

DIGITAL TERRAIN AND AUTOMATED MAPPING SYSTEM DEVELOPMENT USING UNIFIED MODELLING LANGUAGE (UML)

Alias Abdul-Rahman¹ and Jae-Hong Yom²

¹Institute for Geospatial Science and Technology
Universiti Teknologi Malaysia
81310 Skudai, Johor,
Malaysia
alias@fksg.utm.my

²Department of Geoinformation Engineering
Sejong University
98 Gunja-Dong, Gwangjin-Gu, Seoul,
South Korea
jhyom@sejong.ac.kr

ABSTRACT:

Automated mapping and digital terrain system could be developed by using several approaches where an object-oriented technique is one of them. The technique has been considered by many software engineers and developers as one of the most efficient approaches. This paper discusses the design and development of digital terrain and automated mapping system using the Unified Modelling Language (UML). The design and the development aspects will be highlighted and forms major discussion of the paper. The digital terrain modeling component is based on Triangular Irregular Network (TIN) data structure. We will demonstrate the system by using photogrammetrically captured data. The application of the system will be demonstrated and finally we will highlight the system's outlook.

KEY WORDS: Modelling, Design, Automation, Software, DEM/DTM, GIS

1. INTRODUCTION

UML is a notation set for building object models. It is the standard for communicating the meaning of object modeling diagrams throughout the industry. UML is a language for designing a conceptual schema using object-oriented programming (OO) concepts, and enhancement of the widely used Entity Relationship Model (ER Model). It is currently one of the most common languages for conceptual data modeling and system development. We could utilize this tool to model any GIS spatial modeling tasks as well as for mapping and digital terrain problems. Basic concept of the UML modeling is not discuss here but rather a discussion how to use and implement a system based on the language. Readers may refer to several texts on the basic of UML such as Wood (2002) and Laurini (2001). Here we focus on the design of the conceptual data model, the first step in the modeling process.

This paper addresses the development of automated mapping in Section 2 and digital terrain using the modeling language (UML) in Section 3. Finally, we conclude the experiments and the system outlook.

2. UML BASED AUTOMATED MAPPING SYSTEM

An experiment was done at the Glasgow University for the development of an automated mapping system by utilizing the UML tool (Yom, 2000).

An **automated mapping system** is defined as a composition of subsystems namely; the image acquisition subsystem, positioning subsystem, triangulation subsystem, object

extraction subsystem and the visualisation subsystem. A practical level of automation is assumed where human interaction is still required at many stages.

2.1 Image acquisition subsystem

The image acquisition subsystem of the automated mapping system, involves a human operated platform, such as a fixed wing airplane or a helicopter. The imaging sensors are a key component of the imaging subsystem. Some examples of imaging sensors are CCD digital cameras, video cameras, linear array scanners, laser profiling scanners and multi-spectral scanners. Use case diagram of the image acquisition subsystem is as shown in Figure 1.

2.2 Positioning subsystem

In an automated system the imaging sensors of the image acquisition subsystem (e.g. CCD cameras) are synchronized with the positioning sensors to compute the attitude and the position at the instant of the image capture. Some of the most popular positioning sensors in use today are GPS (Global Positioning System) receivers and IMU (Inertial Measurement Units). The GPS receivers provide the positional coordinates of the exposure stations whereas the IMU provides the attitudes of the images at the instant the exposure. Figure 2 shows the use case diagram of the positioning subsystem.

2.3 Triangulation subsystem

Triangulation is defined as the task of locating the ground position of any points seen on the image by using their observed image coordinates and the values obtained from the

positional sensors. The main reason for the aerial triangulation process is to reduce the cost of ground control point surveying by replacing it with computed coordinates from image observations.

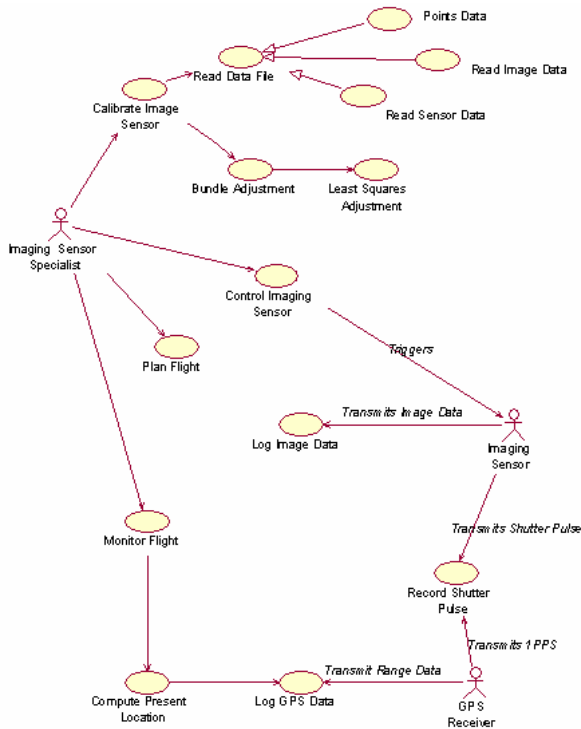


Figure 1. Use Case Diagram of the Image Acquisition Subsystem.

