한국항해항만학회 제30권 제1호 춘계학술대회논문집 pp. 61~66, 2006. 6

A Study on the Radar Image Generation Method in Ship Handling Simulator

Min. Jung · Sin-Geol Lee** · Chea-Uk Song****

*,** *Graduate school of Korea Maritime University, Busan, Korea*

*** *Division of Maritime System Engineering, Korea Maritime University, Busan, Korea*

ABSTRACT : This paper proposes a method for generating radar images used in ship handling simulator, which includes mathematical logics based on radar equations and information from Openflight format files. In order to make radar image much similar to that of real radar in PPI type, the proposed mathematical logic derives radar video signals under the consideration of not only the data form flight format file of simulation scenes, but also geographical radar's position. The proposed method is considered useful to make radar images in ship handling simulator with accuracy and reality.

KEY WORDS : *ship handling simulator, radar image generation, 2D raw map, openflight, terrain map, 3d database*

1. Introduction

Ship handling simulator, a kind of simulators widely used in maritime education institutes, is designed to improve navigator's ability in ship handling. It is also designed to decide both safety and effectiveness when facilities such as traffic route and harbor are constructed. To achieve these aims, Ship handling simulator should be made under the consideration of both accuracy and sense for the real.

The accuracy of the simulator having sense for the real depends on how exact the mathematical models are used to describe hull's movement at sea as well as the performance of navigational equipment. Its reality could be achieved by three dimensional simulation scenes which were made by VR technology and shown to trainees during simulation. Modern high VR technology has contributed to the reality of those scenes in ship handling simulator

However, in spite of the advance of VR technology, radar image of simulator is still in low level, so trainees have expressed their troubles in getting information from radar image during simulation exercises. Radar image of ship handling simulator bases on radar video signals resulted from calculation of radar performance in three dimensional simulation scenes around the position of radar. Since precise mathematical logic to calculate radar video signal is not so simple and also takes long time to process, most of simulators have used a simple method which considers only information about coastal lines and altitude of terrains in simulation scenes, in disregard of geographical radar's position.

Prior to developing the Radar Simulator, it is necessary to make 2D raw map using 3D OpenFilght data which is the basis information for terrain database. And we are to develop BRI(Basic Radar Image) based on 2D Radar Map.

2. The technique on using Openflight

2.1 About OpenFlight File

OpenFlight is an industry standard real time 3D scene format developed, owned and maintained by MultiGen, Inc. It was originally developed by MultiGen in response to a need for database transport ability within the visual simulation

* 종신회원, immina78@bada.hhu.ac.kr 011)695-5369

** 종신회원, sglee@saracom.net 016)566-7339

*** 종신회원, songcu@mail.hhu.ac.kr 010)6556-9199

community. In visual simulation, OpenFlight is the defacto standard format. The FLT, a form of hierarchical and logical scene description is using in Open Flight.

2.2 Openflight file structure

The OpenFlight database hierarchy organizes the visual database into logical groupings and facilitates real-time functions such as field-of-view culling, level-of-detail switching, and in Each OpenFlight database is organized in a tree structure.[1]

The database tree structure consists of nodes, which historically called beads. Most nodes can have child nodes as well as sibling nodes. In general, nodes can be thought of in three hierarchical classes. Starting from the top of the hierarchy, these three node classes include container nodes, geometry nodes and vertex nodes.

2.2.1 Basic Nodes

The following contents are some important nodes to construct OpenFlight database.

