AUTOMATIC 3D RENDERING OF HIGH RESOLUTION SPACE IMAGES

C. Destruel [a,], C. Valorge [b]

[a] CS-SI, Rue de Brindejonc des Moulinais BP5872 - 31506 Toulouse Cedex 5 - France
[b] CNES, DSO/OT/QTIS/HR, 18, av. Belin – 31401 Toulouse Cedex 4 – France

Commission V, WG V/6

KEY WORDS: High resolution, 3D models, visualization, virtual reality, Pléiades

ABSTRACT:

Virtual flights over a Digital Surface Model (DSM) textured with an image are now widely used for communication purposes towards the general public, decision-makers and non-specialists in general. Furthermore, recent advances in computer science and virtual reality make very easy to perform such flights, even on a standard laptop. Examples will be given with SPOT5 2.5m colour images applied to HRS-derived DSM on a 60 km by 60 km surface.

Performing the same kind of virtual flight over metric images needs to solve a certain number of problems related to the fact that we have to move from a 2.5D description of the landscape to a real 3D description, essential for the recognition of all the objects that can be seen in such images. Therefore, the surface can no longer be described as a regular grid of altitudes, especially to be able to account for vertical walls. It is necessary to use a vectorized description of the landscape geometry, for example an irregular network of triangles (TIN) in a Virtual Reality Mark-up Language (VRML) format.

The problem of estimating a 3D model from images will not be addressed here. For our purposes, we will consider that this 3D geometric description is available from other sources (city models…) and that we "only" want to be able to virtually generate the landscape by applying a metric image over a given 3D model.

In this case, the first step is to check its quality and eventually simplify the 3D model -its quality and level of detail can vary considerably from one source to another, even for a same town- and then to merge it with a traditional Digital Terrain Model, giving the altitude of the ground and of the object's bases.

The second step consists in registering the image with the geometric model with a sub-pixel accuracy.

The third step deals with the texturing of the geometric model with the image, which means associating each triangle with the image chip representing its radiometry. This extraction process must take into account the viewing angle of the surface in the image and its orthogonal projection.

At this level, we have produced a virtual mock-up modelling a real 3D landscape that we can use to generate several outputs: an image in any projection (an ortho-image, for instance), any perspective view of the landscape or a virtual flight.

We will illustrate the different stages of the processing with Pléiades-HR simulated images texturing the 3D model of Toulouse.

1. INTRODUCTION

Nowadays, advances in Earth Observation allow the layout of big databases using high-resolution images (Valorge, 2003). Images taken from Space are now commonly used in presentation or demonstration applications, no longer reserved to professionals but also intended to neophytes. Among these applications, one can note the significant place of software presenting these data by using Virtual Reality metaphors.

In Virtual Reality, the use of teledetection data implies the knowledge of the landscape (and not only of the image) on which the user wishes to fly. Even if navigations using a Digital Elevation Model (DEM) as geometric base are now widespread, the increase in image resolution asks for using a true 3D geometric data set instead of a classic 2,5D DEM.

2. 2,5D NAVIGATION

Now it is quite common to see and handle 3D virtual navigations on a DEM textured with satellite images (Appleton, 2001), even if this remains a difficult exercise to ensure a sufficient frame rate (Zhao, 2001).

Our software, VirtualGeo, makes it possible to fly over any SPOT5 scene in real time and without any constraint.

