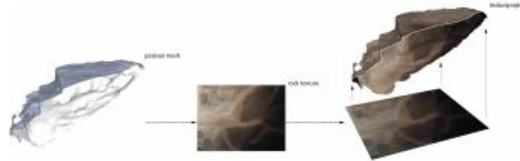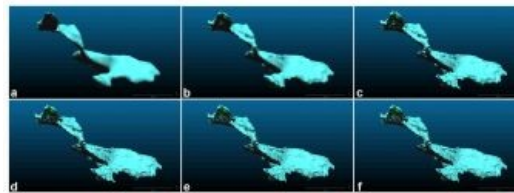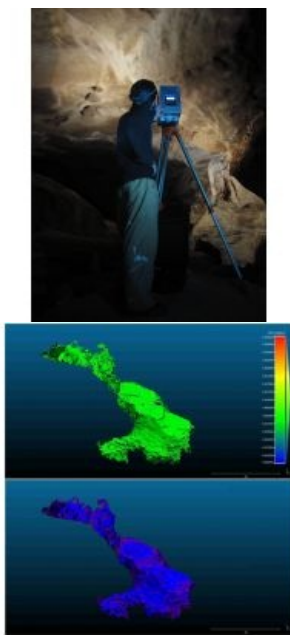## *3D RECONSTRUCTION OF THE POLLERA CAVE, ITALY*

High-precision Laser Scanning for Cave Tourism

The Pollera cave is situated in the region of Liguria, Italy, and is the subject of a strategy aimed at improving and promoting tourism in the area. In this project, 3D laser scans of some of the more easily accessible cave rooms were needed to create multimedia products such as movies, images and scenes for 3D virtual tours, and to generate geocartographic products for tourist excursions. To achieve this, the cave was scanned using a Z+F IMAGER 5010 high-precision laser scanner. Surveying and data processing proved to be particularly difficult and costly due to the underground environment, but by using free and open sourced software a fully textured 3D model was successfully generated.

Survey Planning and Laser Scanning

In normal conditions, the Z+F IMAGER 5010 laser scanner (Figure 1) can obtain millimetre accuracy on every single point and is generally useful for architectural applications. Instead of planning a topographic survey (that would have led to considerable logistical difficulties), the locations and number of survey positions were strategically chosen using a cave map and the experience of the speleologists. Despite careful planning of the scanning positions, in some places it was almost impossible to establish a proper position. In total, eleven scans in high resolution and high quality mode were needed in order to detect the entire cave surface.

Figure 1, Z+F IMAGER 5010 laser scanner inside the Pollera cave.

3D Laser Scanning

Because individual scans would eventually be combined into a single 3D model, and since natural reference points were not obvious in the cave environment, physical reference targets (special support trestles) were placed in order to aid in point cloud registration. Due to the peculiar and complex formation of the cave, only a small number of targets were placed in easily accessible locations so that they could be visible from multiple scans.

Point Cloud Registration

The point clouds from the eleven scans were registered with Z+F LaserControl, the software of the instrument. When scene geometry is simple (like in cities), it is easy to match two scans. However, in the case of more complex geometry such as in the cave, more obvious and distinguishable tie points need to be chosen as a first approach to matching point clouds. Unfortunately, in this case, some of the reference targets were not easily identifiable or visible from multiple scans, and most of them were placed at the same height due to practical constraints. To mitigate the effects of this, specific natural cave features (such as little rocks or other distinct cave characteristics) were carefully identified as additional targets. The registration was carried out with a mean error of 6cm for the whole project.

Figure 2, From a to f: various reconstructed meshes as octree depth parameter changes between 6 (a) and 11 (f).

Mesh Creation

A 3D mesh was generated using the registered point cloud. In this project, the results of two different types of open source software, MeshLab and Cloud Compare, were compared. Both types of software are designed to process and manage point clouds and meshes, but MeshLab is oriented towards working with meshes whereas CloudCompare is more suitable for working with point clouds. Although there are many algorithms to generate a mesh from a point cloud, it was decided to use the Poisson algorithm because it was present in both types of software. Using the registered point cloud, a mesh was created using MeshLab. To validate this mesh and to see whether the algorithms worked in the same way, a second mesh was created with CloudCompare using the same point cloud. This also helped to verify whether it was possible to optimise the whole procedure using only one type of software.

To generate the mesh in CloudCompare, the most important parameters to be set were octree depth and samples per node. The octree depth is the maximum depth of the tree used for surface reconstruction; higher numbers mean higher precision in reconstruction but also higher processing times (Figure 2). 'Samples per node' specifies the minimum number of sample points that should fall within an octree node. Small values can be used for noise-free samples, but for noisier samples larger values may be needed to provide a smoother reconstruction. In this case, since a subsample of points was taken, it was assumed that the sample was noise-free.

Figure 3, Top: M3C2 signed distances between the MeshLab mesh and the CloudCompare mesh vertexes clouds. Bottom: Significant changes between the clouds (blue: no significant change = smaller than local Level of Confidence 95%; red: significant change).

Comparing Meshes

The MeshLab mesh was compared against the CloudCompare mesh in the CloudCompare environment (Figure 3). CloudCompare uses the vertexes of the two different meshes and two tools to measure difference. First, the M3C2 plug-in computes the signed distances between the reference cloud (MeshLab mesh vertices) and the vertices of the CloudCompare mesh. The main parameters for distance can be derived using the 'param guess' to set the values of normal scale (to orient a cylinder, inside which equivalent points in the compared cloud will be searched for), the projection scale (the diameter of the cylinder) and the max. depth (the cylinder height) parameters. The whole MeshLab vertexes cloud was used as core points, to compute the distances in the entire cloud. In this case it was decided to compute two different distances: one with default normals (using the normal scale parameters) and the other using vertically oriented normals, in pseudo 2D areas of the cave. The calculated distances have a centimetre order of magnitude.

The C2M tool searches the nearest triangle in the reference mesh for each point of the compared cloud. Meshes provide information using the normals, so cloud-mesh distances have a sign (+ or -) stored as scalar field in the compared cloud; they can also be split along the 3 main axes (X, Y and Z), generating three scalar fields, one for each axis. To obtain a comparison between different meshes that is easier to interpret, portions of the 3D model and 2D cross sections were extracted. 3D portions were generated with the segment tool simultaneously cutting the two point clouds on the ceiling (above the plateau), the plateau (internally), the entrance and the slide. Real distances were calculated with M3C2, using normal default and vertical normal, and with the C2M tool.

Figure 4, Colouring the mesh with SketchUp.

Cave Image Projection

Google SketchUp was used to drape cave surface images as texture to the mesh, thus enhancing the visualisation (Figure 4). This process should be done for every area of surface that needs to be covered, meaning that more photo coverage will result in better visualisations. It is best to take narrow overlapping shots of the cave surface (ceiling, ground and walls). Each surface of the cave should be photographed in separate batches and the image should be taken with the lens perpendicular to the surface to minimise distortions.

Conclusion

What emerged from this experience is that MeshLab seems less intuitive but provides a larger range of filters, useful plug-ins and processing tools than CloudCompare, which is a faster and easier environment to manipulate even huge point clouds. The use of four different types of software (Z+F LaserControl, MeshLab, CloudCompare and SketchUp) was an ideal solution to carry out post-processing, from the registration to the creation of a textured 3D model. A suggested solution for the future consists of making a scan at the highest resolution to be used just for the registration phase. Then, to achieve reasonable processing times in mesh reconstruction, it could be useful to resample the point cloud, keeping only one or two out of ten data points.