Surveying Software Tools for Parametric modelling of Architectural Heritage

Pierre SMARS*

Abstract

Various techniques are available to survey Historical Buildings: traditional hand measurements, digital photogrammetry, reflectorless total stations, laser scanners. . .

In certain circumstances (objects and requirements), any given technique can offer some benefit (quality, accuracy, speed, cost . . .). But in practical situations, the choice of a single technique is rarely the most efficient way to survey a building. In consequence, hybrid methods combining various techniques to produce efficiently accurate models appeared and their interest is now well recognised [1, 2, 3, 4, 5].

The *aim of the project* presented in this paper was to develop *software tools* facilitating total station measurements through (1) real-time control of the measurements, (2) parametric modelling of architectural elements and (3) use of digital photographs.

Real-time control is important because the construction of models from limited set of measurements requires the formulation of hypotheses (like the flatness of a wall or the regularity of an arch) and the best moment to test those hypotheses is when the surveyor is in direct contact with the object and has the possibility to adapt his measuring actions accordingly.

Parametric models are based on a set of shapes fitting measured points on the object. There are used in an attempt to use surveying techniques mimicking the construction process.

Digital photographs are used in parallel with the total station during the model construction phase. They also allow controlling the conformity of the model (test the modelling hypotheses). Various tools were developed to quickly orient photographs and to integrate them with the other measurements.

1 Design philosophy

The tools presented are meant to be used on the field. They are devised to be simple to use and robust. Their development started in the framework of previous projects [5, 6, 7, 8, 9, 10, 11] and they make large use of open source libraries (chiefly VTK [12] and VxL [13]).

The construction of a 3D model is a complex process, involving many steps. Each case is specific and flexibility is necessary. The idea is to develop sets of programs and modules achieving elementary tasks efficiently. In that way, the "pipeline" of operations can be optimised for a particular documentation problem, in function of the type of building, of the number and type of instruments available and of the number of operators and of their preferences (Fig.1).

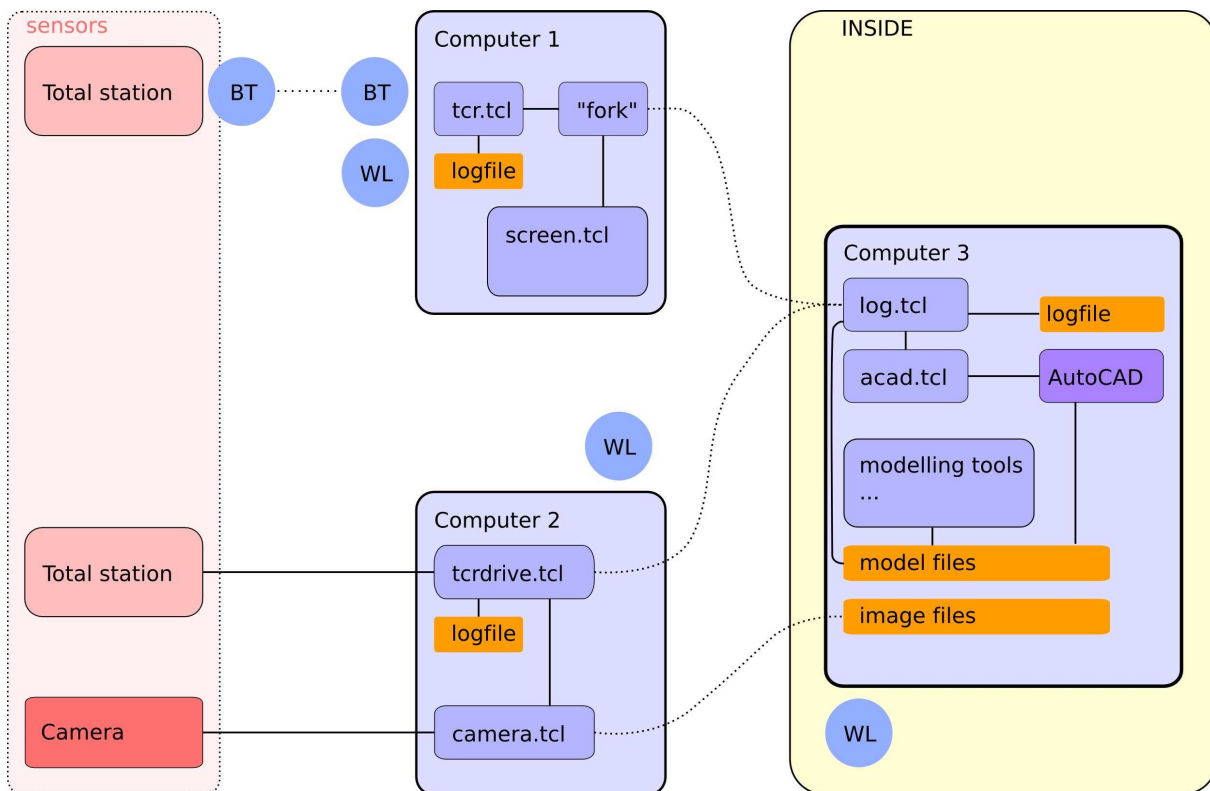*National Yunlin University of Science & Technology, Dou-Liu, Taiwan

