A GIS Platform for Automatic Navigation into Georeferenced Scenes Using GIS Scene Explorer (GIS-SE)

Christos P. Yiakoumettis, Georgios Bardis,Georgios Miaoulis,
Dimitri Plemenos, and Djamchid Ghazanfarpour

Abstract. During the last decade the rapid evolution of GIS systems has led to enchanced accessibility and increased availability. The traditional content has been enriched by multimedia information and three dimensional models. Virtual copies of the real world have been created, widely known to the average user, yet the information retrieval and the efficient exploration of 3D large scale scenes are still pose serious problems. In this paper, a GIS platform for automatic navigation is presented, using a prototype application, entitled “GIS-Scene-Explorer” or GIS-SE, addressing these problems. Our platform is based on an existing GIS platform. Google Earth is used for this purpose. The prototype is a standalone application, adopting the Google Earth API to retrieve and visualize data. It takes advantage of both public remote database of the GIS system and a custom, collaborative database. Three dimensional models of buildings and multimedia data are stored to the collaborative database, operating under a server – client model architecture.

Keywords: automatic navigation, virtual globe, GIS systems, scene exploration, virtual tour, scene understanding, visualisation, public participation, city modelling, urban planning.

1 Introduction

The first attempts and ideas of GIS platforms were introduced in the early of 1980s, but the technology limitations of that time shrunk their popularity. After decades commercial GIS platforms with real geo-referenced scenes have been released. These

Christos P. Yiakoumettis · Djamchid Ghazanfarpour
Université de Limoges,Laboratoire XLIM, CNRS, UMR 6172
83, rue d’Isle, 87000 Limoges France
e-mail: {djamchid.ghazanfarpour@unilim.fr}

Christos P. Yiakoumettis · Georgios Bardis · Georgios Miaoulis
Technological Education Institute of Athens
Department of Informatics
Ag.Spyridonos St., 122 10 Egaleo, Greece
Tel.: (+30) 2 10 53 85 826; Fax: (+30) 2 10 59 10 975
e-mail: {chgiak@teiath.gr, gbardis@teiath.gr, gmiaoul@teiath.gr}

D. Plemenos & G. Miaoulis (Eds.): Intelligent Computer Graphics 2010, SCI 321, pp. 105–122.
springerlink.com

applications are known as *virtual globes*. Most of them offer a three dimensional environment, a representation of the real world's surface. For the reconstruction process aerial and satellite photographs were used (Zhang, et al., 2007). Dedicated databases are used to store the data. Data are calibrated on a referred positioning system. In most cases the global positing system is used for that purpose. The data include hypertext, photographs three dimensional models and other georeferenced data. Commercial applications have been released by Google (Google Earth), NASA (NASA World Wind), Microsoft (Virtual Earth 3D), open source community and more, comprising more than simple mapping applications since they represent a 3D morphological model of the surface of the earth. Beyond the commercial approaches there are plenty others from open source community. They include information about the streets, points of interest, 3D models of buildings and multimedia content (Jones, 2007). Each one is using its own two or three dimennsional environment for visualizing the data in the form of a browser.

Virtual Globes applications support a 3D or 3D graphical interface. The user can freely explore the scene and "fly" among the obstacles. However the navigation on the scene can only be manually accomplished by keyboard and mouse. On the other hand the size and the complexity of the whole scene make the task of manual navigation and the insight gained by it generally inadequate. Moreover the user's incomplete knowledge of the scene increases the inefficient of manually exploration of 3D scenes.

A GIS platform for automatic navigation into a 3D georeferened scene will be presented. All required data of the 3D scene are stored in a collaborative database. The basic architecture supports multiple databases and it is based on an existing GIS platform. It may also serve as the platform for future features, for example planning personalized paths of automatic navigation into scenes, considering the user's preferences, profile and requests.

2 Virtual Globes

2.1 Simple Two Dimensional GIS System Architecture

One of the first virtual globes platforms, that became widely popular, was released By Google in 2004. It was Google Earth. A three dimensional environment is offered, that allows user to browse, navigate and explore 3D model of the scene. It is based on server-client GIS system architecture. A typical web browser is used on the user's side. The typical scenario is initiated by the client requesting data based on user's position in the virtual scene. In response to the request, the data is placed on the server side in a GIS database and subsequently the retrieved data are visualized on client's side with the aid of a specialized application. Initial releases offered a two dimensional environment (Wikipedia, Google Maps), (Wikipedia, Virtual Globes). Google Earth started publishing static maps, evolved to static web mapping applications and then to GIS distributed systems (Sriphaisal & Pujari). The basic functionality of a server-client GIS system is presented in figure 1. This block diagram presents the functionality of Google maps, a two dimensional GIS platform.