1) Data base Node: This node is located on the top of database. It is automatically generated in case of making new files and the node name is 'db'. It generates a unit of database having information of coordinates and FLT file, including date information. This header is not deleted or selected.
2) Group Node: A group node distinguishes a logical subset of the database. Group nodes can be transformed (translated, rotated, scaled, etc.). The transformation applies to itself and to all its child. Groups can have child nodes and sibling nodes of any type, except a header node. Attributes within the group record provide bounding volumes that encompass the group's children and real-time control flags. The basic group node is starting the name 'g'. The 'g1' and the 'g2', automatically generated, are called the node of master group. All database elements are located under this group node.
3) Object Node: An object node contains a logical collection of geometry. It is effectively a low-level group node that offers some attributes distinct from the group node. The basic object nodes are starting the name 'o'.
4) Face Node: A face node represents geometry. Its children are limited to a set of vertices that describe a polygon, line, or point. For a polygon, the front side of the face is viewed from an in-order traversal of the vertices. Face attributes include color, texture, material, and transparency. The face node is starting the name 'p'.
5) Vertex Node: A vertex node represents a point in space, expressed as a double precision 3D coordinates. Each vertex is stored in the vertex palette record. Vertex attributes include x, y, z and optionally include color, normal and texture mapping information. Vertex nodes are the child of face nodes and light point nodes.

The following nodes are possible to insert for expression of special effects beside the mentioned above basic nodes.

6) Degrees of Freedom (DOFs) node: It is the node having some extent information of movement or action.
7) Levels of Detail (LODs) node : A level of detail (LOD) node serves as a switch to turn the display of everything below it on or off based on its range from the viewer, according to its switch-in, switch-out disand center location.[4]
8) Light sources node: A light source node serves as the location and orientation of a light source. The light source position and direction are transformed by the transformations above it in the tree.
9) Sound node: A sound node serves as the location for a sound emitter. The emitter position is the sound offset transformed by the transformations above it in the tree.

2.2.2 The structure of Database node

The 3D information is divided into a few groups and is combined to the database classes. The FLT makes all works easy by selecting this structure. Precisely, information of all models are saved dividing group node, each part of model is an object node, polygons of the object node are making the class of face node that is consist of vertexes and edges. The

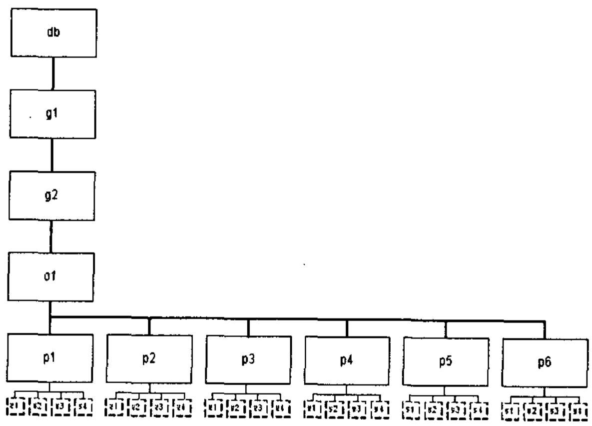

Fig.1 An example, the structure table of Database in OpenFilght file

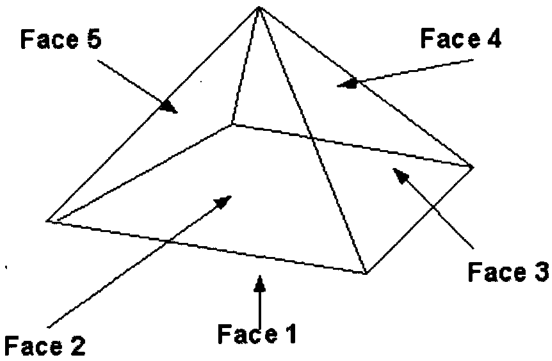

Fig.2 The picture of modeling the pyramid

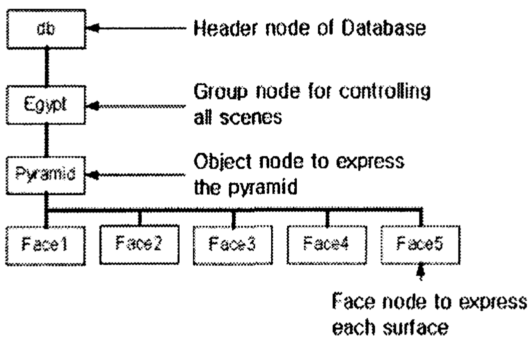

Fig.3 The structure of the pyramid database

precaution to construct the node is the object node having only the face node but comparatively group node can have any node below it.