During the aerial triangulation process, the operator identifies the surveyed ground control point in the images in which they appear and measures the image coordinates of these points. These measurements and the surveyed coordinates are then fed into an aerial triangulation program to compute the coordinates of the perspective points as well as the attitudes of the observed and other connected images at their instants of exposure. These (i.e. the image coordinates, the perspective centre coordinates and image attitudes) are then used to compute ground coordinates of any points on an image.

In the triangulation process of an automated mapping system, the perspective point coordinates and the attitudes, are not computed indirectly, but are observed directly using the positional sensors. In theory, the triangulation process would not involve any ground control point surveying.

Aside from the fact that the perspective coordinates and the attitudes are directly observed, the setting up of the observation equations and its adjustment is very similar to the traditional aerial triangulation task.

2.4 Object extraction subsystem

Object extraction refers to the interpretation and recording of objects from images. Automation of such tasks is still a challenge to many researchers in both the photogrammetric field and the computer vision field. Object recognition techniques are used by the photogrammetrist to capture the semantic information at a certain location and populate the GIS database, which is identical to the task of the human stereoplotter operator. The automation effort in this field involves research into image segmentation, feature extraction from images and grouping extracted features such as points and edges. Detection and interpretation of simple features such as road centrelines has been successful, but other spatial objects, buildings (i.e. in particular, are still being researched

(Roux et al., 1994), (Gruen et al., 1996), (Boichis et al., 1998), (Haala et al., 1998).

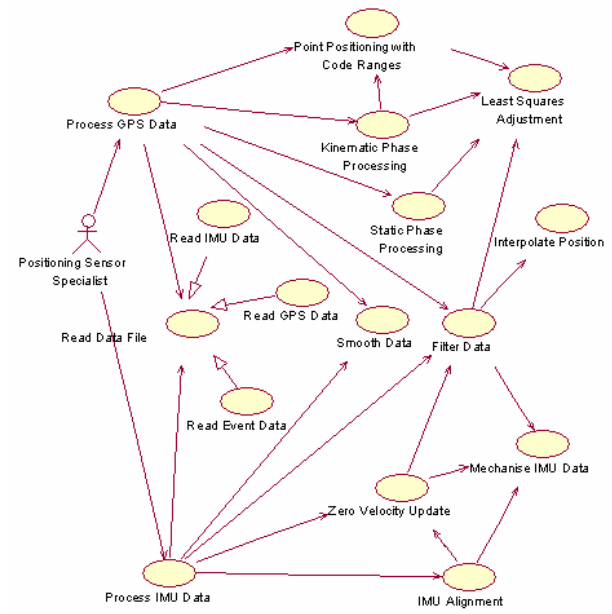


Figure 2. Use Case Diagram of the Positioning Subsystem

2.5 Visualization subsystem

Schroeder (1998) defines visualisation as "... the process of exploring, transforming, and viewing data as images (or other sensory forms) to gain understanding and insight into the data.". The concept of automation and the involvement of computer processing is inherent in this definition of visualisation. This definition resembles closely the activity of a cartographer, except that a cartographer deals mainly with geospatial data.

The end product of a visualisation process in a mapping system might be a three dimensional perspective view of a landscape processed from the digital terrain model and a scanned aerial image of that area. The images of the mountains could also be marked with contour lines, streets could be labeled with street names and commercial buildings could be coloured red. In other words, the traditional cartographic process of symbolising information on paper maps is incorporated into the visualisation process as symbolising information on three dimensional image views.

The bundle adjustment use case of the triangulation subsystem was implemented as shown in Figure 3. The figure shows text files opened in three child windows each showing the attributes of the images, control points and the imaging sensor respectively.

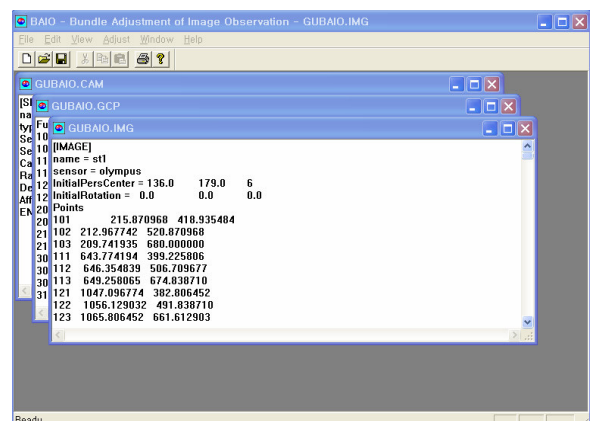


Figure 3. Implementation of the bundle adjustment use case.