Figure 1: Example of setup: 3 computers, 2 total stations and 1 camera

As modules and program are doing elementary operations, it is important to facilitate their inter-communication to achieve complex tasks.

2   Total station

Two modules (`tcr.tcl` and `tcrdrive.tcl`) and one program (`tcrout.tcl`) were developed to control total stations from computers and send the data to a designated target. The module `tcr.tcl` offers a graphical user interface (Fig.2). Measurements can be triggered either from the total station (usually the easiest), either from the software. The module `tcrdrive.tcl` is a server program accepting requests for the total station on a computer port. The program `tcrout.tcl` is used in case no computer is brought on the field and all the data was recorded on the internal memory of the total station.

The modules allow to measure points, lines, polylines and rays (rays are lines drawn from the instrument sensor to the estimated points; they are useful when the direction is trusted but not the distance, which is often the case for the measurements of edges).

3   Images

A module `camera.tcl` was developed to control a digital camera (tested with a Nikon D200) from a computer. It is basically a graphical interface to various external software. To communicate with the camera, it uses *Gphoto*2 [14], a command line software allowing to trigger photos, download them, and specify the settings of the camera from the computer. To correct the radial distortion of images, `camera.tcl` uses our software `unwarp` and
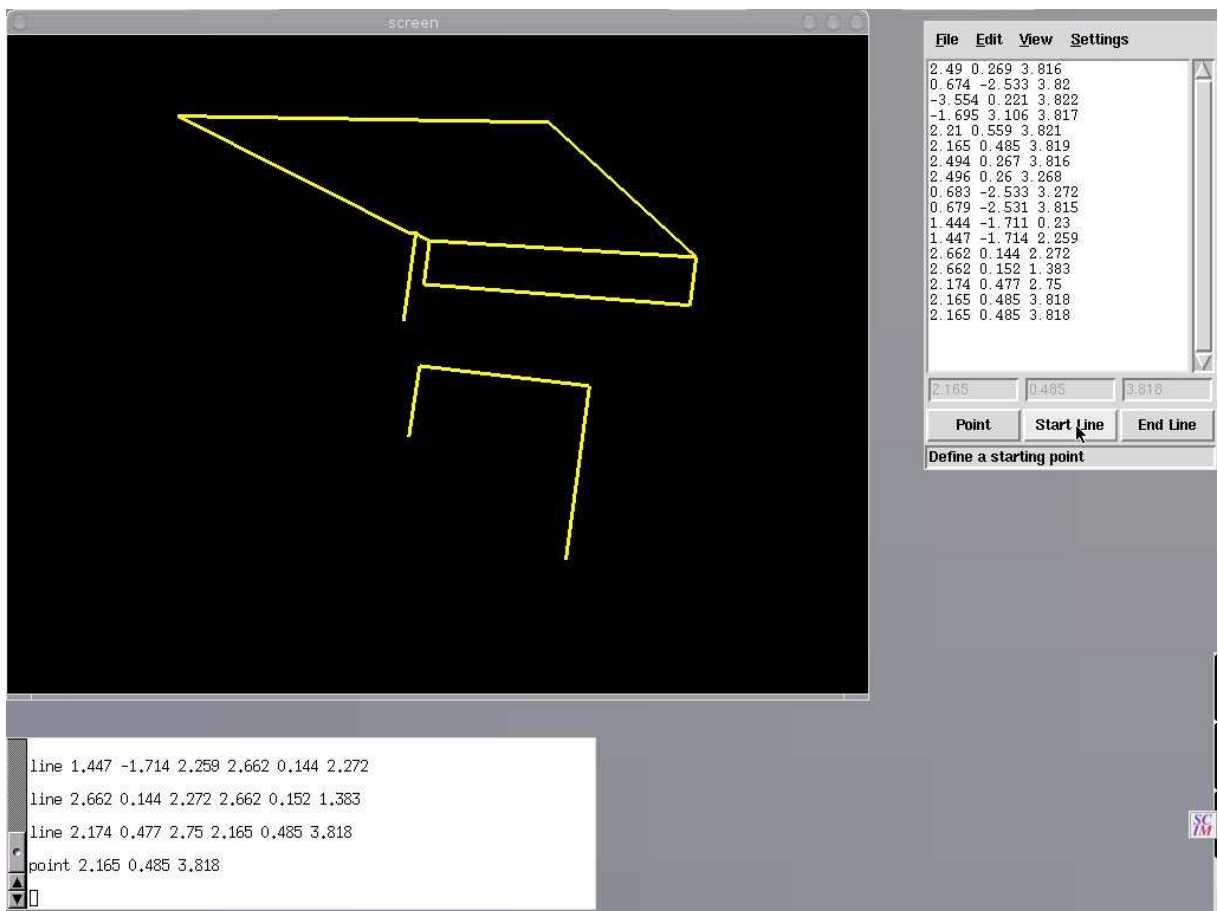
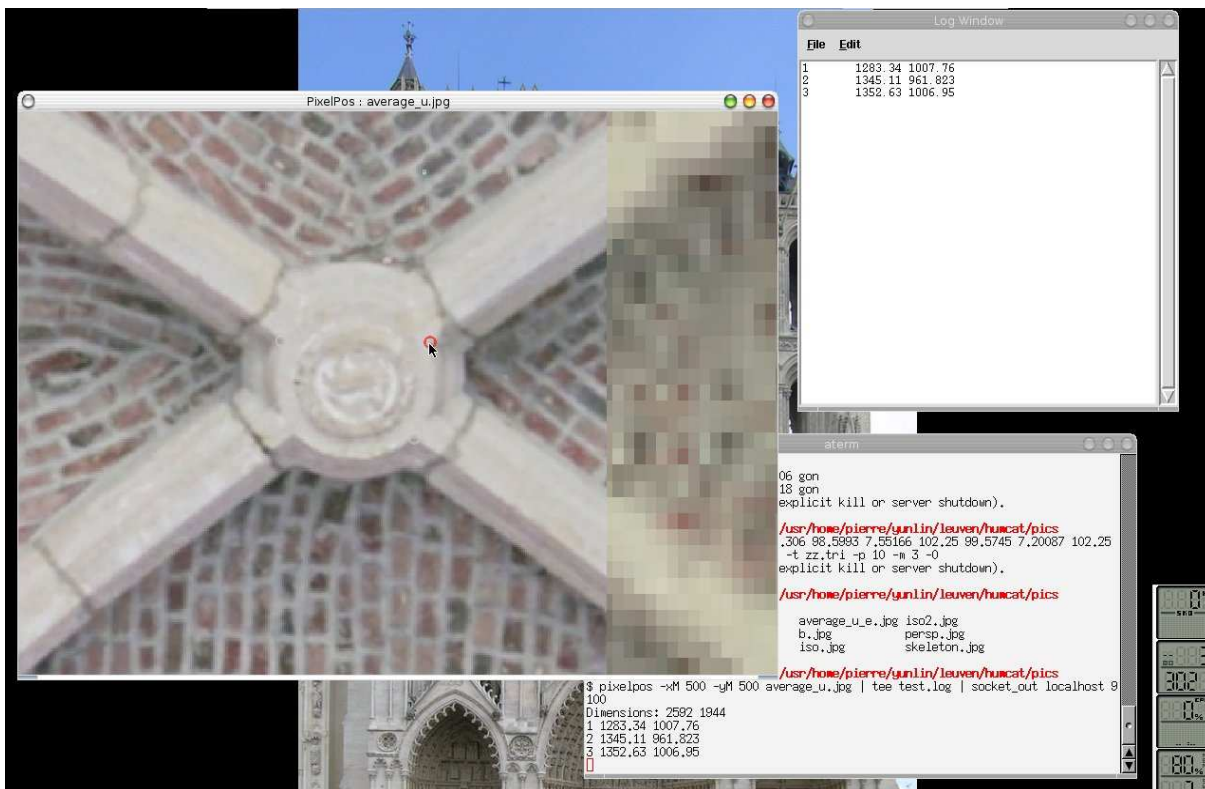Figure 2: `tcr.tcl` module in action with a `screen.tcl` window

