Enhancing processing and visualization efficiency of 3D Scanned Meshes

(Invited Paper)

M. Callieri, P. Cignoni, A. Fasano, C. Montani, P. Pingi, F. Ponchio, R. Scopigno
Visual Computing Lab
Istituto di Scienza e Tecnologie dell'Informazione (ISTI), C.N.R., Pisa, Italy
Email: r.scopigno@isti.cnr.it

***Abstract*— The construction of detailed and accurate 3D models is made easier by the increasing diffusion of 3D scanning devices. These allow to build accurate digital models of real 3D objects in a cost- and time-effective manner. The talk will present the capabilities of this technology focusing mainly on some issues which are preventing its wider use, such as the considerable user intervention required, the usually incomplete sampling of the artifact surface and the complexity of the models produced. Another emerging issue is how to support the visual presentation of the models (local or remote) with guaranteed interactive rendering rates. Some examples of the results of current projects, mainly in the Cultural Heritage field, will be shown.**

I. Introduction

Modern 3D graphics technologies allows us to acquire accurate digital models of real objects or complex scenes; moreover, 3D graphics allows also to present those digital data to the public in an interactive and pleasant manner.

3D scanning technology evolved considerably in the last few years, both in terms of hardware devices and of algorithms for processing the raw data produced by scanning devices [1]. 3D scanning devices are usually based on optical technology (laser or structured light) and use either the *triangulation* approach (small and medium scale objects) or the *time of flight* approach (large scale objects, e.g. architectures).

The scanning of complex objects is therefore performed by taking a [usually large] set of partially overlapping range scans. The classical pipeline which characterizes a 3D scanning session is rather complex, involving many different operations (introduced in Section III).

Once we have reconstructed a digital 3D model of the scene or object of interest, some issues arise from the very dense sampling resolution granted by modern scanning devices. Being able to sample ten 3D points per squared millimeter (or even more) is of paramount value in many applications which need a very accurate digital description. On the other hand, all these data is not easy to process, render and transfer.

This technology opens great opportunities for a very broad set of applications (movie/animation, medicine, industrial inspection, urban and terrain management, cultural heritage, design, etc.).

Our group focused in the last few years on the software problems introduced by the need to process efficiently the huge dataset produced with 3D scanning devices. We have proposed some solutions, briefly described in the following. We describe more in detail two issues: how to improve the automation of the post-processing phase (to minimize the human-assisted phases) and how to present complex 3D data with both extreme efficiency and simple interaction. Our main field of application and assessment of the technology developed has been the Cultural Heritage (CH). Accordingly, most of the examples or figures comes from this application domain.

II. Previous Work

Many previous works concern the use of 3D technology either to reconstruct digital 3D models of Cultural Heritage masterpieces or to present those models through digital media. An exhaustive description of those works goes well beyond the brief overview that we can draw in this section. We prefer to cite here only some seminal papers on the technologies proposed for 3D scanning and interactive visualization.

Automatic 3D reconstruction technologies have evolved significantly in the last few years. An overview of 3D scanning systems is presented in [2]. Unfortunately, most 3D scanning systems do not produce a final, complete 3D model but a large collection of raw data (*range maps*) which have to be post-processed. The post-processing pipeline is presented in the excellent overview paper by Bernardini and Rushmeier [1]. Many significant projects concerning 3D scanning and Cultural Heritage have been presented in the last few years [3], [4], [5], [6], [7], [8].

The high resolution meshes produced with 3D scanning are in general very hard to render with interactive frame rates, due to their excessive complexity. This originated an intense research on: simplification and multiresolution management of huge surface meshes [9], [10], [11]; and interactive visualization, where both mesh-based [12] and point-based solutions [13], [14] have been investigated.