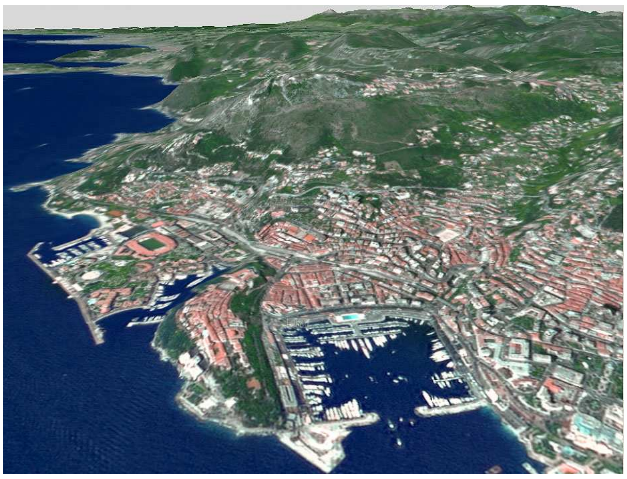

Figure 1. Virtual navigation over SPOT5 data

A classic navigation scenario (Vadon, 2003) is composed of a 60km x 60km area made up of 20 or 50 meter resolution DEM and a color image of 2,5 meter resolution (see Figure 1).

One can easily realize that taking into account the geometrical models made up of DEM is not enough to represent the 3D objects that the user perceives on a high resolution image. The first produced navigation using a 4 meters DEM clearly shows the limits of an approach considering only 2,5D of the space (see Figure 2).

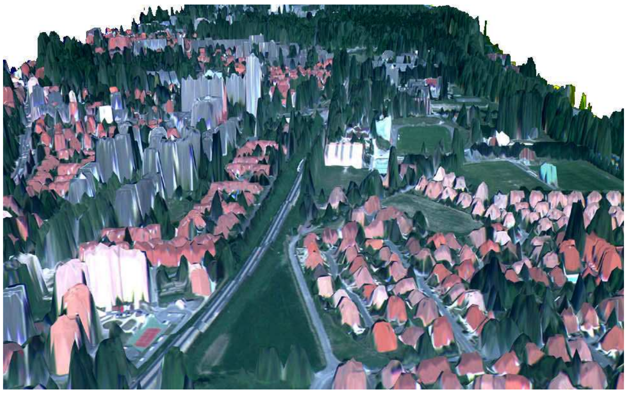

Figure 2. 4 meter DEM instead of real 3D objects

The natural solution to this problem consists in considering all the objects in a scene as really 3D objects.

3. NAVIGATION IN REAL 3D

3.1 Objectives

The main goal of this work is the visualization of a geometry textured with high-resolution images (one meter or better) in such a way that the integration of these two complementary elements (i.e. images and geometry) improves the perception during a virtual overflight of the different objects present in the environment.

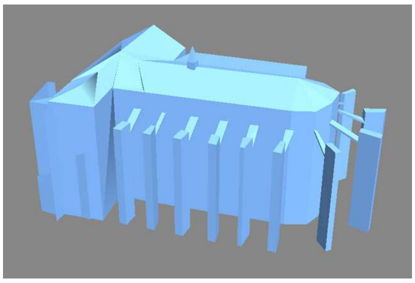

Figure 3. Real 3D object

Taking into consideration the 3 dimensions of objects (see Figure 3) on top of a traditional 2,5D description of the ground, makes it possible to obtain a fully virtual model of a given zone: the walls constituting the buildings can be perfectly vertical and faithful to reality, the possible concavity of the buildings is also considered and visualized (Beck, 2003).

Furthermore, if the virtual mock-up is precise enough, it could be used like a representative model and therefore, it will be possible to calculate any desired sight, from any position and any virtual model.

3.2 Source of geometrical data

Different sources can be used to retrieve useful 3D information:

- photogrammetric methods (from images, laserscanning….) (Ulm, 2003),
- 3D surveying.

Several papers have presented research in this domain which is not the topic of this paper: our point of view is a consumer one and we are mainly interested in the characteristics and quality of the 3D models retrieved from these different methods in terms of capacity to produce a virtual mock-up and allow a good quality virtual overflight.

Note: *even if all the processes we describe here can be applied to, we do not take into consideration techniques based on graphics designer work. Actually, a great part of algorithms implemented in this work are used to build missing information or to correct data errors. We can suppose that a graphics designer will produce an error-free model exactly matching visualization requirements.*

3.3 Existing data utilization

3.3.1 Goal

Our goal is to generate a 3D overflight of a scene textured with any high-resolution image and described by an existing 3D model.