Figure 3: Pointing with `pixelpos`

the parameters of the Camera/lens combination. To edit the image, `camera.tcl` opens it in *The Gimp* [15], an image manipulation program. To pick points on the image, `camera.tcl` opens it in `pixelpos` (Fig.3), a module developed to display an image and precisely measure points on it. The strength of the module lies in its speed. It is much faster to navigate in a large image, zoom up to the sub-pixel level and adjust lightness and contrast than using usual image manipulation programs. The module also offers the possibility to edit selections, to trigger the total station from `pixelpos` and to send the point correspondence to a file, to adjust lightness and contrast (non-destructive), and to open 48 bits images (useful to identify subtle features).

`camera.tcl` can also take series of 5 or 20 photographs and average them, possibly with different settings (to reduce noise and increase dynamic range, which is useful when working in dark interiors)

To use photographs for accurate measurements, it is important to know precisely its characteristics and the position of the camera when the photo was taken. A program `radial` was written to calculate the radial distortion in an image produced by a camera/lens combination. A program `p_compute` was developed to compute the camera projection matrix (P matrix) necessary to project photographs on 3D models. The required data is the set of measurements taken on the image and on the object from the `pixelpos` module.

Rectification

Sometimes, elements of decoration are plane and photographs only need to be deformed by a simple homography to represent the object on scale. Two programs `h_compute` and `h_remap` were developed for that purpose.

They were used for the documentation of complex plane decoration in a temple. Pho-
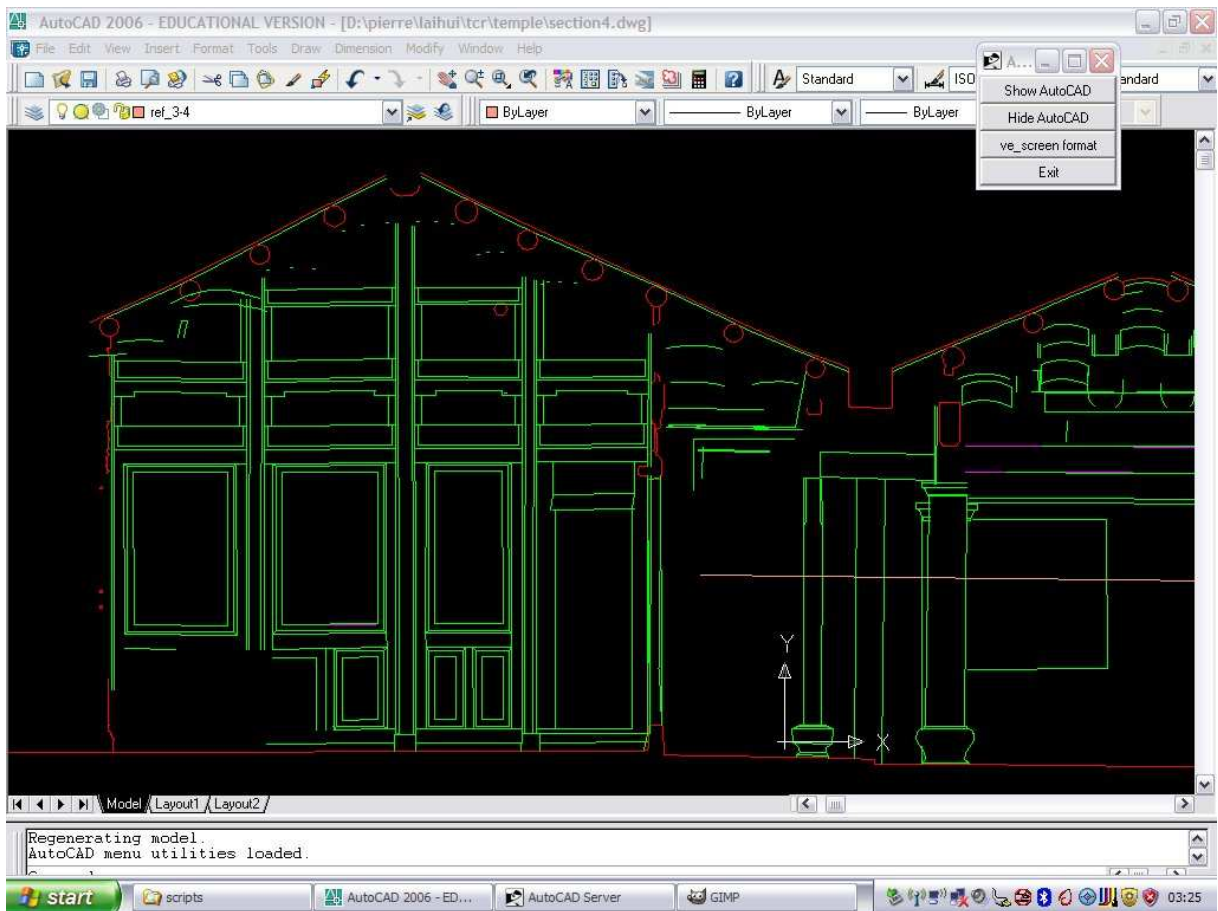
Figure 4: AutoCAD used with `acad.tcl` server module

