

# GIS Scene Explorer, an Intelligent Navigation Tool for Georeferenced Scenes

Christos P. Yiakoumettis<sup>1,2</sup>, Georgios Bardis<sup>2</sup>, Georgios Miaoulis<sup>2</sup>, Dimitri Plemenos<sup>2</sup>, Djamchid Ghazanfarpour<sup>1</sup>

<sup>1</sup>Université de Limoges,  
Laboratoire XLIM, CNRS, UMR 6172  
83, rue d'Isle, 87000 Limoges France  
Email: [{djamchid.ghazanfarpour@unilim.fr}](mailto:djamchid.ghazanfarpour@unilim.fr)

<sup>2</sup>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  
Email: [{chgiak@teiath.gr; gbardis@teiath.gr; gmiaoul@teiath.gr;}](mailto:chgiak@teiath.gr)

**Abstract:** *The rapid evolution of GIS systems during the last decade has lead to improved accessibility and increased availability. Their traditional content has been enhanced by multimedia information and three dimensional models. Virtual copies of the real world have been created, widely available to the average user, yet the tasks of efficient exploration of 3D large scale scenes and information retrieval from them still pose serious problems. In this paper, a prototype application, entitled “GIS-Scene-Explorer” or GIS-SE, addressing these problems is presented. The prototype is a standalone application, adopting the Google Earth API to retrieve and visualize data. It takes advantage of both the public 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:** virtual globe, GIS systems, scene exploration, virtual tour, scene understanding, visualisation, public participation, city modelling, urban planning

## 1. INTRODUCTION

During the last six years many commercial GIS applications with real geo-referenced scenes have been released. These applications, known as *virtual globes*, offer a three dimensional model of the real world's surface, built upon satellite and aerial photographs (Zhang, et al., 2007). All their data are stored in dedicated databases and they are organized by a global positing system. The data include hypertext, photographs and three dimensional models, calibrated on a referenced pointing system (the global positioning system is usually adopted). Commercial applications have been released by Google (Google Earth), NASA (NASA World Wind), Microsoft (Virtual Earth 3D) and more, comprising more than simple mapping applications since they represent a 3D morphological model of the surface of the earth. They include information about the streets, points of interest, 3D models of buildings and multimedia content (Jones, 2007). Each one is using its own environment for visualizing the data in the form of a browser.

These environments offer the user the ability to navigate and freely “fly” within the scene, either in 2D or 3D mode. However, the navigation on the scene can only be manually accomplished. This fact, in combination with the scene complexity and the user's incomplete knowledge of it, make the task of manual navigation and the insight gained by it generally inadequate.

The paper presents an architecture of a GIS system for automated scene exploration. All required data of the 3D scene are stored in a collaborative local database. The basic architecture presented may also serve as the platform for future features, for example construction of personalized paths of exploration into scenes, considering the user's preferences and requests.

## 2. VIRTUAL GLOBES

### 2.1. A simple two dimensional GIS system architecture

Google Earth, released in 2004, was the first virtual globe application that became widely popular. It comprises a three dimensional virtual environment, that allows the user to navigate and explore it. It is based on a server-client GIS system architecture where 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 are 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. The first applications used for this purpose were limited in two dimensional environments (Wikipedia, Google Maps), (Wikipedia, Virtual Globes). Google Earth started with 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 2.1.1.

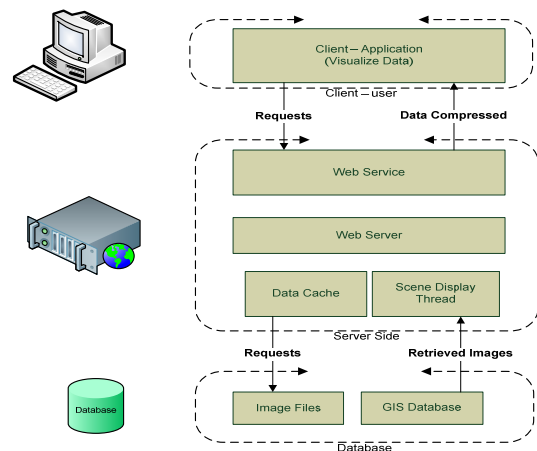


figure 2.1.1 – Block Diagram of two dimensional GIS System

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. A web server was used for the communication of the client's visualizing application and the database. The user was requesting data and the retrieved information was visualized on its side (figures 2.1.1 – 2.2.1).