III. Processing scanned data

Scanning any 3D object requires the acquisition of many shots of the artefact taken from different viewpoints, to gather geometry information on all of its shape. Therefore, to perform a complete acquisition usually we have to sample many *range maps*; the number of range maps requested depends on the surface extent of the object and on its shape complexity. This set of range maps has to be processed to convert it into a single, complete, non-redundant and optimal 3D representation. The processing phases (usually supported by standard scanning software tools) are:

- range maps *alignment*, since by definition range map geometry is relative to the current sensor location and has to be transformed into a common coordinate space where all the range maps lie well aligned; after alignment, the sections of the range maps which correspond to the same surface zone will be geometrically overlapping;
- range maps *merge* (or fusion), to build a single, non redundant mesh out of the many, partially overlapping range maps;
- mesh *editing*, to improve (if possible) the quality of the reconstructed mesh;
- mesh *simplification*, to accurately reduce the huge complexity of the model obtained, producing different high quality Level Of Details (LOD) or multiresolution representations;
- *color mapping*, to produce textured meshes which couple the geometry of the object with its appearance representation.

At ISTI-CNR we have designed and implemented a suite of scanning tools (*MeshAlign*, *MeshMerge*, *MeshSimplify* [15]) which support all the post-processing phases described above. The second generation of our tools has been produced in the framework of the EU IST "ViHAP3D" project (2002-2005).

MeshAlign allows the registration of multiple range maps; it adopts a classical approach based on first, a *pairwise local* and then a *global* alignment [16]. This canonical approach has been implemented with a number of innovations to reduce the user contribution, to improve efficiency and easy of use, and finally to support the management of a large number of range maps (we processed range dataset containing up to six hundreds range maps).

The alignment task is the most time-consuming phase of the entire 3D scanning pipeline, due to the substantial user contribution required by current systems. The initial placement is heavily user-assisted in most of the commercial and academic systems (and it requires the interactive selection and manipulation of the range maps). Moreover, this kernel action has to be repeated for all the possible overlapping range map pairs. This pairwise process can be considered as a graph problem: given the nodes (i.e. the range maps), we have to select a subset of arcs such that every node is linked to some others if they have to be aligned together. If the set of range maps is composed by hundreds of elements (the scanning of a 2 meters tall statue generally requires from 200 up to 500 range maps, depending on the shape complexity of the statue), then the user has a very complex task to perform: for each range map, find which are the partially-overlapping ones; given this set of overlapping range maps, determine which one to consider in pair-wise alignment (either all of them or a subset); process all those pair-wise initial alignments.
Our goals in the design of *MeshAlign* were:

- to support the management of really large set of range maps (from 100 up to 1000); this can be obtained by both providing a hierarchical organization of the data (range maps divided into groups) and by using multiresolution representation of the data to make rendering and processing more efficient;
- since the standard approach (user-assisted selection and initialization of all the overlapping pairs and the creation of the correspondent alignment arc) becomes impractical on large set of range maps, we planned to provide instruments for the automatic setup of most of the required alignment arcs;
- finally, provide visual/numerical presentation of the intermediate status of the alignment process and of the accuracy reached.

MeshMerge [15], our volumetric reconstruction tool, is based on a variant of the volumetric approach [17]. *MeshMerge* can manage large range map set (many million sample points) on low-cost PC platforms with a very good efficiency. Data fusion is performed by the weighted integration of the range maps, and small holes (region not sampled by the scanner) can be optionally filled. Since the adoption of a volumetric approach requires a very large memory footprint on big dataset, *MeshMerge* provides a *split-reconstruction* feature: to process huge dataset it works on sub-sections of the data (out-of-core), loading each time only the range maps involved in the generation of that single section of the voxel set. The various parts of the final model are joined after the split-reconstruction process with a small time overhead; the boundary of the sub-blocks are guaranteed to be identical so the joining of resulting sub-meshes is trivial.

The reconstructed models (when produced using a voxel size equal or smaller than the inter-sampling distance used in scanning) are usually huge in size (i.e. many millions faces). Most applications require significant complexity reduction in order to manage these models interactively. Two problems arise when we try to simplify such models: we need a solution working on external memory to cope with these big models; simplification has to be accurate if we want to obtain high-quality multiresolution models and accurate visualization [12].

Our *MeshSimplify* tool [11] has no limits in terms of maximal size of the triangle mesh in input, sice it adopts an external-memory approach; at the same time, it ensures high-quality results, since it is based on edge collapse and takes into account both geometry accuracy and shape curvature [9], [10].