In most cases, this 3D model is totally independent from the image (not retrieved from stereoscopy using the image to drape): typically, local authorities often own the 3D model of their city in order to manage its growth or to manage certain risks. These models are often provided by local surveyors.

We wish to preserve the possibility of wide navigation areas as we have made in navigations on SPOT5 data. To ensure the continuity of geometry data during the virtual overflight, we wish to merge the 3D data with a traditional 2,5D DEM.

Moreover, we aim at preparing this overflight in a fully automatic way.

3.3.2 Difficulties found

The heterogeneity of the data that we have previously described leads to a non-uniformity problem. Indeed, according to the acquirement method employed during the constitution of the database, we can find data resulting from:

- Stereoscopic process
- Ground measurements operated by surveyors
- Models drawn by architects
- Etc.

This diversity of methods introduces significant differences into the data structures:

- Absence/presence of the ground
- Continuity/discontinuity of the facetization
- Concavity/convexity of the buildings
- Used facets number for the produced model
- Etc.

All these differences should not interfere with the process to design.

The Figure 4 shows the detected objects: a building (in grey) and two independent structures (in red and green). This example shows how a discontinuous acquisition can result in perceiving a single building as the union of distinct buildings and thus perturb the building identification. Moreover, one of the procedures to be carried out deals with the baselines detection (see §3.3.3.3). This identification problem will also affect this process so that the three different objects will be connected to the ground and the final object baseline will erroneously include both green and red object baselines.

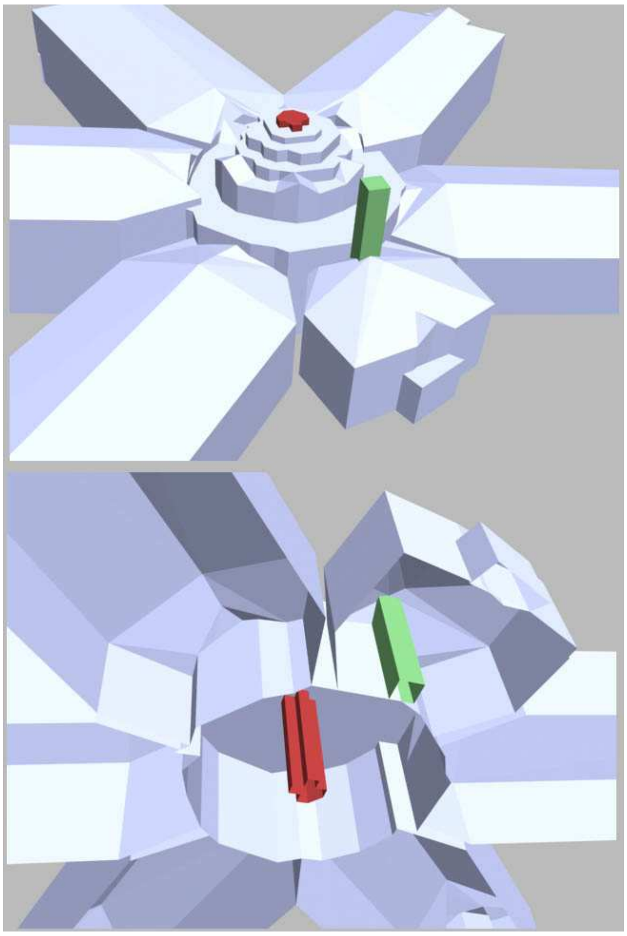

Figure 4. Composed building

3.3.3 The successive steps of the process

3.3.3.1 Fusion of close points

In a great number of cases, the 3D data contain "close" points (in fact, the same point with slightly different coordinates), which must be merged. This type of error can be really a trouble for building construction: two facets constituting a wall do not have a common edge (as it should be) but produce two superimposed edges. The space between both edges can be visible and disturbs the user, see Figure 5.