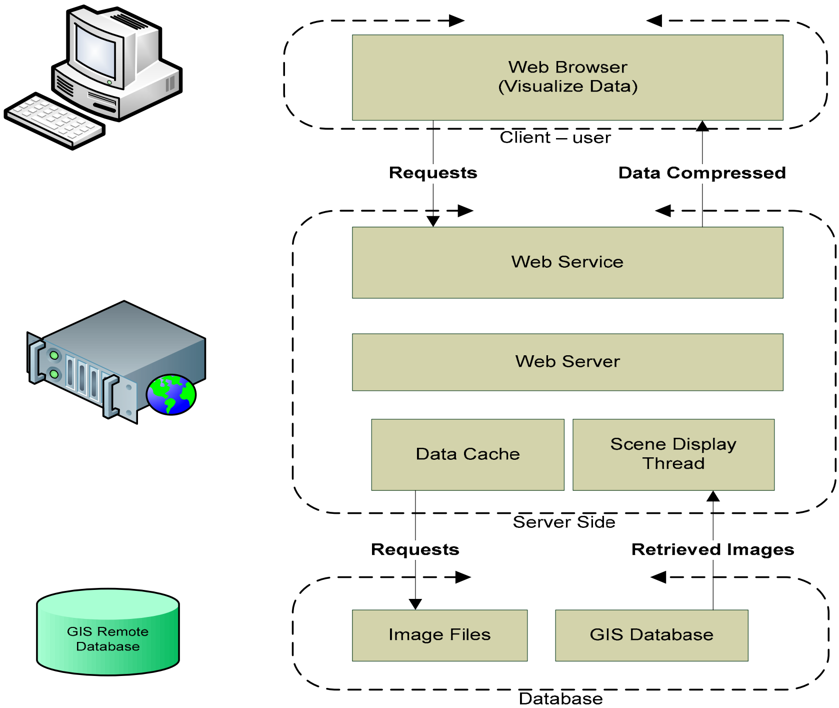

Fig. 1 Block Diagram of two dimensional GIS System

Requests from the users were submitted to the server. Communication between client and server was accomplished by using AJAX and XML technology. The user was requesting data depending on its position (longitude, latitude and altitude). The requested data were mined either from a temporary cache memory of by the georeferenced database. The retrieved information was sent back to the user's side in compress format and it was visualized on its side. The web server was used for the communication between the client's visualizing application and the georeferenced database.

Satellite or aerial photographs have been used to provide the necessary database content and the global positioning system has been used to calibrate the images (figures 1 – 2).

2.2 *The Evolution of GIS Architecture*

Google maps became known to general public. The quality of the services and the amount of information were increased by the time.. Initially, additional information was added to the database based on two dimensional aerial photographs. The names of the streets and the roads were the first data that were added to the database. Afterwards, metadata such as text, external links, hypertext, and placemarks enriched the database.

Despite this evolution, the fundamental architecture has remained practically the same (figure 2). All the data were imported to databases and calibrated over the same geographical positioning system.

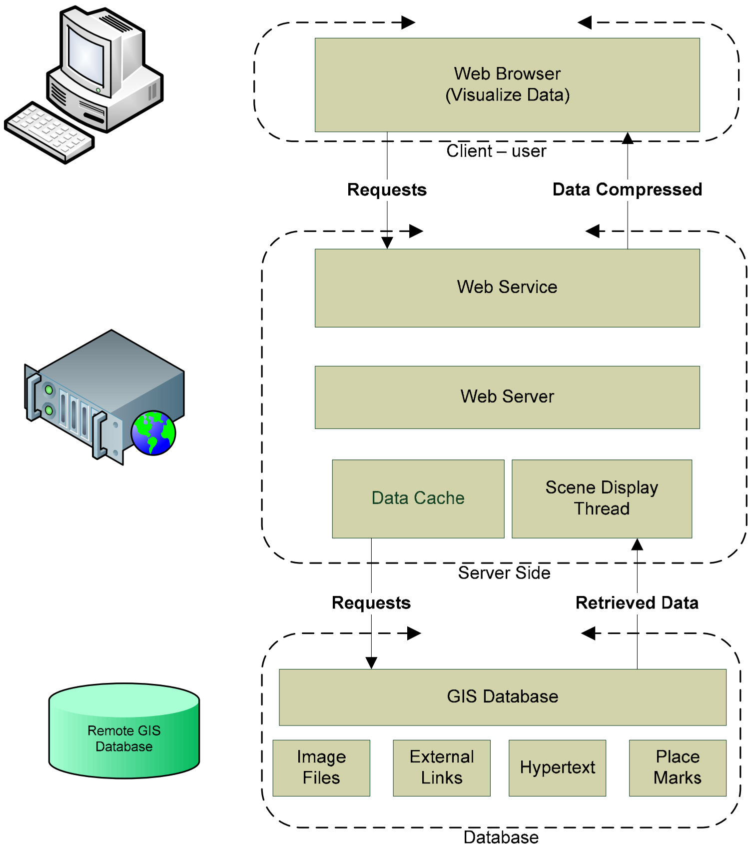

Fig. 2 Block Diagram of improved GIS system

The data repository of the system is comprised by four different resources, each of them potentially forming a separate database. All databases have to use the same geographic positioning system. This architecture allows the system to be very flexible and easily distributed to the extent that, for example, each database may be hosted by a different provider. Moreover, there is no limit on the number of physical databases that may constitute each of the aforementioned resources.

Beyond the two dimensional platforms, new browsers was presented with 3D exploration capabilities. Apart of images and hypertext, information about the morphology of the ground was added. In addition 3D models of buildings were enriched the database (figure 3). Initial models of buildings were limited and only specific cities were modeled.