To show an example in Fig.2, if it makes a pyramid to database, the following results would be constructed.

The pyramid has 5 faces and each face holds a vertex, is contained ID and position information, it would connect faces as sequence of Vertex ID as bellow Fig. 2.

Fig 3 is a picture to draw the database structure of pyramid model. It can add some egypt group nodes beside the egypt group nodes under the head node. There are contained information of each face and vertexes below the egypt group node.

3. Technique of 2D raw map Generation

3.1 The concept of transformation 2D raw map

This chapter shows the method to generate 2D raw map image by adding necessary data, to extract Openflight files using graphic tool shown in the chapter 2.

The above method will be passing 3 steps.

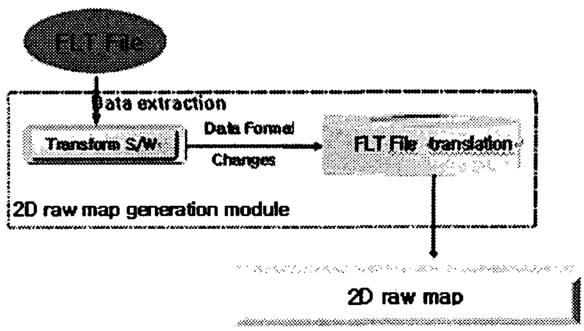

Fig. 4 Transformation processes of 2D raw map

The first step is the procedure analysis, it is reading the openflight files and parsing the all 3D database by each node and then extracting the included real data.

The second step is the converting procedure. It is converting to necessary data for the 2D raw map using extracted data from Openflight files.

The last step is building the BRI using the converted data on step-2.

The following fig. 4 is a simple diagram to generate the 2D raw map image by above steps.

3.2 The procedures of transformation Image

3.2.1 Step 1 - Analysis of Openflight files

This procedure is the Openflight file analysis; it is parsing nodes that are made the Openflight database to the tree structure and extracting the data having each node. In here, it is classifying all nodes to essential data and useless data for building the 2D raw map image. That is, the essential polygon data, is the number of 3D's polygon and the coordinates value of each point, to draw the 2D raw map

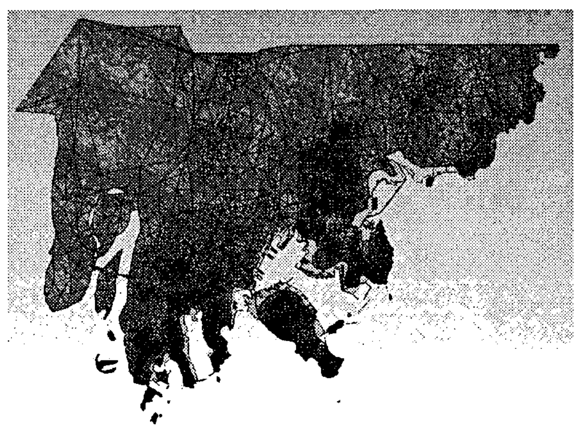

Fig.5 The OpenFlight image of Busan port

Fig.6 Processed image by the stepl

image are extracted in all nodes the nodes containing the information of movement and sound are deleted.

In this step, explaining the example by Busan port images. The Fig. 5 is the 2D image of Busan port for using formation s/w. The fig 6 is an image after passing stepl. It is deleted textures of the surface and remained the altitude, the coordinates(x, y) information. And then the Busan port 3D image file is spilt into small polygons.

3.2.2 Step2 - Converting to divide into dots.

The coordinates Z of each polygon extracted in stepl means the altitude. But the information including basic polygons isn't generating precise images. Therefore, it is to generate more precise terrain data by inserting additive information of Stepl to the basic polygon.