tographs were taken and their radial distortion removed using `unwarp`. They were scaled using `h_compute` and `h_remap`. Using Gimp [15] and a Wacom tablet, lines were drawn on the photograph. A program `im2li` was used to vectorise the data and `li2dxf` to produce *.dxf* files which could be sent to AutoCAD.

4   Modelling

4.1   Visualisation

Two modules `ve_screen` and `ve_canvas` are used to draw and visualise 3D data (Fig.2). They can be controlled locally typing commands interactively or they can be operated remotely as a server accessible on a given port (`screen.tcl`, Fig.1).

Commands are used to draw elements (points, lines, polylines, triangle meshes), navigate the data, manage layers (create, erase, change colour, thickness...), project images on the model, export and import data (dxf, images)

A `ve_screen` is normally used to control measurements taken with the total station.

4.2   AutoCAD

The `ve_screen` module was tested for the documentation of a traditional house. This experience was interesting because it helped us identifying a few new problems which had

to be solved.

Because of the sun brightness, it was difficult to work outside with a computer. It resulted more efficient to place it on a table inside (at a short distance so that controls are still possible). A new module `acad.tcl` was developed to act as a server waiting for commands to be sent to AutoCAD, From the Measuring module `tcr.tcl`, this new module accepts the same commands of the `ve_screen` visualisation module but points and lines are sent to AutoCAD. Outside, another computer is used to run the measuring module `tcr.tcl` controlling the *total station* (a cheap eee-pc). The two computers are connected in a network.

Usually, measurements were actually not directly sent to AutoCAD. They were sent to a *socket_fork* redirecting the data in two directions: a 3D window on the local computer (used by the total station operator for control) and a buffer `log.tcl` where the data waits until the *AutoCAD* operator working on the other computer requires it explicitly. This setup is used to solve a problem of priority of access to AutoCAD and to offer the possibility to work with more than one total station if necessary (Fig.1).

4.3   Modelling tools

About 60 small (or not so small) programs were developed to manipulate points, lines, triangles and construct more complex shapes (`pt2tri`: mesh point clouds, `view_pt`: show point files in a 3d window, `view_li`: show line files in a 3D window `view_tri`: show triangle files in a 3D window, `append_pt`: append a set of point files, `append_li`: append a set of line files, `append_tri`: append a set of triangle files, `translate`: translate point, line or triangle files, `rotate`: rotate point, line or triangle files, `mirror`: mirror point, line or triangle files, `tridist` was developed to calculate the distance between a triangle mesh and the points of a point cloud, . . .). A complete list is available on-line.
`http://smars.yuntech.edu.tw/projects/survey/standalone.php`

A set of new programs was developed to fit specific shapes to point clouds: `fit_line`, `fit_plane`, `fit_circle`, `fit_sphere` and `fit_quadric`.

