Layered Geometric Information System

Pierre Smars
Dept of Architecture & Civil Engineering, University of Bath
Bath BA2 7AY, United Kingdom
Phone: +44-1225-826654, E-mail: absps@bath.ac.uk
Koen Van Balen, Kris Nuyts
Raymond Lemaire Centre for Conservation, K.U.Leuven
Kasteelpark Arenberg 1, B-3001 Heverlee, Belgium
Phone: +32-16-321687, E-mail: koenraad.vanbalen@bwk.kuleuven.ac.be

Keywords: Cultural Heritage, Database, Management, Augmented Models, Reconstruction, Visualisation

Abstract

Virterf is a software combining recent developments in computer vision and reverse engineering targeted for professionals working in the field of architectural conservation. The first part of the software aims at building relatively automatically a 3D model. In this article, focus is given to the second part, offering tools to qualify and query it. Information can be added, organised in themes or linked to the model. The resulting database with the software's visualisation capacities facilitates comparison, synthesis and decision.
Some reference concepts are first presented, the functionalities offered described and - following first applications - some possible development are discussed.

1 Introduction

1.1 The Virterf project

From September 1999 to September 2001, a project called *Virterf* ran at the University of Leuven. It resulted from a collaboration between three departments of the University. The aim was to develop a "three-dimensional digital information system for the documentation, representation and conservation of our architectural heritage".
The department of electrical engineering (ESAT) had experience in 3D reconstruction of scenes and developed a module to acquire geometrical data (points) from a set of images and reference points.
The department of mechanical engineering (PMA) had experience in 'reverse engineering' and developed a module to mesh and optimise point clouds coming from the first module.
The department of architecture (ASRO) had experience in the use of computers in conservation practice (Van Balen) and developed a module to visualize, enrich and query the model built with the first two modules.
This paper will concentrate on this last contribution. The ones of ESAT and PMA are also presented in this symposium:
J. Schouteden et al., *Image-Based 3D Acquisition Tool for Architectural Conservation*
B. Lauwers et al., *The Application of Reverse Engineering Technology in Model Reconstruction for Conservation*

1.2 Recording

Heritage conservation is a complex activity. The study object is very often unique. It is certainly interesting to us for his peculiarities (Torsello). The survey (*largo sensu*) aims to recognize and record them (Feiffer). Every day new dimensions are considered. Specialists gather facts. These substantially analytical phases typically leads to some synthesis (interpretations or decisions). All aspects are interrelated. It is certainly interesting to organise the information in a common structure, to provide an *integration tool*.

Heritage conservation is partly *subjective*. The differences of attitude according to country or period illustrate very well the cultural dimension. A specialist interprets facts. A decision-maker weights contradictory requirements. But it is also true, that every respectable school of thought wish to base opinion on objective bases (Torsello). The present interest for cultural heritage strengthens this exigency with respect to conservation practice. What is not often clear enough (intentionally or not) is the position of the demarcation line between objective and subjective information. It is certainly debatable and conventional, but piece of information have admittedly different level of objectivity and a better view on them would contribute to clarify the decision-chain.

In the recording activity, the question of *time* is important to discuss. Antagonistic requirements weigh in. On one hand, faster surveying is desirable. Apart from the evident financial benefit, it can mean a lot indirectly: underprivileged context can be studied, new scale of problems tackled, and objects studied more in depth. The new techniques implied also may extend our vision on the reality. On the other hand, timesaving can be dangerous. A 'machine' delivering a complete model at the pressing of a button does not fulfil all the requirement of a proper survey. Surveying is a learning process. Time is necessary to understand. The user needs to assimilate the features recorded whatever the faithfulness of this recording. It is also clear that every aspect of surveying does not

bring the same amount of insight. The ideal solution would be to accelerate the less interesting aspects and to help and refine the more beneficial ones. In order to do that, new tools of understanding should be provided with automatic techniques.