Finally, the *Weaver* tool [18] supports the reconstruction of textured meshes from a sampling of the object appearance. We usually perform the acquisition of the *apparent color* (reflected, illumination-dependent) using digital photo cameras, which is the easier and more practical approach since acquisitions in lab conditions (e.g. controlled lighting) are often impossible in the CH field. To map color data on the 3D model *Weaver* computes first the inverse projection and intrinsic parameters for each photo (from the image to the 3D mesh). Then, it computes an optimal coverage of the 3D mesh with sections of the original images, packs all the used

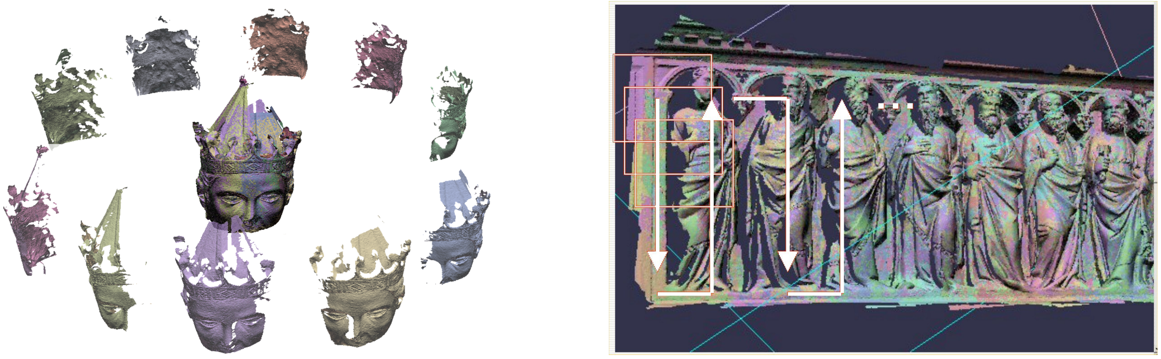


Fig. 1. Range maps are taken in a row-wise order: an example of *circular* stripe around a statue's head (left); an example of *raster-scan* scanning order adopted for the acquisition of a bas-relief (right).

portions in a new texture map and stores UV parameterization in the traingle mesh. Finally, it reduces color (hue/intensity) disparity on boundaries between overlapping photo parcels.

IV. Making alignment an automatic process

Solutions for a completely automatic scanning system have been proposed, but either these systems are based on the use of complex positioning machinery, or adopts silhouette-based approaches which do not guarantee the needed accuracy. An alternative approach is to design new solutions for the classical scanning pipeline which would transform those phases into an unattended process. In particular, the range maps registration phase is the only task where a considerable human intervention is still requested. Several papers proposed methods for automatic alignment, usually based on some form a shape analysis (see [19] for a survey paper).

In designing a new solution [20], we started from a few initial conditions directly gathered by our experience in 3D scanning. First, the *detection of overlapping range maps* can be reduced to a simpler task: 3D acquisitions are usually done by following simple scanning pose paths. Users usually acquire range maps in *stripes*, following either a *vertical*, *horizontal*, *raster-scan* or *circular* translation of the scanning system (see Figure 1). The different types of stripes share some common properties: they contain an ordered set of n range maps, such that range map R_i holds a significant overlapping with at least R_{i-1} and R_{i+1}. Vertical, horizontal or raster-scan stripes are often produced when acquiring objects like bas-reliefs, walls or planar-like items. Circular stripes are indeed more useful when acquiring objects like statues, columns or cylindrical-shaped objects.

If we can assume that the acquisition has been performed using one of these stripe-based patterns, then we may search for overlapping and coarse registration on each pair of consecutive range maps (R_i, R_{i+1}). From the point of view of the registration algorithm, all the stripes pattern defined above are equivalent: an automatic registration module can processes each couple (R_i, R_{i+1}), to produce in output the roto-translation matrix M_i that aligns R_{i+1} to R_i.

The subset of registration arcs defined above is usually sufficient for a successive ICP application, since our *MeshAlign* is able to complete the needed arcs (interconnecting R_i with all the overlapping range maps) in an automatic manner. *MeshAlign* maintains a *spatial indexing* technique, which stores for each 3D grid cell the set of range maps passing through that region of space. The initialization of this data structure requires the scan-conversion of every range map in the discrete space. We can easily retrieve groups of overlapping range maps by a simple visit of the bucketing structure, and tell how significant are those overlap extents. Given the occupancy grid information and once a single alignment arc is provided for each range map, our registration system is able to introduce all needed arcs (in a completely unattended manner), by selecting and processing only those which satisfy a minimum-overlap factor.