Graphical User Interface (GUI) is another aspect to be improved in the automated mapping system using software engineering technology. An improved GUI version of the bundle adjustment could be as shown in Figure 4, where the control points, image boundaries are graphically demonstrated, instead of showing them as texts and numbers.

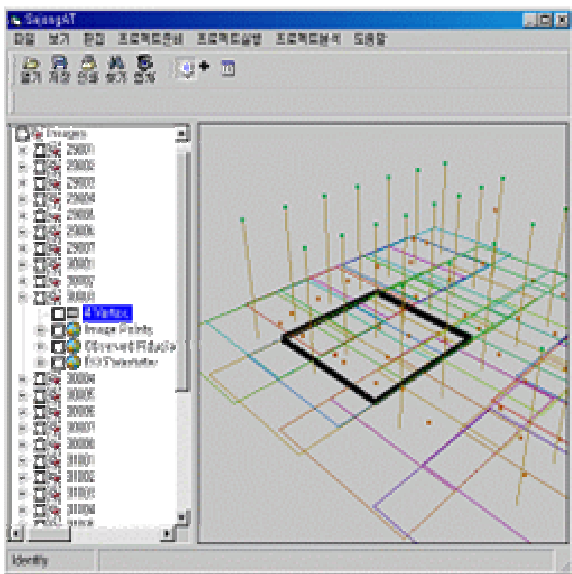


Figure 4. Improved GUI for bundle adjustment

3. UML BASED DIGITAL TERRAIN MODELLING SYSTEM

In this section we describe the development of the system that utilizes UML as a modeling tool. The digital terrain system is based on DTM data structure called Triangular Irregular Network (TIN). It is a popular structure in DTM domain – offers several advantages over other structures like grid. One of the advantages is it can accommodate linear features that exist on the terrain quite well. Abdul-Rahman (2000) constructed a system based on the Booch modeling tool (Booch, 1994) and could be extended by using the UML as describe below.

In this section we discuss the use of UML for constructing 2D and 3D TIN. First, the 2D TIN – the module has three major operations, they are Distance Transformation (DT), Thiessen polygonization, and 2D TIN topology. Figure 5 shows the class diagram of the operations.

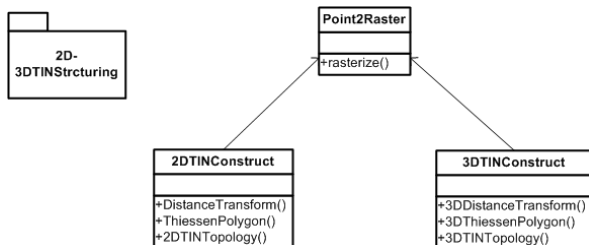


Figure 5. The UML class diagram for the 2D and 3D TIN data structuring.

The results from the 2DTINConstruct and the 3DTINConstruct classes, i.e. the 2D and 3D TIN structured data become an input to the Digital Terrain system. The modelling and the development of the system is discussed in

the following sections. Here, we provide a discussion of the OO TINs spatial data modelling techniques. Conceptually, the general modelling steps (the conceptual, the logical, and the physical) could be used for TIN spatial data modelling. Others called it as the three-step approach. The class schema for spatial data modelling are described below.

3.1 The Class Schema

The schema is based on several classes, they are Spatial Objects (the super class), and four major subclasses which are Node, Edge, Polygon, and Solid.

Spatial objects

The spatial object class is a general class of the real world objects. It is the super class in the class hierarchy. We assume that all other objects are derived from this super class. All terrain objects could be categorised into several sub classes such as points, lines, areas, and solids (volume) features. In OO modelling, these feature types are the classes in the modelling hierarchy.

Node

A node can be considered as the most basic geometrical unit in spatial data modelling. It may represent point entities or point objects at a particular mapping scale. Examples of point objects are wells, terrain spot heights, and the like.

In geoinformation, we may represent these objects by a class called a node class. The coordinates of the nodes (including the nodes represent edges) are held by a coordinates container class, called XYZContainer class.

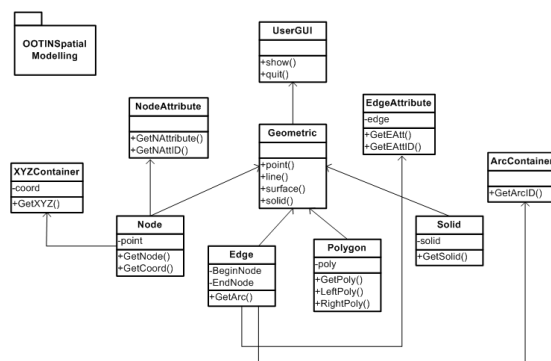


Figure 6. UML class diagram of the system functions.