If it is easy to create 3D model with point (0D) and curve (1D) entities, experience show that it is difficult to work with them. They remain always visible, even if in the reality they would have been hidden by some surface. They are also intrinsically poorer in content. It may not be problematic for façade surveys but it become tedious for objects having a pronounced third dimension (like buildings, vaults or some sites). Volume entities (3D) would be ideal. They have probably a clearer meaning (being for instance related to architectural elements). Unfortunately, one only sees surfaces. Ways exist to investigate (directly or indirectly) the inner structure of elements but they normally call for new hypotheses. Finally, surfaces (2D) are probably the best entities to deal with if one want to keep high 'objectivity'. Furthermore, information can be added. Geometrical dimensions are not the only ones. Surfaces have colours. Photographs can record them easily and systematically. It has to be noted that high dimension allows extraction of carefully selected lower dimension information.

There are some complications though. *Objectiveness is more a target than a fact*. Survey is a two-time process. Elements are first chosen and then measured. A qualitative moment is followed by a quantitative moment (Feiffer). In this way, a continuous problem is transformed in a discrete one. Each particular technique makes hypotheses about the behaviour of discretised elements. A wall can be considered vertical, inclined or bulged. Likewise, only a limited number of measurements is taken to survey curves or surfaces (some points usually). The quality of the results follows the quality of the two phases. Then come a problem with automatic systems. They are usually better in the 'quantitative' phase. They produce rough data, calling for qualification. Differences may be made between windows and walls, between sandstone and limestone.

The discrete character of surveying has not to be seen as a weakness. Reality conveys infinite amount of information. To grasp it, one needs to *abstract*. It is the surveyor's work to emphasise the important, to find an expressive representation of the situation. The variations of colours are subtle on a photograph but don't tell whether a stone is sand or limestone. An interpretation can produce readable representation (with the help of other data). Abstraction can range from a low level (type of stone) to a much higher level (value). Automatic systems - because of their usual lack of qualitative expertise, have to provide more information to allow extracting qualitative data. The precise measuring of edges (corners, mouldings) requires for instance a very dense automatic measuring pattern. This would not have been necessary if the shape was first analysed. But concomitantly, they may be more open to subsequent interpretations.

Figure 1.   **Diagram showing the structure of *Virterf* software**

1.3 Implementation

The ASRO module is built in four levels.

The *first level* uses an existing C++ visualisation library: VTK (Visualization Toolkit). It is an "open source, freely available software system for 3D computer graphics, image processing, and visualization". It provides basic methods to define, show and transform 3D data.

The *second level* implements the general structure of the module in a set of new C++ classes. An efficient low-level language allows computation intensive functions to perform fast.

The *third level* implements a set of commands giving access to the C++ classes of the second level. They are extensions of the Tcl scripting language (Tool Command Language). Scripting languages gives more flexibility to the user who can customize the code (not compiled). It also helps the developer who can test his program more easily. New Tcl commands can be written in C++. They are compiled in a library (DLL) that can be loaded in a Tcl interpreter, extending the functionalities. The code can be written manually but we used a freely available software tool, SWIG to automate the process. SWIG connects programs written in a variety of low-level programming language (such as C++) with a variety of high-level languages (such as Tcl/Tk). It automatically creates interface code using the original C++ classes' headers and files specifying the interface.

The *fourth level* uses the original Tcl commands; the one developed in the third level and Tk commands to build a user interface. Tk (toolkit) comes with Tcl (usually referred as Tcl/Tk), it allows to quickly create graphical applications.

The user can interact with the model at the third level (with commands typed in a console window) or at the fourth (with the graphical user interfaces). All the actions are recorded in a log file. This allows the user to reconstruct a previous stage of the model (not yet automatically). Log files are also a record of how something was done and by whom and open the possibility to judge about the quality of every piece of information of the model.

The ability to store set of commands in small files, Tcl scripts, allows storing the key used to produce a result instead of the results itself. This can save memory and remain valid if the model is changed.