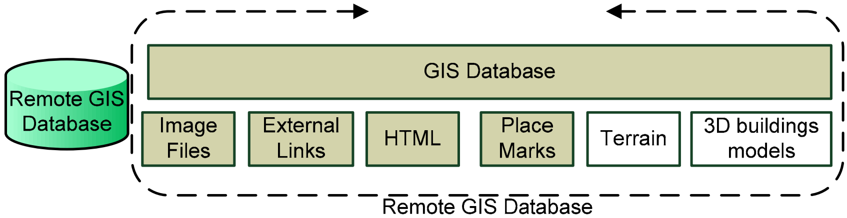

Fig. 3

3D models were designed and built in a 3D object description language called COLLAborative Design Activity (COLLADA). It has a language with XML schema and it has been adapted by many software and hardware developing companies.

On the user's side a standalone application or web browser with visualizing capabilities of COLLADA is required. Components have been developed by distributors for this purpose using OpenGL or DirectX graphics languages.

2.3 Recent GIS Platforms

Recent years many databases with georeferenced content have been released. The popularity of such systems and the advancements in computer technology have made possible the application of three dimensional environment on GIS systems, based on the above architecture. Information about the altitude and the morphology of the ground has been added to the database. In addition, 3D models of buildings are also being gradually added (figure 3). The goal is to create a virtual world that simulates the real one. There is no limit on the data that they can be added. There are GIS databases that are freely available for public use, whereas some others are not.

Data are requested based on user's position and preferences. Users can filter the data using the GIS platform interface. Data might be organized in dedicated databases. The user's application is presenting the data organized in layers. Each layer contains data from a different database. When all the data are being presented, a synthesized image is been created from the projection of the layers onto the same geographic position system. The result is a synthesized three dimensional scene.

The latest versions of user's application have the ability to cache the data, or even to load external data stored on local user's computer. The local data give the ability to create and use a custom local database. The user's client can be used as viewer or browser, also visualizing the local data.

3 State of the Art

In the field of geo-referenced and navigation systems, several methodologies have been introduced aiming to solve individual problems.

3.1 Geo-referenced Systems

One well-known approach is using a topic that represents the context of the image in order to classify geo-referenced images. The topic produced by a probabilistic latent semantic analysis (pLSA), a well-known technique in the text-mining community. Recently vocavulary was extended to describe, in a compact way, the contents of an image database (Bosch et al., 2006). The input of the probabilistic latent semantic analysis is a set of images and their location. The analysis classifies the images (Cristani et al., 2008).

In the field of geo-referenced databases, there are many distributed systems that have been developed. Most of the approaches are using a standard database schema with some extra geographical data for each object. On the other hand, the first implementations of spatial query languages were in 1980's. Borrmann Andre and Rank Ernst have developed a 3D spatial query language for building information models. The language is based on directional relationships among the buildings. The relationship between two buildings was calculated by the euclidean distance of the centers of the objects and the relative position of them, such as *northOf*, *southOf*, *eastOf*, *westOf*, *above* and *below* (Borrmann & Rank, 2008). Recently, the vocabulary of the language was extented and topological relationships were added such as *within*, *contain*, *touch*, *overlap*, *disjoint* and *equal* (Borrmann & Rank, 2009).

3.2 Generation of 3D City Models

There are several approaches on 3D city models generation. It can be accomplished eigher by reconstraction or by generation process based on rules.

Fig. 4 3D Large Scale Scene of 17.362 buildings (Larive & Gaidrat, 2006)

According to the first method, an efficient approach has been implemented using photogammetry. A set of sensors are scanning the building and the scene (Koutsoudis, et al, 2007). The representation of the model is very accurate but experts, special equipment and heavy computational power are required.

Another approach is based on LIDAR (Light Detection And Ranging) (Lan, et al, 2009). It is an optical remote sensing technology that measures properties of scattered light to find range and/or other information of a distant target. The prevalent method to determine distance to an object or surface is to use laser pulses.

On the other hand the 3D model can be generated by an engine. A declarative engine for generating a 3D large scale scene was introduced by Larive & Gaildrat (Larive & Gaidrat, 2006) – (figure 4).

In addition several commercial applications have been presented in 3D reconstruction of georeferenced scenes. They can generate the model based on rules and on image recognition of satellite and aerial photographs.

3.3 Scene Understanding and Exploration

Several approaches have been implemented on scene understanding and exploration. The exploration of a virtual environment on behalf of the users can be classified in three distinct categories: (a) navigation, presenting data from the surface of the world, (b) presentation of an object or a path and (c) presentation of an object or item from specific views in order to reveal as much information as possible (Tab et al., 2001).

The most interesting methodologies are based on the last approach. Scene exploration can take place in two steps. In the first step, the best points of view are calculated and then in the second step, the path of the camera is designed based on the best views of the previous step.

An efficient approach for the calculation of the best view of a 3D scene has been presented by Sokolov and Plemenos. It is based on the coverage of the surface of the scene (Sokolov & Plemenos, 2005), (Sokolov & Plemenos, 2007), (Tab et al., 2001). This approach is considering the complexity and the total number of surfaces of the scene. Each view is evaluated individually and then the best views are selected for the trajectory of the camera.

In addition, Alcorer presented in 2003 a methodology for quality measurement of a certain view of a scene. It was based on information that can be extracted from a specific aspect of view of an object (Alcorer, 2003).