The following fig. 6 shows an example; in case of the left polygon has only 3 points information, but if it is added point information as the right one, we can generate more precise information of terrain data.

When it is adding some points on each polygon, the interval of each point is decided as the following.

Assuming the radar screen resolution is 1024 pixel x 1024 pixel and the minimum detection range is 0.75 nautical mile, the display distance of the screen is the following.

D = range x 1852m = 0.75 x 1852 = 1389m

G = 1389m/512pixel ≈ 2.5

The value of D is 1389m. If the fixed D expresses 512 pixels, the interval of each point (G) is to be 2.5m. In here, the polygon is added G by 2.5m interval.

1.1.3 Step3 - Generation of the 2D raw map

The step3 shows the 2D raw map image using the

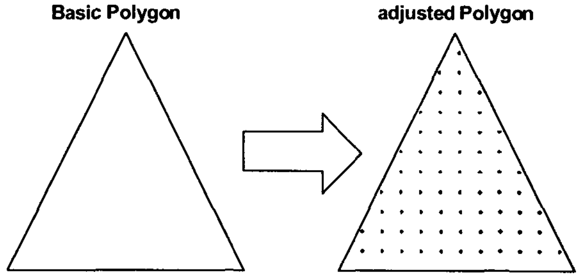

Fig. 7 Example polygon adding additional points

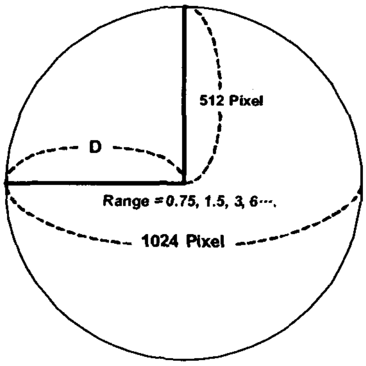

Fig. 8 Deriving the distance of radar screen

converted Openflight data of step2. In this time, it is fixing the RGB value of an interval of 5m unit using the height of each polygon's center. And the printing of image is drawn by OpenGL. Therefore, the image will be displaying as following Fig.9.

The fig.9 is the origin openflight file for stepl previous processing image. And then the image has processed stpel to remove the texture information. It remains only polygon's