Dealing with monuments, long-term concern is an issue. Text files are used for the databases. They always start with an explanation of the format. In this way, the chance to recover information in the future is maximized.
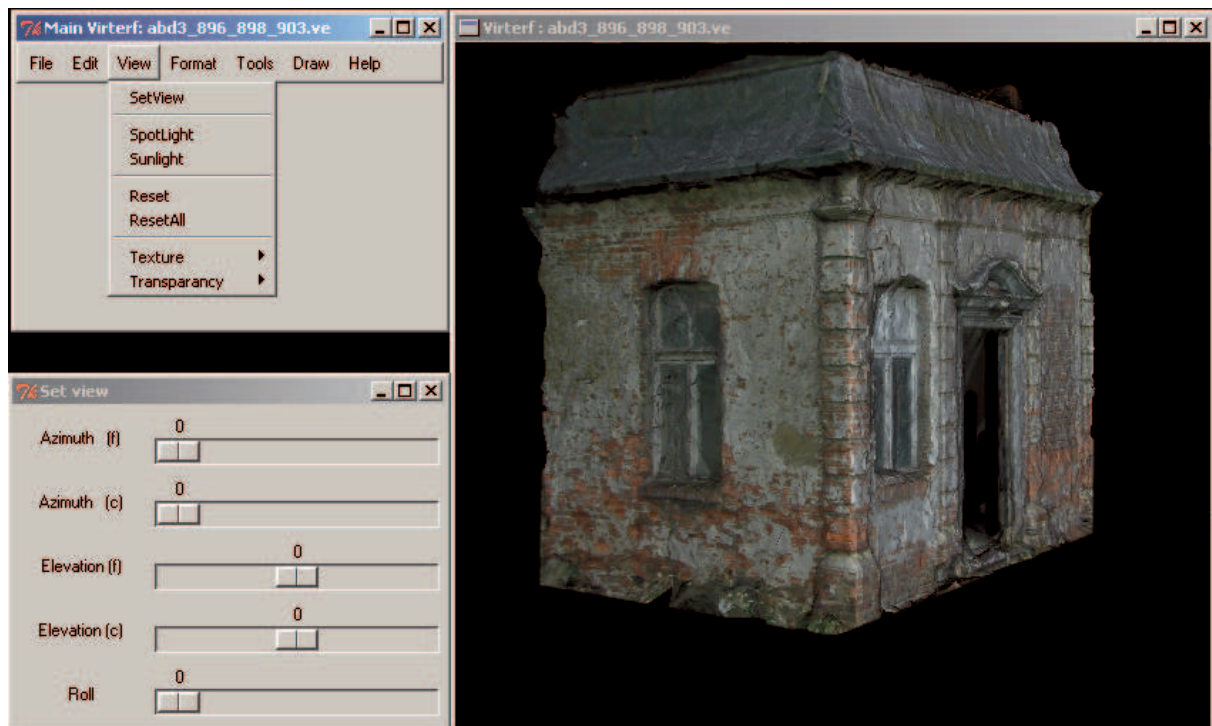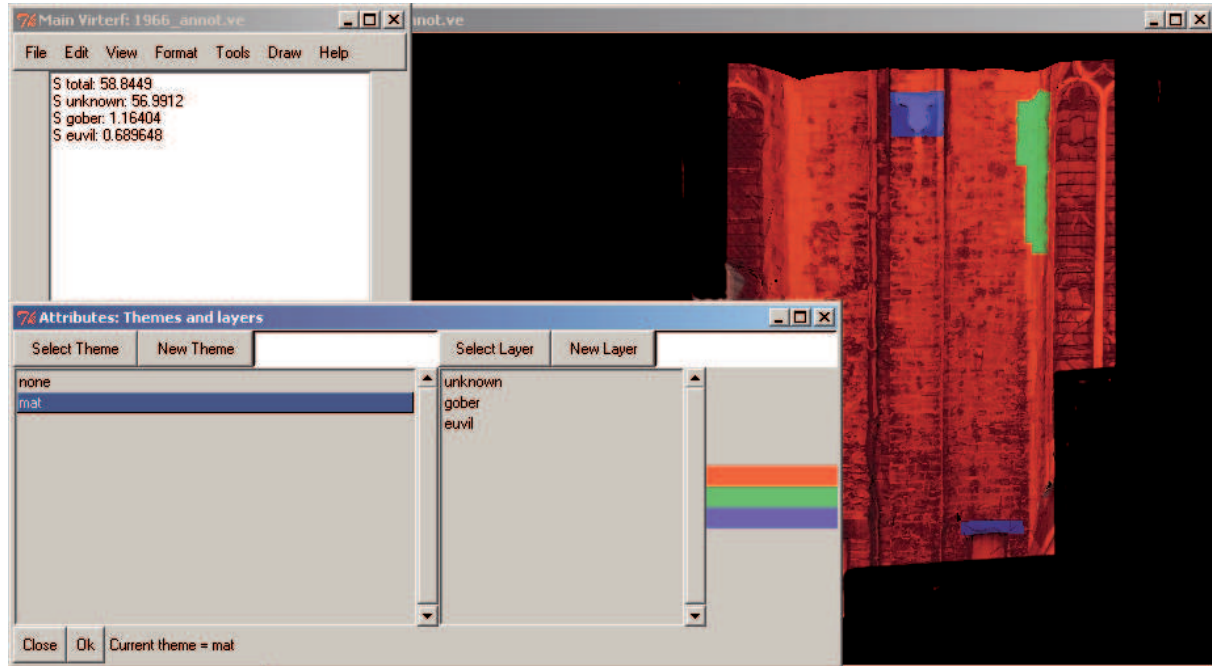
2 Model construction