Another approach is trying to optimize the area of the visible surface comparing to the sum of the area of the surface of the whole scene (Sokolov & Plemenos, 2005), (Sokolov & Plemenos, 2006), (Sokolov & Plemenos, 2007), (Tab et al., 2001).

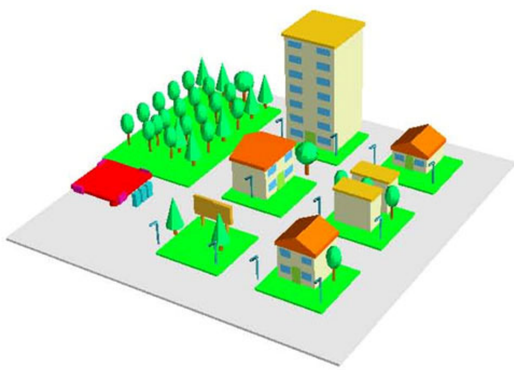

Fig. 5 Initial urban scene

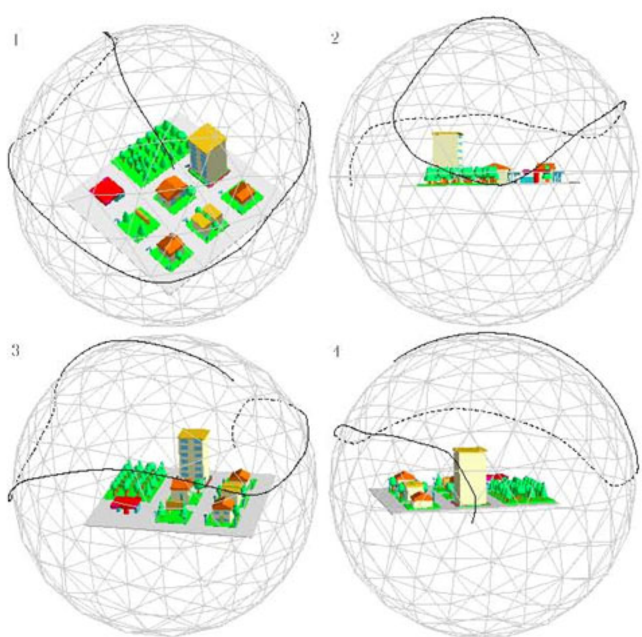

Fig. 6 Camera trajectory of an urban scene (Sokolov & Plemenos)

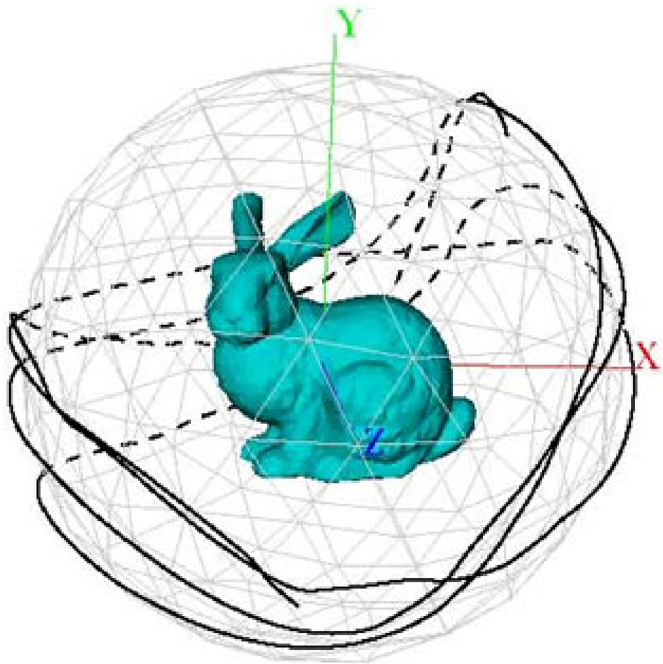

Fig. 7 Camera trajectory of rabbit (Sokolov & Plemenos)

Each object of the scene, either 2D or 3D can be presented from a different point of view. The angle of viewing an object in a three dimensional space requires five variables to be defined. The first three were used to define the position of the camera on an orthogonal 3D coordinate system and the other two were used for the horizontal and vertical orientation of the angle of view. Each point of view is evaluated individually. All possible points of views have been marked on the figures, where the points of the net of a virtual sphere are shown (Sokolov & Plemenos, 2007).

In the figure 5 the whole scene is presented. The best points were being considered for the trajectory of the camera as it was presented on the figure 6 from different aspects of view. In the figure 7 another example is presented using the best viewing algorith of Sokolov and Plemenos.

Moreover, Mackinlay introduced another approach of path planning and navigating in a 3D scene. It is based on points of interests (Mackinlay et al., 1990), (Ruger et al., 1996), (Wolf, 2000). The path is planned considering the user's desired points of interest. The path has to be calculated based on a set of parameters. Each parameter is defined by a weight or a degree of interest (Lucas & Desmontils, 2007). Measuring the various parameters and their weight, it is possible to optimize the information that can be gained following the designed path. The parameters may be pre-defined by the system or defined and evaluated by the user. The evaluation can be done in several ways. A weight value or a degree of interest can define each point of visit, representing the importance of it, compared to the others (Bonnefoi & Plemenos, 2002).