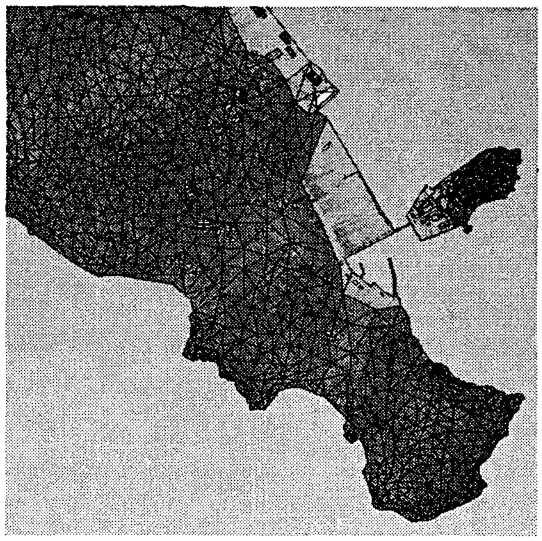

Fig. 9 The origin Openflight file for step 1

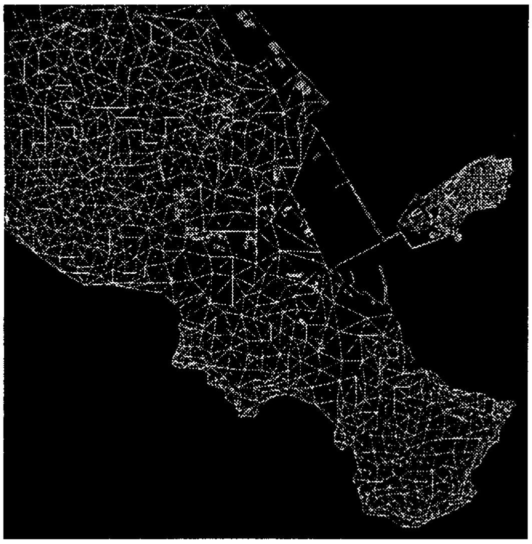

Fig. 10 The processed image of step1

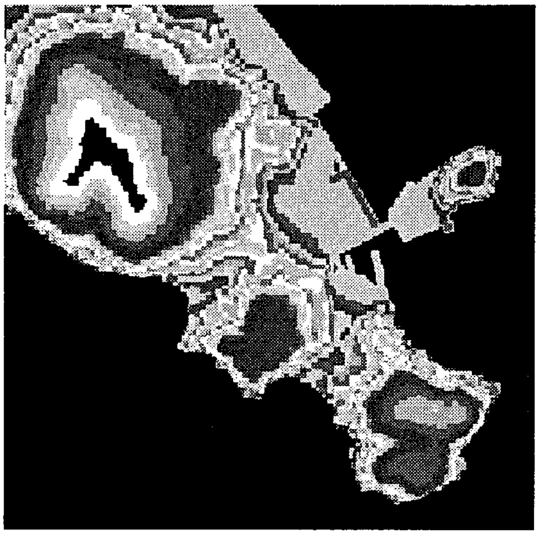

Fig. 11 2D raw map image after processing step2

coordination and altitude information after processing the step1. The processed image after processing step1 is shown in Fig. 10.

After processed step 1, the image is passed step 2, and then adjusted information is used making for generation of 2D raw map. Through these steps, the Fig. 11 of 2D raw map image was attained. The 2D raw map image expressed to various colors. It is easy to distinguish the altitude from color differences. Each color has the information of altitude. It will be using to generate Basic Radar Image.

The following Fig.12 draws the embodiment for building 2D raw map.

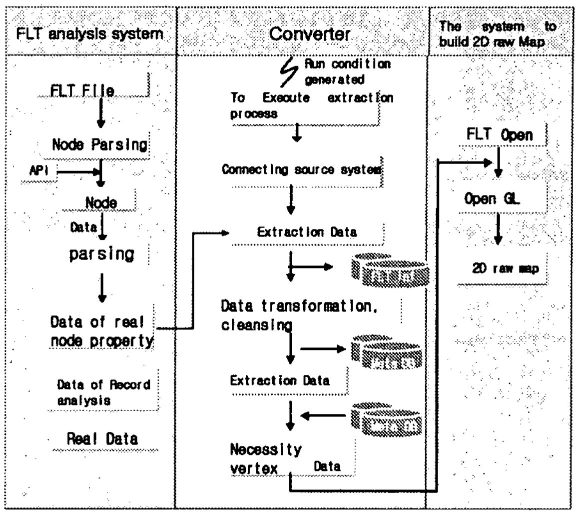

Fig. 12 The embodiment chart for building 2D raw map

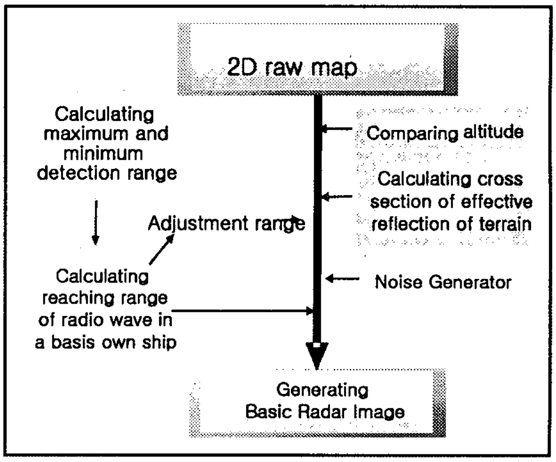

Fig. 13 The outline of generation BRI from 2D raw map

4. Building Basic Radar Image

It is building the BRI(Basic Radar Image) using the altitude information and the position(Latitude, Longitude) after generating 2D raw map. The fig. 12 draws the outline of generation BRI from 2D raw map.

