ridge detection or drainage computation. For those problems, triangulated irregular networks (TINs) are appropriate. In this paper, we deal with triangulation of raster data. We assume that raster data are represented as rectangular dissections.

On the other hand, resolution reduction is important with respect to the number of rectangles in rectangular dissections. Resolution reduction methods have been investigated for rectangular dissections with octgrid models (see e.g., [5]). Those methods provide heterogeneous rectangular dissection from homogeneous rectangular dissections.

First, this paper introduces a method that obtains irregular triangular dissections with the nodes attributed by elevation values from irregular triangular dissections with partial nodes attributed by elevation values. Next, we employ above triangular dissections and introduce a concept of ridge line detection method.

2. Related Work



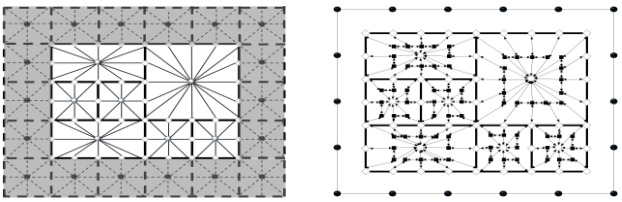
A rectangular dissection

An octgrid representation of a resolution reduced rectangular dissection

A resolution reduced rectangular dissection

Resolution reduction with an octgrid model is obtained in [3]. *Octgrid methods experimentally provide rectangular dissections with about one third or less number of rectangles than quadtree based methods do, and faster than rectangular dual based methods.*

An octgrid based triangulation algorithm of rectangular dissections without altitudes is introduced in [5]. *This triangulation method is expected to provide triangular dissections with less triangles than quadtree based methods do.*



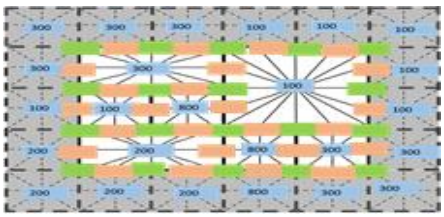
3. Triangulation

We propose an altitude assigning algorithm for octgrid represented triangular dissections and complete our triangulation method of rectangular dissection based surfaces of octgrid models.

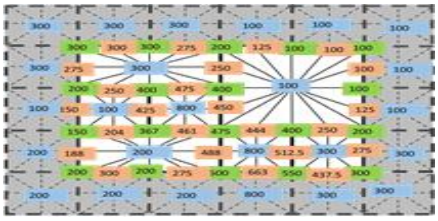
Algorithm *AltitudeAssigning4TriangularDissection* (*outline*)

INPUT: A triangular dissection *T*
OUTPUT: A triangular dissection *S* with altitude values
METHOD: Initialization alt(*v*) = 0 for all *v* in T
for all nodes *v* in *T* **do**
if *v* is a center node, put alt(*v*) the altitude of the rectangles
if *v* is a cross point of rectangle edges, **then** put alt(*v*) the mean value of altitudes of four surrounding rectangles
if *v* is at a T-corner of rectangle edges, **then** put alt(*v*) the mean value of altitudes of three neighboring rectangles
if *v* is located on an edge of rectangles, **then** put alt(v) the mean value of altitudes of two neighboring nodes and two neighboring rectangles
end

(A detail of the algorithm)
Initialization
We assume that the altitudes of rectangles are assigned to the center nodes.



Step 1
For the nodes at cross points of rectangle edges, put altitudes be mean values of altitude of three or four surrounding nodes.
Step 2
For the nodes on a edge, put the altitude be the mean value of altitudes of two neighboring junction nodes and two neighboring center nodes



† Kantou Gakuen Universiry
‡ Ryutsu Keizai University
‡ † Nihon University
‡ ‡ Tokyo Gakugei University

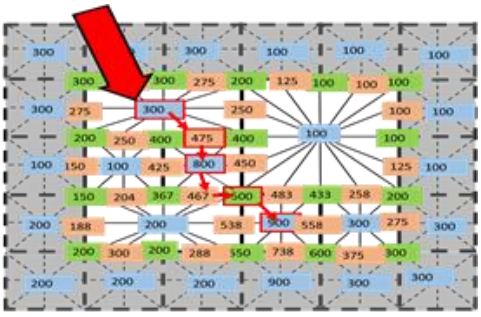
4. Ridge Line Detection

We propose a ridge line detection method along edges for octgrid represented triangular dissections.