In addition, methodologies of moving the camera on a virtual sphere have been published. In these approaches the object can be watched closer, either by user's intervention or by the system automatically (Tab et al., 2001).

Finally, the last approach is inspired by imagination and, in particular, the answer to the question: "how would it be if we could see the world from the eyes of the interesting object?" (avatars).

4 Proposing a GIS Platform for Automatic Navigation into a 3D Scene

4.1 Motivation

The massive use of simulators and virtual globe applications makes necessary the design and develop of an intelligent platform for evaluation and automatic navigation into the scene.

The complexity of the scene, the amount of information and the lack of knowledge of the surrounding makes difficult or impossible to navigate efficiently into a 3D large scale scene. Applying efficient techniques and methodologies that were presented in previous sections, our proposed intelligent system can be achieved. In the figure 4.2.1 the main architecture of our purposed platform is presented.

4.2 Functionality of the Proposed Intelligent System

Considering the above methodologies and approaches, it is possible to design *an intelligent platform that can evaluate a number of parameters, design a path and automatic navigate into a three dimensional environment presenting all the interesting points and important information based on user's preferences.*

The proposing GIS platform has to offer the following functionalities:

- Ability to extract knowledge from the scene. The features of 3D models have to be extracted such as location, shape, size, volume, texture, etc.).
- Support of multiple databases
- Automatic navigation into the scene based on user's preferences and criteria

In order to design such a system, focused on a 3D large scale space, it is necessary to be aware of the virtual environment and the objects it contains. The altitude of the ground can be known. The system must be able to calculate the volume and the size of the obstacles (3D Models) in the environment. Their shape and their type could be important. In addition there are cases that properties of models or specifications have to be considered in order to plan a flight. Moreover, unique specifications will help the system to decide and better present an obstacle.

All these data have to be built or retrieved from an existing GIS system. Additionally, information and data that might be used will be stored in a local collaborative database. When the system has all of the above data and knowledge, it will be able to choose the points or objects to present and potentially the ones to avoid. These choices will take place based on the user's criteria, preferences and subsequently, the path will be planed according to them.

In the final processing stage of the system, the solution will be presented using a three dimensional viewer or browser. It has to give the opportunity to the user to interact with the environment and maybe change the parameters of the plan during the flight on the virtual world. The system has to be able reconsider the path based on the user's new preferences or criteria.

4.3 System Architecture

The above functionality is supported by the proposed architecture presented herein. A prototype application has been developed, called "GIS-Scene-Explorer", implementing this architecture. Among the available GIS systems, Google Earth was selected due to the advantage of availability of many supported tools like Google Building Maker and Sketchup. The platform has been developed upon the Windows operating system using the Google Earth component within a web browser. AJAX and XML technology was used to communicate between the client and the public database. The communication was accomplished using Google Earth API. The interaction and the control of the data were limited by Google

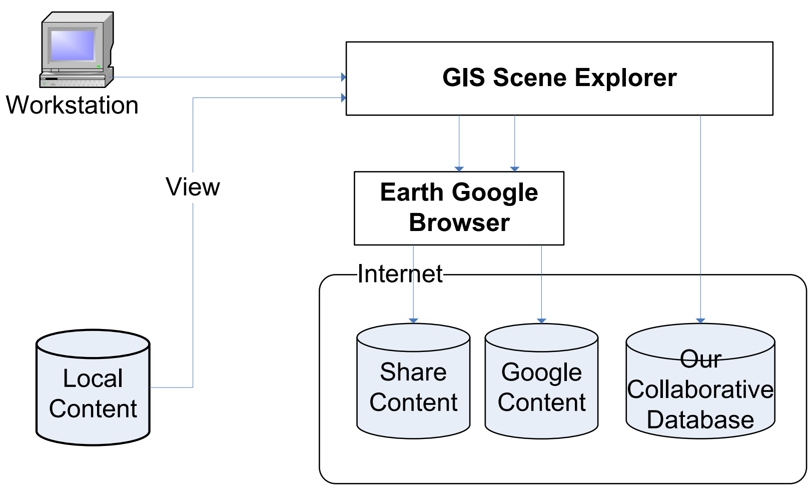

Fig. 8 Our GIS platform architecture

Earth API. Google Earth does not allow retrieval of the 3D models' information. The public GIS database was used only for retrieving data information for the morphology of the ground and for visualization purposes. Moreover, information about 3D models that exist on the public database is not useful. A collaborative GIS database was built for this purpose and all the 3D models of the scene were stored.

Google Earth component was used inside the GIS-Scene-Explorer only for visualization. The access and retrieval of the data was controlled by the GIS-Scene-Explorer. The morphology of the ground was retrieved by the public GIS database and the 3D models of buildings from the local, custom collaborative database (figure 8).

Multiple databases were used. With the aim of achieving our first requirement of extracting knowledge of the scene, all the required information and data of the whole scene must be stored on our own collaborative database. All the 3D models of the buildings were stored on it. Each model was containing information about its shape, size and location. Moreover the mapped images of texture of each individual side of the model were also contained on the database.

Google's public database was used for retrieving information about the morphology and texture of the ground (terrain). The morphology of the ground was not necessary to be known for the position of the buildings on the environment.