Figure 2.   **Model of a Garden pavilion of Park Abbey in Leuven**

In the following two sections, the functionalities of the software will be presented.

The product of ESAT and PMA modules is a file describing a textured triangle mesh (figure 2). The file contains also information about the data origin. This forms the 'objective' part of the model.

The ASRO module allows the user to add more 'subjective' information, resulting from the work of the different contributors.
The information is organized in three groups: attribute themes (2.1), vector graphic themes (2.2) and links (2.3). Themes give a partial view on the model and are typically created by one specialist. Links are more flexible.

2.1 Attributes

Figure 3. **Façade of the nave of Antwerp Cathedral, material theme in construction and query on areas**

Attribute themes may group information on materials, pathology or construction phases. They are not imposed but created by the users, according to their needs. Themes are further divided in layers related to the attributes types (stone and wood in a material theme for instance). When a new theme is created, a layer 'unknown' is automatically assigned to all the triangles. The user must then create specific layers and qualify the geometry relating triangles to layers. This is a two times process: making a selection and choosing an attribute to apply to the selection. The colours of the deselected elements become lighter to give a feedback to the user.

Different options are available to define a set of geometrical data. Triangles can be selected (or de-selected) one at the time or using polygons (inscribed or round). A selection set can be redefined by intersection, union or difference with the triangles belonging to a given layer. It can also be inverted, made empty or include all the triangles. With this system it is for instance possible to select elements inside a given area, built in stone and prior to a given construction stage. As the model's construction progresses, adding new attributes becomes therefore easier. Small *Tcl* scripts can be written with all the commands necessary to easily reuse selection sets.

When a selection is made, an attribute can be assigned from the list of existing attributes (layers) from the active theme. The selection set remains active until the complete set is reselected. So more attributes can be applied to the same set. One element complicates the process: feature edges may not correspond to triangles edges. The transition line between brick and stone may for instance fall inside a triangle. If the triangles were very small (their dimension being of the order of magnitude of the details), it would not be so important, but it is unlikely to be so. A tool was therefore devised to refine the mesh. It uses a cutting segment defined on the model with the mouse. The triangles created receive by default the attributes and texture from the original triangle.