Edge

An edge can be represented by two nodes at each end (i.e., a start node and end node). In this study we consider two end points make a straight edge. We used this edge type to represent linear features. The arc container class, called ARCContainer holds all the arcs. The arcs container also serve any other class which requires arcs data in their operations for example the polygon class needs the arcs in order to form polygons.

Polygon

A polygon (sometimes known as a surface) is used to represent area features such as lakes, ponds, etc. A polygon may be constructed by chains of closed edges.

Solid (or Body)

This is a representation for solid or body features such as buildings, trees. A chain of points and lines form body objects for example, 3D TIN can be represented by a series of triangle nodes and edges.

The class schema in Figure 6, depicted using UML notation is the representation of the TIN spatial data model. The schema has: four geometric classes namely (TNode, TEdge, TPoly, and TSolid); two types of containers (geometry and attribute). The geometric containers contain the XYZ locations whereas the attribute containers are for the thematic values, e.g. names.

The following figure shows the results of the OO TIN UML based subsystem.

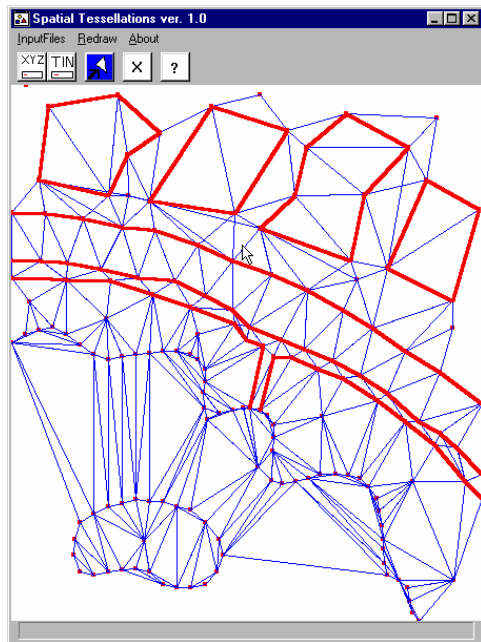


Figure 7. The terrain features display of the query.

4. CONCLUSIONS

The automated mapping and the OOTIN systems are two separate subsystems. Both systems were developed by using the UML as the modeling tool. The experiments that we performed have produced an acceptable mapping output, that is to say the UML served as reliable modeling tool for system developers. We realized that the system could be easily developed and combined since the UML permits the team development effort and reusability of the developed mapping and digital terrain modeling classes. It is our aim to enhance the existing subsystems or results into one integrated automated mapping package that based on UML.

REFERENCES

- Abdul-Rahman, A. (2000). The design and development of two and three-dimensional triangular irregular network-based GIS. *PhD thesis*, Department of Geography and Topographic Science, University of Glasgow, United Kingdom, 250 p.
- Boichis, N., Cocquerez, N., and Airault, S. (1998) A Top Down Strategy for Simple Cross Roads Automatic Extraction. *Proceedings of the Commission II Symposium : Data Integration Systems and Techniques*, XXXII (Part 2), Cambridge, United Kingdom, 19-26 p.
- Booch, G., (1994). Object-oriented analysis and design – with applications. Addison-Wesley. 589 p.

Gruen, A. and Li, H. (1996) Linear Feature Extraction with LSB-Snakes from Multiple Images. *International Archives of Photogrammetry and Remote Sensing*. XXXI (B3), Vienna, 266-272 p.

Haala, N., Brenner, C., and Statter, C. (1998) An Integrated System for Urban Model Generation. *Proceedings of the Commission II Symposium : Data Integration Systems and Techniques*, XXXII (Part 2), Cambridge, United Kingdom, 96-103 p.

He, J., L. Li, and M. Deng. (2002). The design of urban underground pipeline GIS based on UML flexibility software developing model. *IAPRS*, Vol. 34, Part 2, Comm. II, Xiang, China.

Laurini, R. (2001). Information systems for urban planning. Taylor and Francis, 349 p.

Roux, M. and McKeown, D. M. (1994) Feature Matching for Building Extraction from Multiple Views. *Proceedings of the ARPA IUW*. Monterey, California, ARPA. 331-349 p.

Schroeder, W., Martin, K., Lorensen, B. (1998) "Introduction," *The Visualization ToolKit : An Object-Oriented Approach to 3D Graphics*. Prentice Hall, 1-15 p.

Wood, J. (2002). Java programming for spatial sciences. Taylor and Francis, 320 p.

Yom, J.H. (2000). The design and development of automated mapping system – image handling. *PhD thesis*, Department of Geography and Topographic Science, University of Glasgow, United Kingdom, 200 p.