The GIS platform was able to use only the mined data for our own collaborative database. The rest data either form Google of from third party databases was used only for visualizing purposes on user's side.

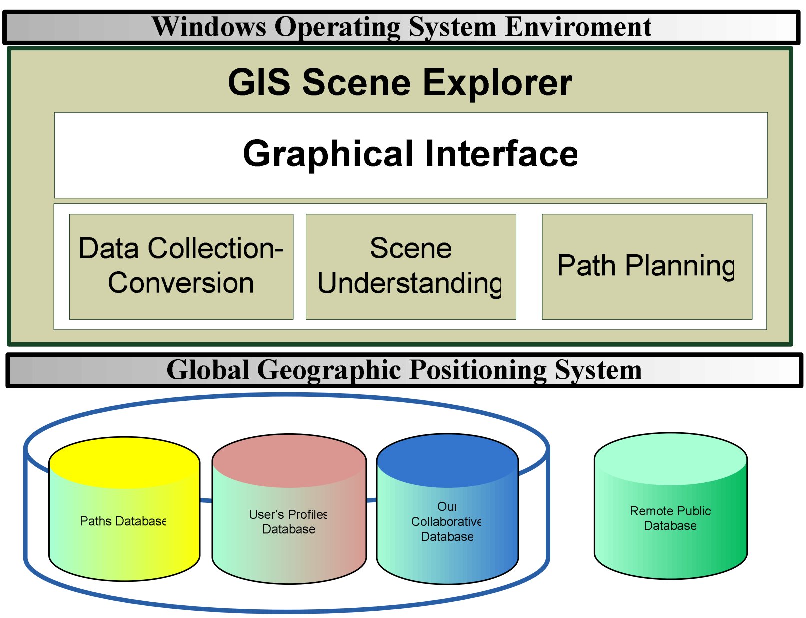

Fig. 9 Block Diagram

Regarding the knowledge of the models of the entire scene, it is possible to understand the scene and, afterwards, to plan a path, exploring the scene based on user's preferences or search criteria. This path is automatically calculated by the prototype application and visualized inside the GIS-Scene-Explorer using the Google Earth component (GIS viewer/browser) (figure 9).

GIS Scene Explorer (GIS-SE) has three modules. The first part of our approach was to design a collaborative database. The first module that it has already been completed is to gather data of 3D models. For this purpose a Collaborative web based platform was designed. 3D buildings were reconstructed using Google Building Maker. If it was required the models was improved by Google Sketchup. Apart of those models, data are retrieved from free online public databases. Likewise it was stored in our own collaborative database, in order fulfill the scene.

Afterwards the 3D models were mined from the collaborative database and they were analyzed. This module is still under development. Features of 3D models are extracted, such as location, size, shape and the mapped images of the texture of it. Considering these data, the whole scene is known. Only data from the collaborative database can lead to scene knowledge and understating. Data that are retrieved from other databases can't be used on this step of the process.

The aim of our platform is the automatic navigation and exploration of the whole 3D scene, based on user's search criteria, user's profile and preferences.

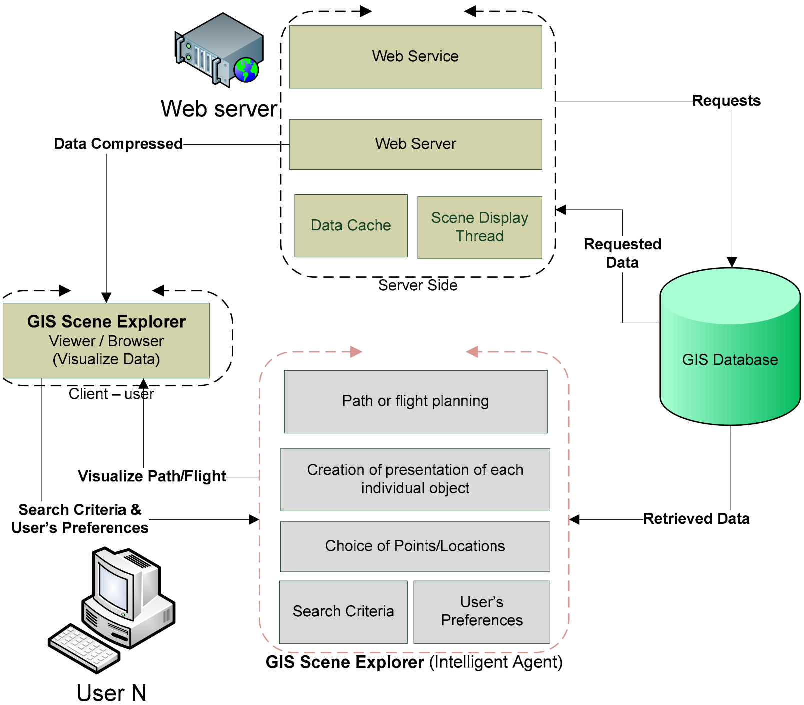

Fig. 10a Detailed Block Diagram of GIS-Scene-Explorer

The last module of our platform has to be built. It is responsible for the automatic navigation into the 3D scene.

A more detailed functional diagram of our platform is displayed below (figure 10). Each module is presented in details, analyzed in components.