2.2 Vector graphics

Points and segments can be drawn. They are placed in the active layer of the active theme. They are drawn via picking or via coordinate introduction. Display points defined by the mouse (2D) are projected on the model (3D). It is possible to draw a segment connecting the two 3D points so defined or a 'polyline' intersecting the model. These tools allow the user to draw very quickly the joints of a stone, cracks or any other 1-dimensional features.

Section can also be drawn, selecting or defining a plane (via 3 points). The easiness of the process allows the user to quickly search for the best position. In just one command, it is possible to generate iso-lines of a vault.

Geometries can be imported and exported via DXF files. Of course, only the limited number of spatial elements supported could be imported. Original layers are mapped on layers of the active theme. If the names do not match, new sub-layers are created.

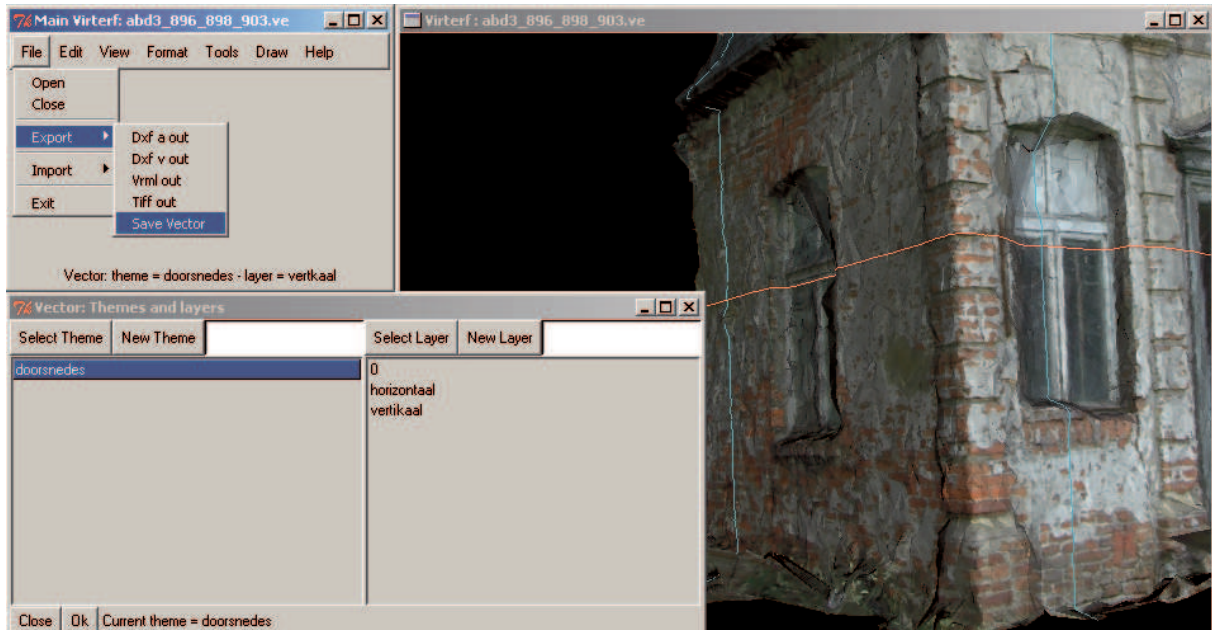These facilities allow to import results coming from other sources (deformation, FEM, …).