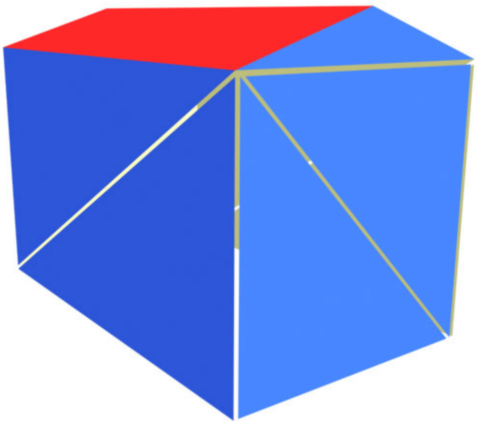

Figure 5. "close" vertices instead of "merged"

To solve this problem, we set up a fusion method considering point neighborhoods. Two points meeting the following criteria are automatically merged:

- Both points are "close",
- There is no edge between them.

The difficulty consists in the definition of the concept of proximity between the points. If we had chosen a fixed distance criterion, applied with a cube or a sphere centered on the considered point, the resulting geometry modification wouldn't be known a priori. The criterion we chose is intended to preserve the initial geometry by eliminating the duplicated points. To avoid geometry modification, we chose an adaptive criterion. We consider a sphere of influence whose center is the given point. The problem is now to calculate the ray of this sphere of influence:

- For each vertex of the model we calculate the length of all edges using it,
- The ray is calculated using the minimal length edge by applying a percentage of reduction, for example 5% of its shortest edge.

Considering two points, "A" and "B", if "A" belongs to the sphere of "B" and "B" belongs to the sphere of "A", then both points are merged.

Obviously the triangles defined by a merged point are updated. After this phase, all the degenerated triangles, it means non-made up exactly of 3 different points are deleted.

3.3.3.2 Division into objects

The available 3D data is not very often spatialized: the facets describing the objects are not organized in space. If we consider data source coming from CAD software, we can have data organized in a semantic way: facets are grouped by layers where the first one describes all the walls of the scene, another one all the roofs, etc.

In order to optimize all the necessary processing, it is important to increase high-level information with special specific tiling. For this reason we organize the available scene in facet groups, which will constitute our basic working objects for all next processes. This group definition only considers the connexity of the final objects: for all facets constituting an object, there is at least another facet of the object sharing with it a vertex or an edge.

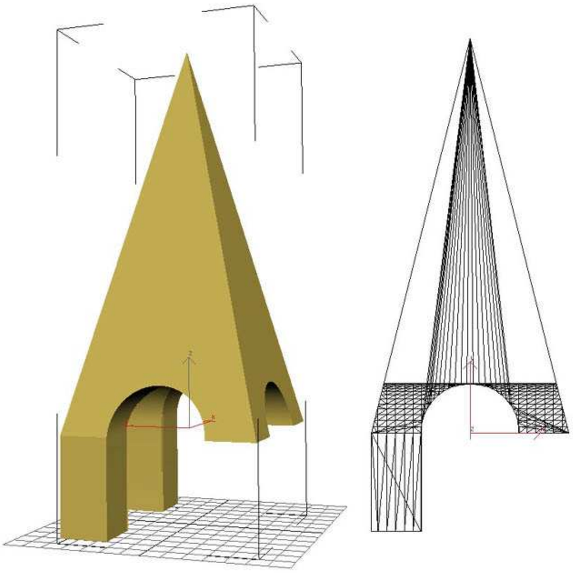

Figure 6. Object example

After this first stage, the identified objects exactly and strictly fulfill the geometrical criterion of continuity. It appears significant to increase the given definition of an object with a semantic dimension. The Figure 4 presents an object where the continuity criterion is not enough to fulfill the semantic definition of a building: we have got three different objects instead of a single edifice! Therefore, we need to group all intersected objects into a single meta-object.

Finally, two initial objects having each one at least a facet intersecting the other are then merged to form a single object. The Figure 6 presents an example of produced object.