## 2.2. The evolution of the above GIS system

The evolution of the GIS systems has led to the improvement of the services and the amount of available information. In the beginning, additional information was added to the database based on two dimensional aerial photographs. The names of the streets and the roads were among the first data that were added to the database. Afterwards, metadata such as text, external links, hypertext, and landmarks enriched the database.

Despite this evolution, the fundamental architecture has remained practically the same (figure 2.2.1). All the data were imported to databases and calibrated over the same geographical positioning system.

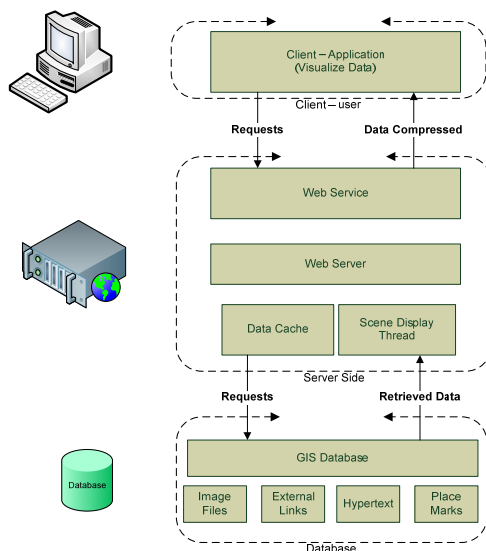


figure 2.2.1 – Block Diagram of improved GIS system

In this case, however, 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.

Today, there are many available databases that can provide information and data for a GIS system. The popularity of such systems and the advancements in computer technology have

made possible the creation of three dimensional 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. 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.

The end-user can choose which information will be retrieved from the GIS system. Each database is different than the others. 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. RELATED WORK

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 it 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 extended and topological relationships were added such as *within*, *contain*, *touch*, *overlap*, *disjoint* and *equal* (Borrmann & Rank, 2009).

Finally, there are approaches of 3D large scale scene models. The 3D model can either be generated by an engine or reconstructed from a real building. According to the first approach, a declarative engine for generating a 3D large scale scene was introduced by Larive & Gaidrat (Larive & Gaidrat, 2006).



figure 3.1.1: 3D Large Scale Scene of 17.362 buildings (Larive & Gaidrat, 2006)

### 3.2. Scene understanding and exploration

During the last years, several approaches have been presented 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 it 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).



figure 3.2.1: Initial urban scene

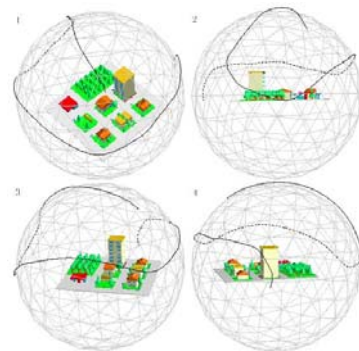


figure 3.2.2: Camera trajectory of an urban scene (Sokolov & Plemenos)

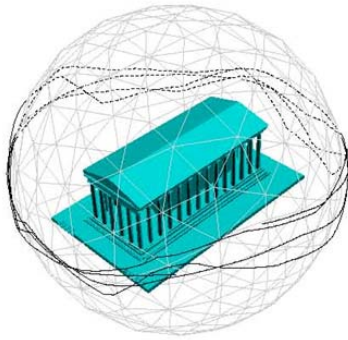


figure 3.2.3: Camera trajectory of temple  
(Sokolov & Plemenos)

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 are used to define the position of the camera on an orthogonal 3D coordinate system and the other two are used for the horizontal and vertical orientation of the angle of view. Each point of view is evaluated individually (Sokolov & Plemenos, 2007).

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 or a degree of interest can represent each point of visit, or a number that represents the importance of it, compared to the others (Bonnetfoi & 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 AN INTELLIGENT SYSTEM FOR PATH PLANNING

### 4.1. Functionality of the proposed intelligent system

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

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) on 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 and preferences and, subsequently, the path will be planned 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.

## 4.2. 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 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.

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 4.7.1).

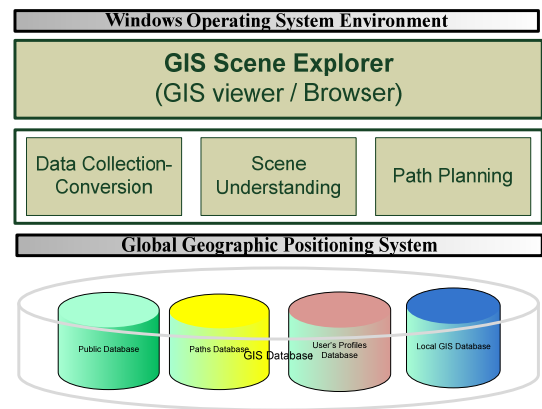


figure 4.7.1 – Block Diagram

A more detailed functional diagram is displayed below. Each module is presented in detailed, analyzed in components.