Figure 4.   **Model of a Garden pavilion of Park Abbey in Leuven, sections through the model in a vector theme**

2.3 Links

Not all the information related to an object is geometrical. Specialists produce reports. They also may want to annotate some parts of the model to clarify the meaning of layers or characterise special elements. Information is not necessarily textual. It can consist of images, figures, databases or *html* documents. These have normally relations with elements of the model (geometric or thematic). The possibility to define *links* was included to express them. Links are materialised by small spheres coloured according to the type of file linked. With the mouse, the user can query their name and a description or visualise their content via an external application. They are only visible in the themes relevant to them. It is probably not interesting to have direct access to an archaeologist report from a theme created by an engineer but one observation of the former may be of importance for the stability. If a specialist thinks that the attribute layers paradigm does not suit his needs (because precise borders are impossible to define for instance), he can create themes only to set links to relevant documents.
Links can be placed on the model or just hang in space. They can be saved in files.
The possibility to link to *html* documents allows the user to incorporate the *virterf model* in a more general scheme.
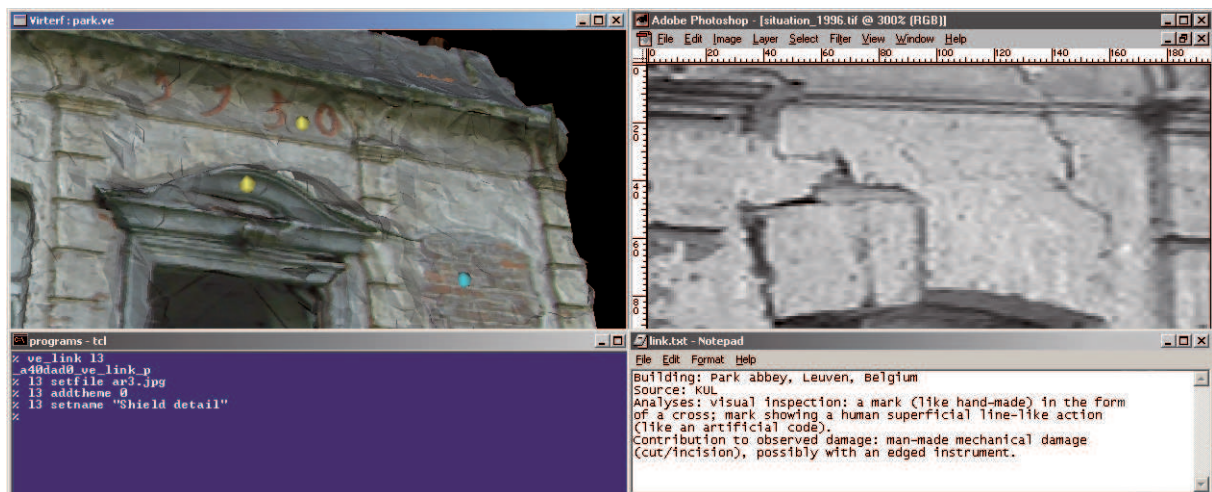