3.3.3.3 Baseline detection

The objects defined at the previous stage must be merged into a traditional DEM. This fusion step requires the detection of the part of each object that must be geometrically in contact with the DEM. We call baseline the part of the object that we should merge into the DEM in order to increase its accuracy.

The detection stage of the object baseline has to identify:

- The main set of vertices identifying all local minima according to the 'y' (vertical) axis,
- All edges connecting these vertices.

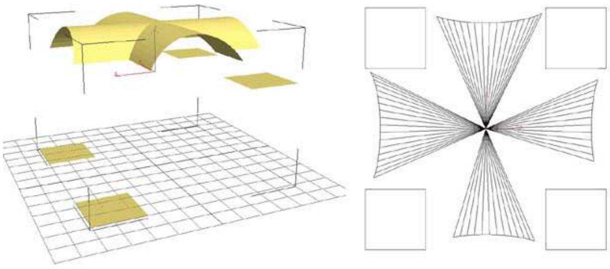

Figure 7. Detected baseline for example object

A vertex is a local minimum according to the 'y' axis if we cannot found any other point in any other facet for which its 'y' coordinates is lower than the considered vertex.

The whole set of edges connecting these vertices constitutes the baseline of the considered object. The Figure 7 represents the baseline extracted from the example object.

The constrained Delaunay algorithm that we use makes it possible to guarantee the respect of contact edges between the ground and the model (Mostafavi, 2003).

3.3.3.4 Arch suppression

The arches can be considered as specific interpretations of part of the detected baseline coming from the previous stage: the arch defines a concave object where the detected baseline, according to local minima's criteria, should neither be merged nor linked with the ground represented by the DEM.

The choice to take into account or not an arch probability can only be taken by a human operator. Actually, it is impossible to automatically distinguish an arch that should not be merged, from an inclined baseline that should be.

Therefore, if the detection mechanism is activated by the operator, it proceeds in three phases:

- Classification of the edges constituting the baseline,
- Detection of the intersections between classes,
- Elimination of the identified baseline classes.

<u>Classification</u>

This stage constitutes the first step for the arches identification in the object baseline.

Classification consists in gathering edges forming the baseline in order to define "altitude classes". An altitude class is made up of the baseline elements (i.e. edges set), each one covering an altitude range (minimum and maximum) intersected by at least the altitude range of another element.

Another way to define this intersection is to consider the projection on the 'y' axis of each baseline edge. These projections produce several continuous areas on the 'y' axis. Any discontinuity on this projected area defines a new altitude class.

<u>Intersections</u>

Once the classification of the baseline completed, we should consider it projected on the 'xz' (horizontal) plane. For each altitude class, we define on this plane a single polygon enclosing all edges of the object baseline forming the given class. This process produces several closed areas, each one covering distinct altitude ranges.

The intersections detection is then made between each polygon.

<u>Deletion of arch classes</u>

For all the edge classes identified at the previous step, we determine their level i.e. the lower points of the class considering a minimum 'y' criteria. We choose in the class set all intersected edge groups that do not have the local minimum level. Finally, we remove all edges of the identified classes from the list constituting the baseline.

This process is applied for all edges defining each baseline of every used object.

The Figure 8 shows the final baseline built for the given object example.

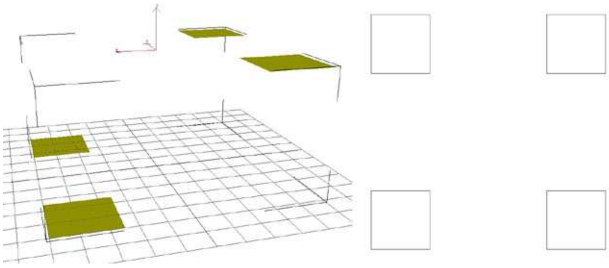

Figure 8. Final baseline for the object

3.3.3.5 Tiling process

Before being modified by the various objects of the 3D model, the DEM over which we wish to navigate is meshed. This allows us to handle it like a simple 3D object.