In conclusion, the stripes approach can be seen as an efficient working strategy, in opposition to the more general task to determine a complete adjacency graph. This approach reduces a $1\ over\ n$ into a $1\ over\ 1$ problem (for each range map, find coarse registration matrices for just the subsequent one in the stripe ordering).

To solve the rough registration of range map R_i over R_{i+1}, we have developed an efficient shape characterization kernel which works directly on the discrete range map space. Like other surface matching algorithms, we look for a small set of *feature points* which characterize the first range map R_i. For the sake of simplicity we consider our input meshes as regularly sampled *2D height fields*. A point-based shape description kernel is defined as follows: for each point p we select a small and regular kernel of adjacent samples (13×13); each element $k_{i,j} \in K_p$ contains the dot product of the pivot's normal vector and the normal vector of the corresponding pivot's neighbor. Then, we calculate the variance of each kernel $K_p \in R_a$:

$$s^2(K^p) \quad = \quad \frac{1}{n^2} \sum_{i,j} (k^p_{i,j} - \mathbf{E}[K^p])^2 \qquad (1)$$

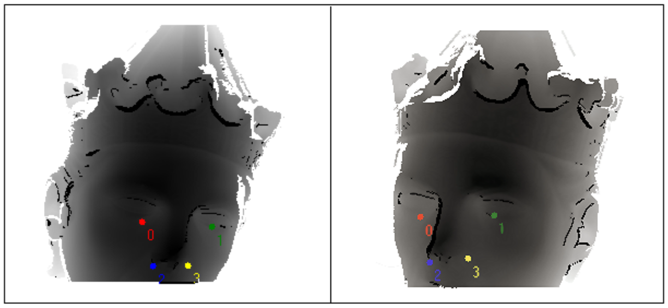


Fig. 2. The four matching point pairs selected by the algorithm on two range maps.

where $\mathbf{E}$ is the average of each kernel:

$$\mathbf{E}[K^p] = \sum_{i,j} \frac{k^p_{i,j}}{n^2} \qquad (2)$$

The variance is used to cluster all range map points in buckets characterized by a similar surface curvature. Low values of $s^2(K^p)$ are relative to flat areas where the normal vectors are relatively uniform. On the other hand, high values of $s^2(K^p)$ correspond to zones having high curvature. Note that if a mesh has open boundaries, then vertices on the proximity of these boundaries produce a high variance value, due to the absence of information on kernel points lying outside the surface. We have chosen to discard all the points having either high or low variance, using two opportune threshold values selected according to empirical experience. Then, a small set of candidate starting points (around 20) are chosen randomly among the remaining points (the ones with medium variance).

In a second step, for each of these k points on R_i we search for the potential corresponding points on the second mesh R_{i+1}. Our method builds up on the same kernel defined in the previous subsection. In particular, we compute the kernel for every vertex $q \in R_b$. Given $p \in R_a$ and its kernel K_p, the metric consists in finding the more similar kernel K_q relative to the point $q \in R_b$. So, for each K_q, we compute the squared difference with K_p:

$$d^2(K_p) \quad = \quad \frac{1}{n^2}\sum_{i,j}(k^p_{i,j} - k^q_{i,j})^2 \qquad \forall q \in R_b \qquad (3)$$

and we choose as best potential matching point the one having minimum distance $d^2(K_p)$.
This kind of metric is invariant with respect to the usual transformations (translations and rotations) that occur to the meshes belonging to a strip. This metric is not invariant to consistent rotations over the view direction of the scanning device. However in standard 3D scanning rotating the scanner along his view axe is rather uncommon (the scanner is usually connected to a tripod, which makes impossible to apply a substantial rotation along the view axe).

Finally, out of those possible k pairs we choose the group of four matching points which gives the best coarse alignment (see Figure 2); if the needed accuracy is not reached, we iterate until convergence by selecting and checking a few different points.