Figure 5.   **Model of a Garden pavilion of Park Abbey in Leuven, links to an image and to a text**
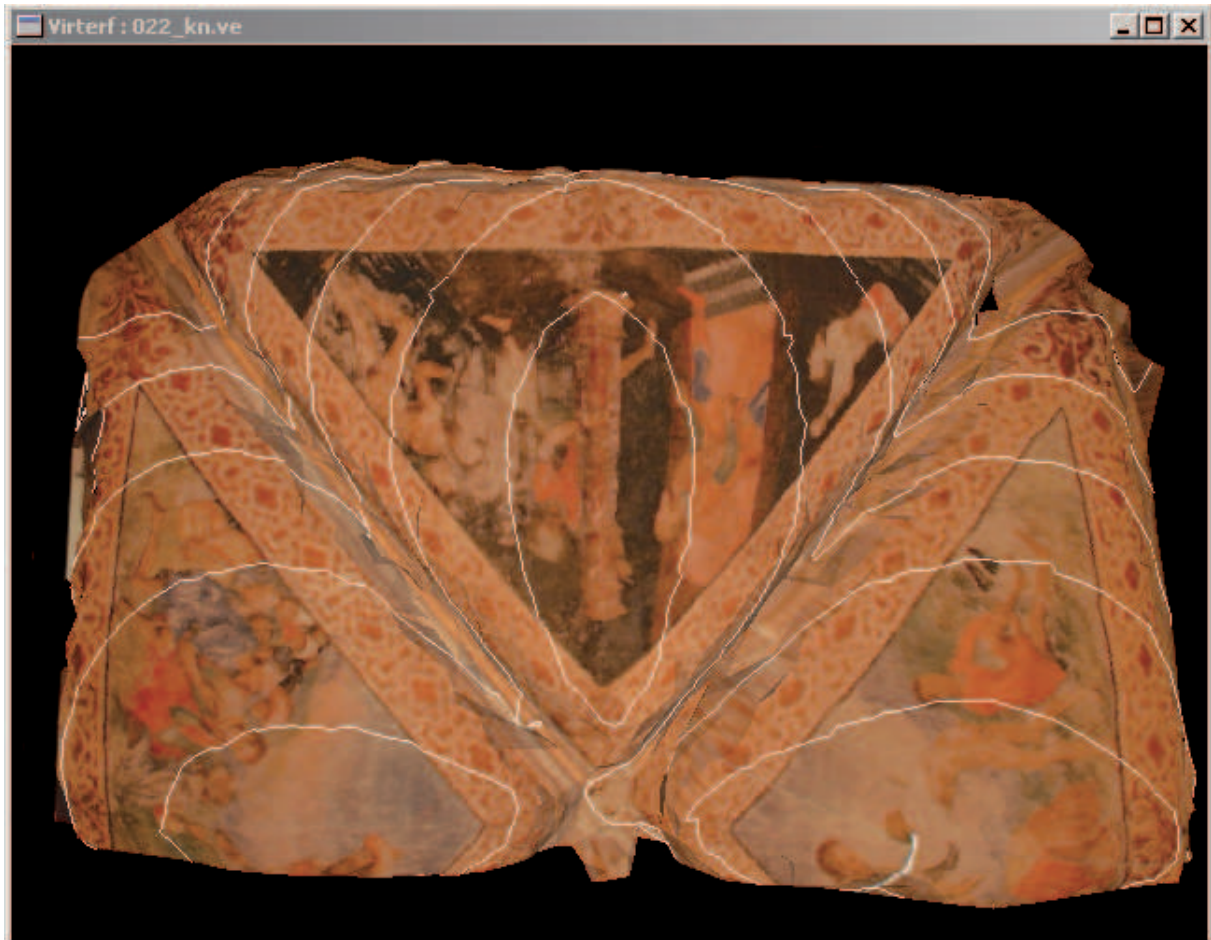
Figure 6. **Model of the vaults of the "Salle Héraldique" in the Arenberg Castle in Leuven, iso-lines generated**

3 Model querying

Model building is not an end in itself. The process of interrogating it is also part of the survey. Different tools are devised to help the user in this task.

3.1 Visualisation

The 3D graphic interaction with the user is made through 3D viewing windows, based on the VTK library. The window is a partial view on the database. It shows simultaneously the geometry ("objective" image, textured) and layers ("subjective" image).
Only one attribute theme and one vector theme can be shown at a time. In an attribute theme, the texture colour is changed to reflect the layer of each triangle. When no attribute theme is selected, the texture colour is the original one. The controls are classical (zoom, pan, rotate, set focus and camera position). They have an intuitive interface (mouse driven) and a numerical one to achieve a precise control. Projection can be central (perspective) or parallel. This can be interesting to produce ortho-photos.

In order to better communicate with external programs, different export facilities are offered. It has already been mentioned that vector graphic data can be exported via DXF files. Selected triangles of the active theme can be exported also like that. Themes layers are exported as DXF layers. TIFF or PPM files can be exported for illustration as well as VRML files.

The easiness with which one can zoom in and out may lead one to forget that *accuracy* is not perfect but related to the measuring process. The 'objective' basis is not perfect. It has not much sense to go to close to the object. A measure of the relation between zooming factor and point precision should be visible. We had not the time to implement this.