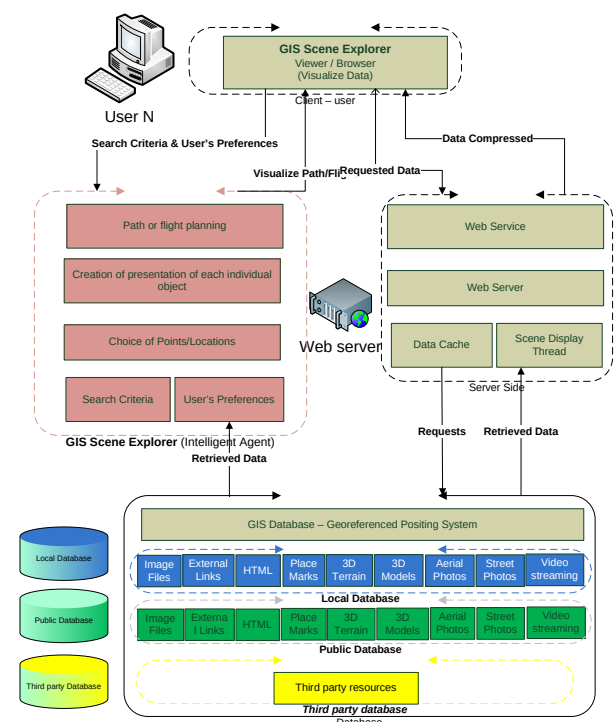


figure 4.7.2 – Detailed Block Diagram of GIS-Scene-Explorer

At the user’s side there is a standalone application with the appropriate interface for gathering user’s preferences and search criteria. Data requests are sending to the collaborative database through the user’s applications. This application is using a web browser for controlling the Google Earth component with JavaScript. This component was used for retrieving terrain morphology. In addition it was used for the visualization of 3D

environment of the scene using OpenGL-DirectX (figure 4.7.2).

The user is able to input search criteria or preferences. The whole scene is retrieved by the local collaborative database. All the required knowledge for the path planning is also stored in the collaborative database. Considering the morphology and the knowledge of the scene, the desired models will be selected in order to be included into the path. 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 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.

#### **4.3. 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

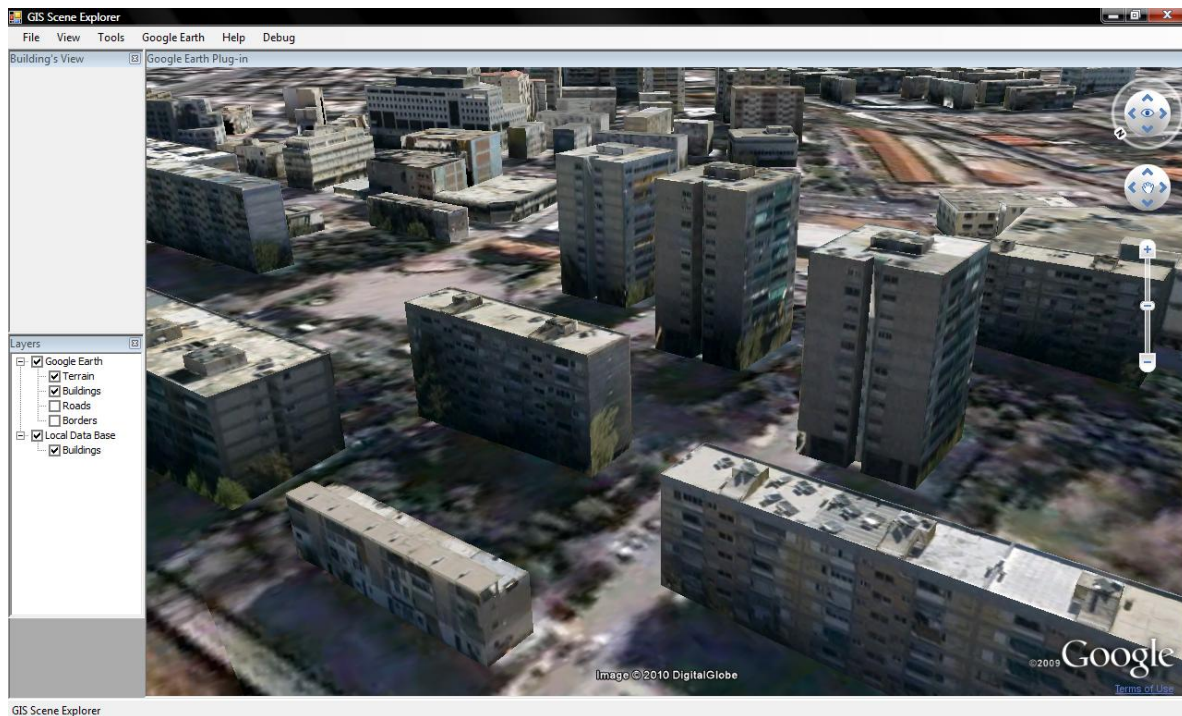
SQL server. A Google earth plug-in component was used to visualize the data and create a 3D dimensional world with OpenGL-DirectX. Google has distributed in public an API for embedded Google Earth use. Using AJAX technology and the Google Earth API, the public database was filtered and the distributed 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.