The maximum detection distance is calculated by the following formula for drawing a real radar image similarly. [2]

$$R_{max} = \left[\frac{p_t \lambda^2 G^2 \sigma}{(4\pi)^3 S_{min}} \right]^{\frac{1}{4}} \quad \cdots\cdots (1)$$

Where, R_{max} = maximum detection range

P_t = transmitter power (watt)

G = antenna gain

σ = target radar cross-section (m^2)

$S_{\min}$ = minimum detectable signal (watt)

λ = wave length (m)

In here, the minimum detection signal is derived by the following equation. [3]

$$S_{\min} = \frac{P_t \lambda^2 G^2 \sigma}{(4\pi)^3 R_{\max}{}^4} \quad \cdots\cdots (2)$$

Also, the cross section of effective reflection is calculated by the following equation for displaying the reflection intensity and the shape of 3D terrain in Openfilght file. The cross section of effective reflection of normal target nearly is not according with the physical area of the target. But if the target is large, the cross section of effective reflection would be too large in proportion to this.

$$\sigma = 4\pi \lim_{R \to \infty} R^2 \frac{|E_r|^2}{|E_i|^2} \quad \cdots\cdots (3)$$

Where E_r = reflected field strength

E_i = beamed field strength

The radar simulation image is generated by above equations and information of generated BRI.

The following fig.14 is the first stage of generated BRI. In the Fig. 14, the radar scanner position is the yellow circle.

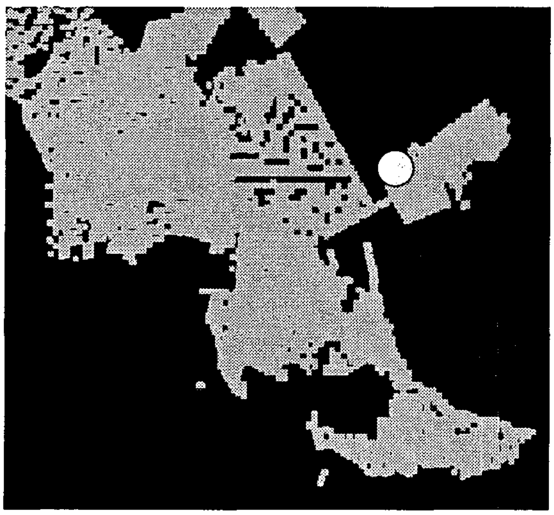

Fig.14 Generated BRI

5. CONCLUSION

In this paper, the authors described the process of building 2D raw map as follows;

(1) Step-1: Analysis of Openflight files of 3D terrain image.
(2) Step-2: Converting the 2D raw map from Openflight files
(3) Step-3: Generation of 2D raw map using the converted data

Passing 3 steps, we used the Busan terrain database and then acquired its 2D raw map image.

Then, the BRI was generated by applying radar equation and basis of 2D raw map.

In above processes, it is ready to making radar image more similar with real radar scope.

As the result, this BRI is a basis image of radar simulator; we are to develop it similar with real radar image next study. That time, we will consider some noise of reflector by developing a noise generator for the radar image having a sense for the real.

REFERENCES

[1] OpenFlight Scene Description Database Specification, Multizen paradigm(http://www.multigen-paradigm.com)

[2] Sang Sul Lee, "The technology of radar", Korea, p 61~p 103

[3] A.G. Bole, Extra Master Mariner, FRIN, FNI, W.o. Dineley, Extra Master Mariner, Mphil, "Radar and ARPA Manual", p 126~ p 138

[4] Guozhu Jin, ChaeUk Song, "A new approach of making terrain database in visual scene system of marine simulator", 2004