Automatic navigation can take place into three steps. Taking in consideration the knowledge of the scene, the user's criteria, profile and preferences some points or buildings will be included or avoided from the calculated path. Secondly the presentation of each building model will be calculated using a best viewing algorithm. Finally the trajectory of the camera will be calculated in a way to include all the desired buildings and points of interests.

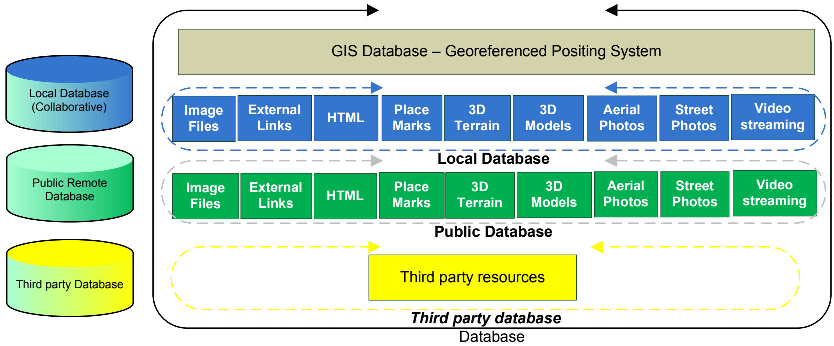

Fig. 10b Detailed Block Diagram of GIS-Scene-Explorer

3D models are mined from the collaborative one and terrain of the ground from Google Earth's one. Support for multiple third party databases have also taken into consideration. In addition it was used for the visualization of 3D environment of the scene using OpenGL-DirectX (figure 10). Apart of georeferenced data, user's profile, preferences and search criteria can stored on our database. The last is considered for the selection of the points of interests or buildings in the module of automatic navigation.

In a similar way, points or models can be avoided. Each model is presented individually based on its own specifications and shape. The best view is calculated for each model and finally the trajectory of the camera is created. The morphology of the ground does not affect the knowledge of the scene, as relative to the ground altitude can be can used The final trajectory – path of the camera contains only the interesting models, ensuring best viewing for each one of them. The path is transformed into the Google Earth's API language and is subsequently visualized by Google Earth component.

4.4 GIS Database

The existing GIS systems have two parts, the server side and the client side. In the client side there is an application that can visualize the data from the GIS database. These applications are only viewers or browsers without any ability to interact with the virtual world. There is no ability to select an obstacle or measure it. The specifications of any object are not available. Alternatively, if all the models are stored on a local database, they can be accessible and easily measured.

In order to design a system that will be able to navigate, search and analyze 3D models, an alternative, fully accessible GIS database has to be used. For this purpose, a collaborative database was designed. The remote access of the database and the distribution of data were achieved through the http web service. A web site was built and the database was hosted in a SQL server. A Google earth plug-in component was used to visualize the data and create a 3D dimensional world with OpenGL-DirectX. Google has distributed in public an API for embedded Google Earth use. Using AJAX technology and the Google Earth API, the public database was filtered and the distributed database was loaded and visualized on client's application upon request.

The collaborative database is able to store and enhance the knowledge and the data of the GIS database. All three dimensional models and morphology of the ground is contained in the collaborative database. Afterwards, in the next module of path planning process, only the collaborative database is considered, ignoring the public one. All required information has to be on the database, i.e. the whole scene.

A team of student volunteers participated in the prototype evaluation. They constructed three dimensional models of buildings in the region of Athens, using Google Earth's tools. The three dimensional models had texture obtained from real satellite and aerial photographs. The models were eventually stored in the collaborative database.

4.5 Navigation to 3D Scene

Using the GIS-Scene-Explorer application, the user can navigate and freely "fly" among the 3D models of the scene (figure 4.5.1). Data are filtered from the collaborative or public GIS database and then they are visualized. The navigation process can easily be controlled using the interface of the Google Earth Plug-in window. The data exchange and communication between the client and the database are based on AJAX and XLM technology. In the next figure the GIS-Scene-Explorer is presented during the navigation of a scene with 3D models of buildings.

4.6 GIS Scene Explorer (GIS-SE)

A prototype "GIS Scene Explorer" application has been designed and implemented in order to fulfill our platform's objectives. The platform has taken advantage of an existing GIS platform. Google Earth plug in was used for visualizing purposes, providing a three dimensional georeferenced space. It was used inside our standalone application (figure 11).

Fig. 11 Using GIS-Scene-Explorer for free navigation into georeferenced scenes containing 3D building models with texture

5 Conclusions

On this paper a prototype GIS platform was presented. Our approach was able to support multiple GIS databases. Data retrieved by our own collaborative databases were contained features of the 3D models. Knowledge of the scene was able to be extracted. The public GIS database of Google Earth was used for retrieving the morphology of the ground and a local collaborative one for the entire GIS scene.

The scene contains 3D building models with texture originating from satellite or aerial photographs. The user can fly and navigate freely in this three dimensional environment just like in Google Earth or any other GIS 3D application.

The purpose of GIS Scene Explorer is to provide a platform where all the content of the 3D environment is available. The public GIS systems do not allow gathering information about their content. They just use a visualizing module and allow the user to navigate into it freely. In our approach the whole scene is stored on a collaborative database. Each 3D model of the scene is accessible and editable by the platform itself. Moreover, all the benefits and advantages of the public GIS systems are still maintained. The user can still navigate freely into the 3D GIS scene using the mouse and the Google Earth interface.