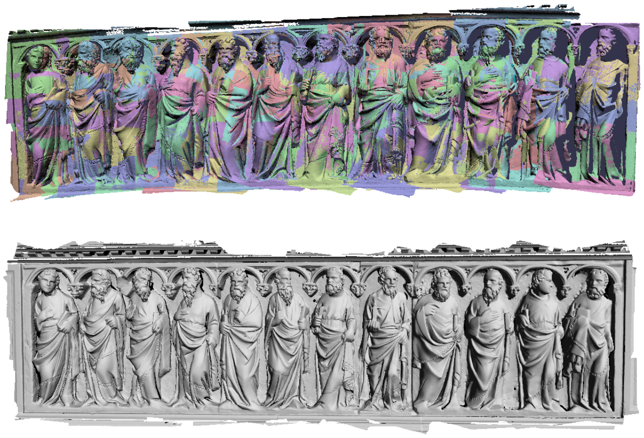


Fig. 3. The coarse alignment of the bas-relief *(top)* and the final model *(middle)*; almost all of the alignments required just 1 iteration.

The proposed registration algorithm was tested on many large datasets coming from real scanning campaigns (each range map contains therefore real raw data, usually affected by noise, artifacts and holes). After the automatic selection of an initial coarse registration matrix with the proposed algorithm, all datasets were finely aligned (pairwise local and global registration) using *MeshAlign*.

An example concerning a bas-relief is shown in Figure 3, whose approximate length is 2.5 meters; in this case two raster-scan (snake-like) stripes were acquired, for a total of 117 meshes (about 45.5M vertices). The overall alignment required 1h:50min (on a Pentium IV 2.4GHz), i.e. much less than the raw scanning time (approximately 4 hours in the case of the basrelief) . The solution presented is sufficiently fast to run in background during the acquisition, processing all the scans in sequence as soon as they are produced.

V. Hole detection and best next view

Producing a complete, hole-free model of a real object with 3D scanning is not an easy task. Marc Levoy noted [3] that scanning in a complete manner the surface of a complex object is very hard (often impossible) and the progressive coverage of the surface comes with an effort which is inversely proportional to the unsampled fraction. A number of papers have been proposed to cope with this problem. Small holes can be easily filled in reconstruction, by adopting one of the many reconstruction methods which allows to fill gaps. But in some applications (CH, medical, etc) it is not so nice to "guess" data following some mathematical approach.

To improve the coverage of the current scan set, we can devise automatic solutions for hole detection and *next best view* determination. We have recently investigated such topic and we have proposed a couple of solutions. The first one is *geometric* [21]: regions with holes are detected, the corresponding surface normals are selected and by clustering surface normal vectors we produce a set of directions for next view, sorted w.r.t. a criterion based on percentage of missing surface. The second solution devised is *rendering-based* [22], making proficient use of the rendering speed of modern GPU's: it allows to locate regions with incomplete sampling (holes) by rendering the intermediate reconstructed model from different positions and choosing the ones which exhibit the larger unsampled surface portions. The camera rendering parameters are set compatible with the real pose parameters of the scanner, and images are rendered using different colors for the background and the front and back side of the surface. Shading is disabled, so that the rendered image contains only one color for each category and the presence of holes can be easily inferred by the presence of non-front-surface color codes.

VI. Interactive visual presentation of huge models

Some issues arise from the impressive increase in data complexity (and richness) provided by the evolution of 3D scanning technology: how to manage/visualize those data on commodity computers; how to improve the ease of use of the visualization tools (as potential users are often not expert with interactive graphics); how to support the presentation of other multimedia information together with the visualization of complex 3D geometry. Our *V*irtual Inspector browser has been designed to give a solution to these issues.

A. *Showing the Arrigo's complex with* V*irtual Inspector*

*V*irtual Inspector is a new visualization system that allows naive users to inspect a large complex 3D model at interactive frame rates on standard PC's. This system evolved considerably from the preliminary version presented in [23]; we describe here briefly its new features. To support the efficient manipulation of massive models, *V*irtual Inspector adopts now a multiresolution approach where view-dependent variable resolution representations are extracted on the fly using a new and highly efficient approach [12]. For each frame, the best-fit *variable resolution* LOD is selected according to the current view frustum and the requested visualization accuracy. LOD selection and rendering are very efficient since we adopt a patch-based representation, where a coarse-grain multiresolution hierarchy is visited on the fly and ready-to-render geometry patches are associated to each logical node of the variable LOD produced. 3D data are therefore not reconstructed on the fly, but efficiently fetched from disk on demand and copied on GPU memory for maximal rendering efficiency.