#### **4.4. 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.



*figure 4.5.1 – Using GIS-Scene-Explorer for free navigation into georeferenced scenes containing 3D building models with texture*

## 5. CONCLUSIONS

On this paper a prototype GIS application was presented. Our approach was able to combine multiple GIS databases, using the public GIS database of Google Earth 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 Scene Explorer system was presented. The content of the data was including only 3D models. The database may also contain photographs, landmarks 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 has not 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.

## **7. REFERENCES**

- [1]. Alcorer, P. (2003). On the selection of good views and its application to computer graphics.
- [2]. Bonnefoi, P. F., & Plemenos, D. (2002). Constraint satisfaction techniques for declarative scene modeling by hierarchical decomposition. 31A'2000 International Conference.
- [3]. Borrmann, A., & Rank, E. (2008). Specification and implementation of directional operators in a 3D spatial query language for building information models.
- [4]. Borrmann, A., & Rank, E. (2009). Topological analysis of 3D building models using a spatial query language.
- [5]. Bosch, A., Zisserman, A., & Muoz, X. (2006). Scene classification via pls.
- [6]. Chen, C. (1999). Information visualization and virtual environments. London: Springer-Verlag.
- [7]. Cristani, M., Perina, A., Castellani, U., & Murino, V. (2008). Content Visualization and Management of Geo-located Image Databases.
- [8]. Jones, T. M. (2007). Google's Geospatial Organizing Principle.
- [9]. Larive M., Gaildrat V., 2006 «Wall Grammar for Building Generation», Graphite'06, Kuala Lumpur, Malaysia, September, 2006
- [10]. Lucas, M., & Desmontils, E. (2007). Declarative modelers. *Revue Internationale de CGAo et d'Infographie*, 559-585.
- [11]. Mackinlay, J., Card, S., & Robertson, G. (1990). Rapid Controlled Movement through a Virtual 3D Workspace.
- [12]. Ruger, M., Preim, B., & Ritter, A. (1996). Zoom Navigation Exploring Large Information and Application Spaces.
- [13]. Sokolov, D., & Plemenos, D. (2006). High level methods for scene exploration. *Journal of Virtual Reality and Broadcasting*, Volume 3.
- [14]. Sokolov, D., & Plemenos, D. (2006). Intelligent scene display and exploration.
- [15]. Sokolov, D., & Plemenos, D. (2005). Viewpoint quality and scene understanding.
- [16]. Sokolov, D., & Plemenos, D. (2007). Virtual world explorations by using topological and semantic knowledge. *Springer*, 173-185.
- [17]. Sriphaisal, W., & Pujari, A. K. (n.d.). A comparative Assessment of Giservices Architectures.
- [18]. Tab Desney, Robertson, George, & Gzewinski Mary. (2001). Exploring 3D Navigation: Combining Speed-coupled Flying with Orbiting.
- [19]. Wikipedia. (n.d.). Google Maps. [http://en.wikipedia.org/wiki/Google\\_maps](http://en.wikipedia.org/wiki/Google_maps).
- [20]. Wikipedia. (n.d.). Virtual Globes. [http://en.wikipedia.org/wiki/Virtual\\_globes](http://en.wikipedia.org/wiki/Virtual_globes).
- [21]. Wolf, C. (2000). Content based Image Retrieval using Interest Points and Texture Features, Technical Report, PRIP-TR-61.

- [22]. Zhang, J., Gong, J., Lin, H., Wang, G., Huang, J., Zhu, J., et al. (2007). Design and development of Distributed Virtual Geographic Enviroment system based on web services.