Once the various elements merged into the DEM, we have a single 3D model describing the whole scene, ground and surface.

In order to guarantee a freely virtual navigation, the improved 3D model is cut out in geometrical tiles (see Figure 9). These tiles are saved separately in order to be dynamically loaded during the virtual flight.

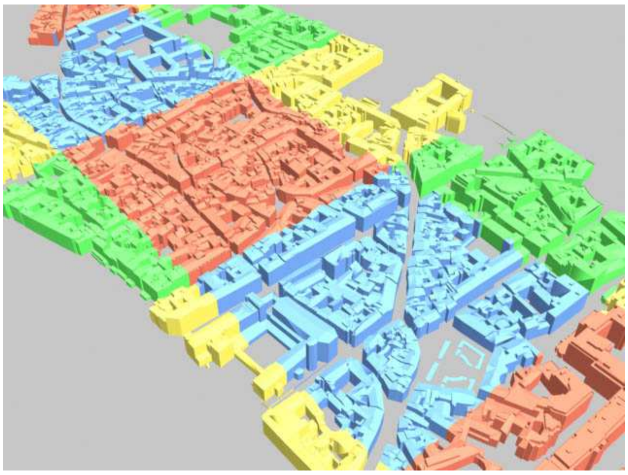

Figure 9. Example of tiled area

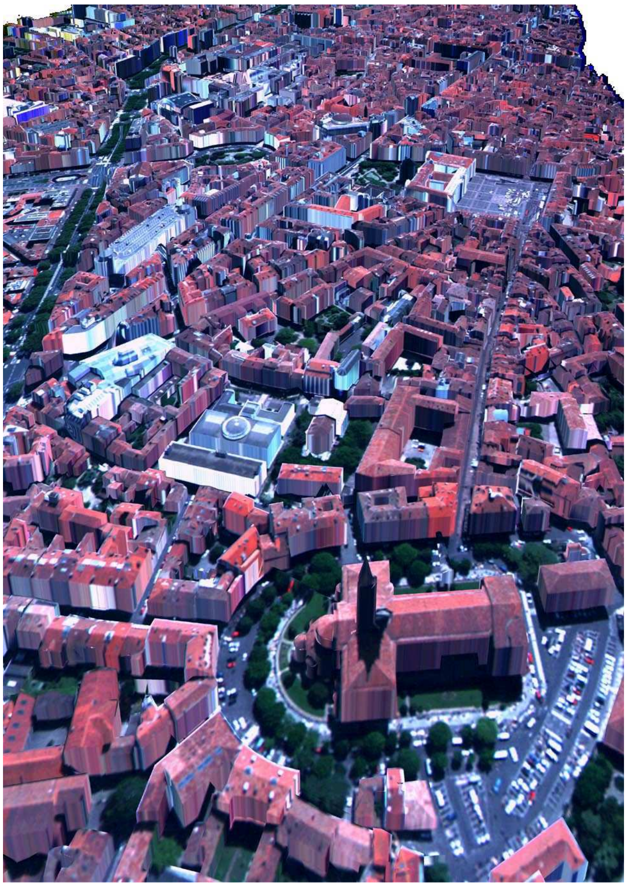

Figure 10. Flight over Toulouse downtown

4. PERSPECTIVES

4.1 Performances

The system described here is designed to enrich the 2,5D geometry of a traditional DEM with real 3D data.

However, we must take into account two significant considerations:

- We wish to increase the geometrical representatitvity (and thus the facet number) of the used 3D models,
- It is essential to guarantee a sufficient frame rate for any virtual overflight.

To achieve these goals, we currently work on a priori simplification methods to be applied to 3D objects.

4.2 Texturation of the 3D model with images

Currently, we texture the produced geometrical model with an orthoimage, which means that we vertically project this image on the 3D model: this leads to quasi-uniform walls, which is not realistic (see Figure 10). Improving this process means resolving two main problems:

- Data availability
- Exact extraction of building walls from images.