References

[1] S.F. El Hakim, J.-A. Beraldin, M. Picard, and A. Vettore. Effective 3D modeling of heritage sites. In 4th *international conference 3D digital imaging and modeling*, pages 302–309, Banff (Canada), October 6-10 2003.

[2] A. Wiedermann, J. Moré, and Rüdiger Tauch. Archimedes3D - an integrated system for the generation of architectural orthoimages. In *New perspectives to save cultural heritage, CIPA 2003 19th International Symposium, Antalya (Turkey)*, 30 September – 04 October, 2003.

[3] Ch. Ioannidis, N. Demir, S. Soile, and M. Tsakiri. Combination of laser scanner data and simple photogrammetric procedures for surface reconstruction of monuments. In *CIPA 2005 20th International Symposium, Torino (Italy)*, 26 September – 01 October, 2005.

[4] P. Ortiz, H. Sánchez, H. Pires, and J.A. Pérez. Experiences about fusioning 3D digitalization techniques for cultural heritage documentation. In *ISPRS, Commission*

V Symposium, Image Engineering and Vision Metrology, Dresden (Germany), 25-27 September 2006.

[5] P. Smars and D. D'Ayala. Parametric shape model for conservation oriented surveying. In *World Heritage in the Digital Age, Heritage Management Mapping: GIS and Multimedia, Alexandria (Egypt), 21-23 October*, 2002.

[6] P. Smars, K. Van Balen, and K. Nuyts. Layered Geometric Information System. In *18th International Symposium CIPA 2001 (Surveying and Documentation of Historic Buildings - Monuments - Sites Traditional and Modern Methods), Potsdam (D), September 18- 21*, pages 462–68, 2001.

[7] K. Nuyts, J.-P. Kruth, B. Lauwers, H. Neuckermans, M. Pollefeys, L. Qiongyan, J. Schouteden, P. Smars, K. Van Balen, L. Van Gool, and M. Vergauwen. Vision on conservation. In B. Fischer, K. Dawson-Howe, and C. O'Sullivan, editors, *Proceedings of the International Symposium on Virtual and Augmented Architecture (VAA'01), Trinity College, Dublin 21-22 June*, pages 125–132, 2001.

[8] K. Nuyts, J.-P. Kruth, B. Lauwers, H. Neuckermans, M. Pollefeys, L. Qiongyan, J. Schouteden, P. Smars, K. Van Balen, L. Van Gool, and M. Vergauwen. From a conservationist's point of view. In Kahmen V, Gruen, editor, *Proc. Conference on Optical 3-D Measurement Techniques, Vienna, October*, pages 179–186, 2001.

[9] K. Nuyts, K. Van Balen, and P. Smars. Virterf, a vision on heritage conservation. In *Seventh International Conference on Virtual Systems and Multimedia (VSMM'01), October 25 - 27, Berkeley, California (US)*, 2001.

[10] J. Schouteden, M. Pollefeys, M. Vergauwen, and L. Van Gool. Image-Based 3D Acquisition Tool for Architectural Conservation. In *18th. International Symposium CIPA 2001 (Surveying and Documentation of Historic Buildings - Monuments - Sites Traditional and Modern Methods), Potsdam, September 18- 21*, 2001.

[11] D. D'Ayala and P. Smars. Architectural and structural modelling for the conservation of cathedrals. *Journal of Architectural Conservation*, 3(9):51–72, 2003.

[12] Kitware. VTK, the Visualization ToolKit. `http://www.vtk.org/`.

[13] VxL, the vision-something-libraries, c++ libraries for computer vision research and implementation. `http://vxl.sourceforge.net/`.

[14] Gphoto, digital camera software. `http://www.gphoto.org/`.

[15] Gimp, GNU image manipulation program. `http://www.gimp.org/`.