*V*irtual Inspector is mainly oriented to the visualization of single works of art (sculptures, pottery, architectures, etc.), and adopts a very intuitive approach to guide the virtual manipulation and inspection of the digital replica, based on a straightforward metaphor: we provide a *dummy* representation of the current inspected model on a side of the screen, which can be rotated on its axe; to select any given view the user has just to point with the mouse the corresponding point on the *dummy* (see Figure 4). *V*irtual Inspector supports interactive modification of the lighting, to simulate in real time the "luce radente" (grazing light) effect that is usually used in real inspection to enhance the visualization of small-scale surface detail.

Other important characteristics of *V*irtual Inspector we want to emphasize here are its flexibility and configurability. All main parameters of the system can be easily specified via XML tags contained in a initialization file, such as: which are the 3D models to be rendered (a single mesh or multiple ones, as it is the case of the Arrigo VII complex), the system layout characteristics (i.e. how the different models will be presented on the screen), the rendering modes (e.g. standard Phong-shaded per-vertex colors or BRDF rendering) and the interaction mode (e.g. model manipulation via the standard virtual trackball, the dummy-based "point and click" interaction, or both).

The design of the Arrigo VII installation has been done with the help of a professional graphic designer. Consequently, the layout of the application, all icons and background graphics elements have been completely redesigned with respect to previous incarnations of the *V*irtual Inspector system. This has been done by the easy specification of the new images and location on the screen of all icons and elements of the GUI in the XML initialization file and did not required neither programming nor recompilations of *V*irtual Inspector. It is a task that can be easily assigned to an operator with very limited IT competence.

Finally, we introduced support for *hot-spots*. Hot spots are a very handy resource to associate multimedia data (e.g. html pages) to any point or region of a 3D model. This allows to design interactive presentations where the 3D model is also a natural visual index to historical/artistic information, presented using standard HTML format and browsers (see Figure 5).

The specification of hot spots is extremely easy in *V*irtual Inspector; modifications to the 3D models are not required. We provide a simple 3D browser to the person in charge of the implementation of the multimedia presentation, which allows to query the 3D coordinates of any point on the surface of the artifact (by simply clicking with the mouse on the corresponding point). Then, a new hot spot is specified by introducing a new XML tag in the *V*irtual Inspector specification file. The hot spot XML tag specifies basically the 3D location and the action that has to be triggered when clicking on the hot spot (e.g. the name of the html file, if we want to open a multimedia page). After activation, the control passes to the html browser, while *V*irtual Inspector remains sleeping in the background and regains automatically the control of the interaction whenever the html browser is closed.

The Arrigo VII visual presentation in the museum has been designed with introductory HTML pages, both to present some general artistic/historic information on the Arrigo VII complex (a group of 14 statues which in the XIV cent. were part of the funerary monument of the German Emperor), and to provide links to activate *V*irtual Inspector on the different statues (see