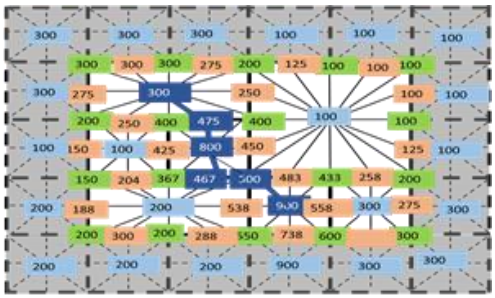
ALGORITHM *RidgeLineAlongEdgesInTriangle* (outline)
INPUT: Triangular dissection *S* with the altitudes of the nodes; A threshold value g0:.
OUTPUT: A triangular dissection *R* with ridge edge lines colored by *blue*
METHOD
set color of all edges *e* to *white* and count of all edges to 0;
for all *v* in *S* **do** {
put *mark*(*v*) and *mark*(*v*, *u*) "*unvisited*" for all *v* and [*v*, *u*];
visit repeatedly steepest ascent edges starting from *v* until traverse arrived a ridge node *r*, while the *count* of the visited edge is increased by one;
RidgeTraverse(*r*)}
/* Finalization */
for all [u, v] which *count* is greater than the threshold values g0 **do** put color of edges *blue*

ALGORITHM *RidgeTraverse*(*r*) (outline)
INPUT: Triangular dissection *S* with the altitudes of the nodes, where nodes and edges are marked *visited* or *unvisited*, a node *r* on a ridge
OUTPUT: A triangular dissection *R,* where nodes and edges are marked "*visited*" or "*unvisited*"; *count*
METHOD
For all unvisited node *s* which are on a ridge and adjacent to *r* **do** {
increase *count*(*s*) and *count*[*r*, *s*] by one;
mark *s* and [*r*, *s*] visited;
if no unvisited adjacent node to *r* is on a ridge **then quit else** *RidgeTraverse*(*s*)
}

We show the processing of the algorithms.
First, a ridge line concerning to one node is detected：

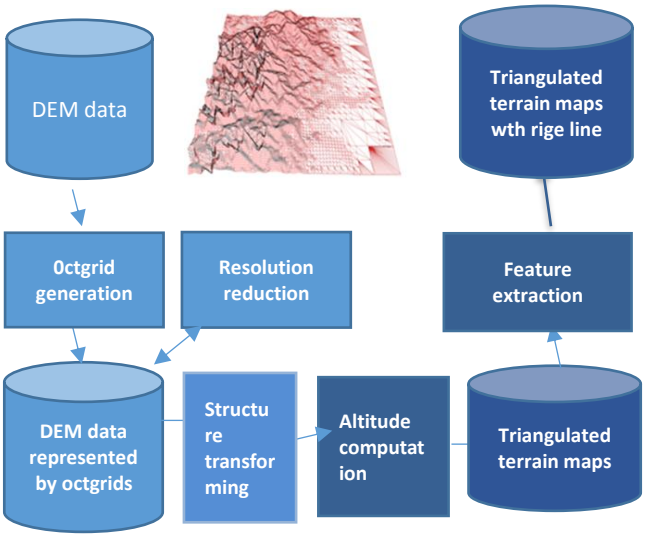


Next, ridge lines (s) is detected.:
The collection of edges and nodes colored *blue* is ridge lines.



5. Application

We illustrate in the figure below the whole system based on DEMs with our triangulation methods. The system generates triangulated terrain maps from DEM data with feature extracting function such as ridge detection.



An image of a triangulation system (dark blue) and a whole display system

6. Conclusion

We introduced a triangulation method of rectangular dissection based terrain data with an altitude assigning algorithms. Next, we introduced an outline of a ridge line detection algorithm for triangular irregular networks generated by above method. Methods are applied to feature extraction of images and surfaces. Among them are edge detection of images and drainage detection for triangulated terrain maps.

Acknowledgement

Authors would like to thank to Dr. Koichi Anada of Waseda University Senior High School and Prof. Kensei Tsuchida of Toyo University for their interest

References

[1] R. A. Finkel, J. L. Bentley, "Quad Trees: A Data Structure for Retrieval on Composite Keys", *Acta Inform.* 4, 1-9 (1974).
[2] R. Pajarola, M. Antonijuanand R. Lario, QuadTIN: Quadtree based triangulated irregular networks, *Proc. IEEE Vizualization* 2002, 395 – 402 (2002).
[3] G. Akagi, K. Anada, S. Koka, Y. Nakayama, K. Nomaki, T. Yaku, "A resolution reduction method for multi-resolution terrain maps". *SIGGRAPH Posters* 2012: 86 (2012)
[4] T. Yaku, K. Anada, K. Anzai, S. Koka, Y. Miyadera, K. Tsuchida, "8k-ary Grid Graph Models of Tabular Forms. Specification", *Algebra, and Software* 2014: 465-477 (2014).
[5] T. Kikuchi, S. Koka, K. Anada, Y. Miyadera, Takeo Yaku, "A data structure for triangular dissection of multi-resolution images". *Proc. SNPD* 2014, 1-7 (2014).
[6] S. Koka, K. Anada, T. Goto, T. Yaku, "Represented by Homogeneous Triangular Dissections", *Proc. ICIS* 2017, to appear.