Data availability

Data availability is a requirement which is very difficult to comply to: it is essential to have a high number of images taken from different points of view to be able to actually see the walls, roofs and streets. If we do not have enough images, the only possible choice will be between facets with ill- or without radiometric information.

Texture extraction

Supposing that we have enough images to texture all the geometrical facets of the model, the next problem deals with the radiometric facets extraction process. Apart from the choice to be made between the various views of the same facet, it is necessary to take into account all used image projections.

The essential pre-requisite for the wall treatment is the exact knowledge of the geometric viewing model for each available image. We develop techniques allowing to precisely adjust these models with respect to the 3D model geometry: we take into consideration methods based on automatic matching. Once the viewing model is accurately adjusted, we use it to project the 3D model in the image's geometry in order to generate a virtual image of the database. This virtual image can be perfectly superimposed over the considered image while making sure of preserving the faceted structure of the initial model.

This couple, the projected model and the image, makes it possible to obtain all available textures in the given image. This process must be applied to all available images for the considered scene.

Once the texture facets are extracted, we will apply the inverse projection model to resample all texture facets. This process allows to come back to the geometry of the scene and to define classic texture coordinates of virtual reality conventions.

This process makes it possible to generate for each geometrical facet as many texture facets as available images. The last step will be to choose, for each geometrical facet, the correct texture between these various possibilities.

4.3 Real 3D

Finally, once the virtual mock up is achieved (meaning modeled and textured), we wish to use it not only for virtual overflights but also to generate perspective views. Indeed, once the 3D scene is available, it will be possible to define a virtual camera system that will be able to produce a virtual image. This image will be virtual considering the used camera but real considering the used data. Using a virtual system to generate these images allows us to simulate any virtual camera.

We can even consider that the problem of re-sampling any high resolution images affected by 3D effects can only be made this way in a correct way: this virtual mock-up is an essential step.

5. CONCLUSION

We are able to virtually navigate over SPOT5 scenes and now, we wish to treat in the same manner the future images of the future high-resolution satellites as Pléiades-HR. We saw that, in order to take into account these images (with sub-meter resolution), it is necessary to increase the resolution of the geometry on which the image is mapped. This increase in data resolution imposes to consider all the various objects constituting the scene (buildings, trees, etc).

We set up an automatic data processing sequence allowing to solve the many problems of consistency of available data and to carry out all the treatments needed to merge in an exact way the 3D data into a traditional DEM. These merged data are from now on available for any virtual overflight.

Finally, we wish to bring a solution to the problem of textures for the building walls. After a geometry simplification step, we wish to perfect an automatic process of radiometric facet extraction in the various available images.

6. REFERENCES

Appleton, K., Lovett, A., Sunnenberg, G., Dockerty, T., 2001, Visualising Rural Landscapes From GIS Databases: a comparison of approaches, possibilities and problems. GIS Research UK (GISRUK) 2001, Glamorgan, Wales, United Kingdom

Beck, M., 2003. RealTime Visualization of big 3D City Models. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXIV-5/W10

Mostafavi, M. A., Gold, C., Dakowicz, M., 2003. Delete and insert operations in Voronoi/Delaunay methods and applications. Computers & Geosciences vol. 29, Pergamon Press, Inc., pp523-530

Ulm, K., 2003. Improved 3D city modeling with cybercity-modeler (CC-ModelerTM) using aerial-, satellite imagery and laserscanner data. International Workshop on Visualization and Animation of reality based 3D models, Tarasp, Switzerland.

Vadon, H., 2003. 3D navigation over merged Panchromatic-Multispectral high resolution SPOT5 images. International Workshop on Visualization and Animation of reality based 3D models, Tarasp, Switzerland.

Valorge, C., 2003. 3D restitution and rendering through high resolution imagery: state of the art and new challenges. …

Zhao, Y., Zhou, J., Shi, J., PAN, Z., 2001. A fast algorithm for large scale terrain walkthrough. International Conference on CAD&Graphics 2001, Kunming, China