Another improvement possible would be to give the possibility to *project images* on the model. This would permit the use of old photographs to track changes, deformations and disappeared features. Special photograph technique (low-angled light, photo-interpreted images, IR, UV) may also put new features in evidence.

3.2 Geometrical querying

From a 3D window on the model, it is possible to retrieve information concerning segments or surfaces. Text results can be sent to the console window, text windows or redirected to files.

A segment is defined picking its extremities. Their coordinates are returned, the distance (d) is computed as well as the distance along the axes (d_x, d_y, d_z) and the orientation (altitude, azimuth and director cosines (c_x, c_y, c_z)). This tool can be used for very different purposes. Controls of verticality or deformation in general are particularly easy.

A surface is defined with the selection tool. The area tool then computes the total area (sum for the selected triangles), the absolute and relative area of the active theme's layers. It can be used for example to compute the surface of brick on a façade or the surface of the damaged tiles on a roof.

The database basis unit is the one chosen for the reference's topographic measurements. Additional user units are definable (for distances and surfaces). In some cases, they can bring insight to the compositional process. Numerical results from the model interrogation are given in the basis and possibly in the user unit system.

Querying on point was also foreseen but we had not the time to implement it. Giving a point, the idea was to retrieve an estimate of the absolute error on the position. If it corresponds to a node of the mesh, the estimate can be computed knowing the origin of the point (ESAT module does that already and reference information is stored in the database). If the point falls in the middle of a triangle the error depends of the precision of the apexes (giving a lower estimate) and of the quality of the mesh (more difficult to assess).

3.3 Database querying

The database querying is provided through the selection tool used to define a criterion, the visualisation capacities showing the triangles respecting the criterion, and the area tool giving quantitative information.

4 Conclusion

During all the progress of the project, regular meetings with a user group were held and helped adjusting objectives with real needs. The software was also tested on a set of test cases to control the technique's effectiveness. The model construction is certainly not entirely automatic, but still much *faster* than other techniques. The information contained in the photographs used to texture the model resulted particularly useful. It facilitates very much the definition of zones on the model, being so the key creation of an *interdisciplinary database*.

On the other hand, some problems remain. The software is still in a prototype form. Developments are necessary to transform it in a real product. The *accuracy* of the models produced was below our expectations. The reasons for that are recognised, technical and remediable (they are discussed by Schouteden et al.). The three modules are not yet very well *integrated*. They run on different operating systems (Linux and Windows) and offer different user interfaces. The user-friendliness should be improved. Good representation of complex geometries with triangular elements requires very fine meshes. This leads to models containing huge amount of triangles. For the time being, such models can be very difficult to handle in a typical PC, especially because the wish to dispose of an integrating tool with a high degree of interactivity requires swift operations. It can be forecast, without any risk, that this situation is temporary.
The problems mentioned are childhood illness and future will make them forget.

The instrument offered is very flexible. In particular, the structure of the database is simple and it should be quite easy to use other sources of geometrical information like reflectorless EDM measurements or laser scanning. To improve the software's effectiveness as an integration tool, an Internet interface and new import/export facilities could also be provided.

The software is used at the University of Bath for a project concerning Wells Cathedral.

Acknowledgements

Reference

English Heritage, *The Presentation of Historic Building Survey in CAD*, sl, sd

C. Feiffer, *Il progetto di conservazione*, 6ª ed., Milano, 1993

P. Torsello, *La materia del Restauro, tecniche e teorie analitiche*, Venezia, 1988

K. Van Balen, K. Nuyts, P. Smars, D. Van De Vijver, *Optimisation of models for the evaluation of the stability of vaulted gothic structures using deformation measurements and crack analyses* (research report, RLCC, K.U.Leuven), Leuven, 1995

IWT, <http://www.iwt.be>

SWIG, <http://www.swig.org>

Tcl/Tk, <http://dev.scriptics.com/software/tcltk>

Virterf, <http://virterf.asro.kuleuven.ac.be>

VTK, <http://public.kitware.com>