The evolution of the proposed and implemented environment will allow the user to search into the scene based on criteria or preferences. The GIS Scene Explorer will automatically navigate the user into the scene, presenting only desired information, presented from optimal viewpoints, also taking into account the user's preferences.

6 Future Work

6.1 Semantic Knowledge

In this paper the architecture of a GIS platform. A prototype "GIS Scene Explorer" application was presented supports our approach. The content of the data was including only 3D models written in COLLADA. The database may also contain photographs, placemarks and any hypertext content that is supported by Geography Markup Language (GML), which has been adapted by the Google Earth. After the construction of a scene, a question arises: How we could navigate or fly inside the scene, watching all the interesting points while avoiding the rest of the data? The management of a large scale three dimensional scenes is one of the major problems of large city models. In addition to the information conveyed by the 3D models themselves, if semantic knowledge is added to the scene, the user could be guided by the system to see only the desired information of the scene.

6.2 Path or Flight Planning

This part of the process can be accomplished by an intelligent agent obtaining all required data for the process from the local database. Before the path planning takes place, the user's profile containing his/her preferences must be known. The user will be able to describe, to choose or set criteria for the navigation. The criteria will suggest points of interest or points to avoid.

In the first part of the process each point of the path and the sequence of them will be calculated. Afterwards, for each point of the path, the optimal way of presenting the object will be calculated, based on its specifications and its type.

6.3 Scene – Path or Flight Viewer

This is the last part of the process. In this part the path calculated in the previous step is visualized. The role of the viewer module can be accomplished by the client of the GIS system. The user may either watch the path/flight without intervention or he/she may alter the initial plan. In this case the system will have to recalculate the path. Dynamic user intervention in the visualized trajectory increases the complexity of the task.

6.4 Possible Applications

Applications of city 3D modeling systems can be found in a variety of fields. Virtual tours can be achieved in places that one can or cannot visit. In the field of tourism, for example, a 3D city helps to develop and present a region to the rest of the world. Furthermore, simulations can be based on 3D city models, with respect to physical phenomena, evacuating processes, etc.

References

[1] Alcorer, P.: On the selection of good views and its application to computer graphics (2003)
[2] Bonnefoi, P.F., Plemenos, D.: Constraint satisfaction techniques for declarative scene modeling by hierarchical decomposition. In: 31A 2000 International Conference (2002)
[3] Borrmann, A., Rank, E.: Specification and implementation of directional operators in a 3D spatial query language for building information models (2008)
[4] Borrmann, A., Rank, E.: Topological analysis of 3D building models using a spatial query language (2009)
[5] Bosch, A., Zisserman, A., Muoz, X.: Scene classification via plsa (2006)
[6] Chen, C.: Information visualization and virtual environments. Springer, London (1999)
[7] Cristani, M., Perina, A., Castellani, U., Murino, V.: Content Visualization and Management of Geolocated Image Databases (2008)
[8] Jones, T.M.: Google's Geospatial Organizing Principle (2007)
[9] Anestis, K., Fotis, A., Christodoulos, C.: On 3D reconstruction of the old city of Xanthi. A minimum budget approach to virtual touring based on photogrammetry. Journal of Cultural Heritage 8, 26–31 (2007)
[10] Larive, M., Gaildrat, V.: Wall Grammar for Building Generation. In: Graphite 2006, Kuala lumpur, Malaysia (September 2006)
[11] Lucas, M., Desmontils, E.: Declarative modelers. Revue Internationale de CGAo et d' Infographice, 559-585 (2007)
[12] Mackinlay, J., Card, S., Robertson, G.: Rapid Controlled Movement through a Virtual 3D Workspace (1990)
[13] Ruger, M., Preim, B., Ritter, A.: Zoom Navigation Exploring Large Information and Application Spaces (1996)
[14] Sokolov, D., Plemenos, D.: High level methods for scene exploration. Journal of Virtual Reality and Breadcasting 3 (2006)
[15] Sokolov, D., Plemenos, D.: Intelligent scene display and exploration (2006)
[16] Sokolov, D., Plemenos, D.: Viewpoint quality and scene understanding (2005)
[17] Sokolov, D., Plemenos, D.: Virtual world explorations by using topological and semantic knowledge, pp. 173–185. Springer, Heidelberg (2007)
[18] Sriphaisal, W., Pujari, A.K. (n.d.): A comparative Assessment of Giservices Architectures
[19] Desney, T., Robertson, G., Mary, G.: Exploring 3D Navigation: Combining Speed-coupled Flying with Orbiting (2001)
[20] Wikipedia (n.d.). Google Maps, `http://en.wikipedia.org/wiki/Google_maps`
[21] Wikipedia (n.d.). Virtual Globes, `http://en.wikipedia.org/wiki/Virtual_globes`
[22] Wolf, C.: Content based Image Retrieval using Interest Points and Texture Features, Technical Report, PRIP-TR-61 (2000)
[23] Zhang, J., Gong, J., Lin, H., Wang, G., Huang, J., Zhu, J., et al.: Design and development of Distributed Virtual Geographic Enviroment system based on web services (2007)