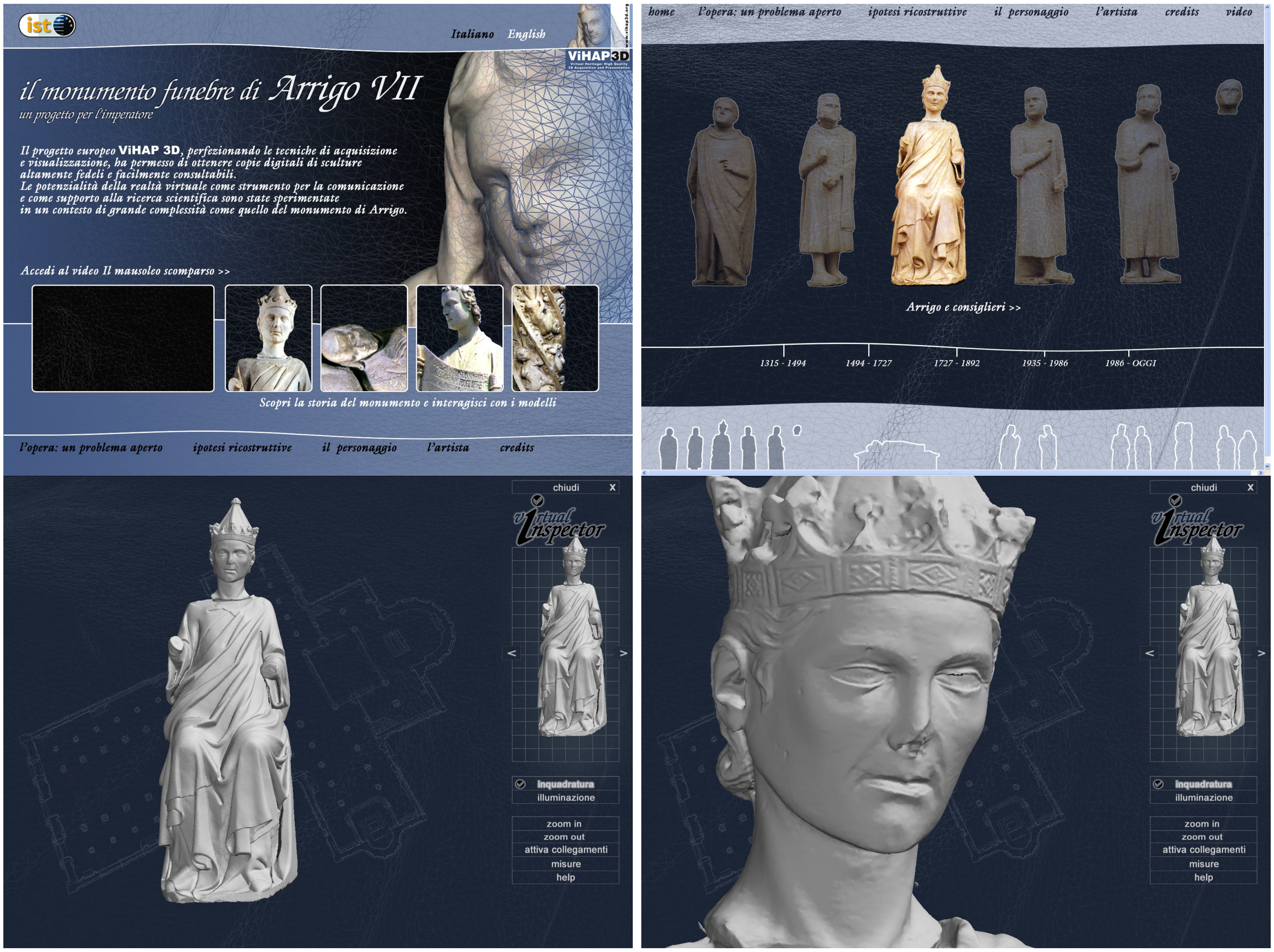

Il progetto europeo ViHAP 3D, perfezionando le tecniche di acquisizione e visualizzazione, ha permesso di ottenere copie digitali di sculture altamente fedeli e facilmente consultabili.
Le potenzialità della realtà virtuale come strumento per la comunicazione e come supporto alla ricerca scientifica sono state sperimentate in un contesto di grande complessità come quello del monumento di Arrigo.

Fig. 4. The initial screen of the Arrigos VII's multimedia kiosk and one of the following sub-index pages are shown above; to provide access to any statue of the Arrigo VII complex, the statues have been divided in four groups (the second image shows the index page related to the "Arrigo VII enthroned" and counsellors group). *V*irtual Inspector can be started by clicking on any of the icons of the statues here presented (image top-right). Visualizations of the 3D model of Arrigo VII enthroned are shown below, with two screen shots of the interactive inspection.

Figure 4).

VII. Conclusions

3D scanning can be considered as a nearly mature technology. The research performed in the last few years has produced significant results, but some issues still remain open. We have presented some recent results on two different sides: how to increase the automation of the scanning process (which, unfortunately, is still user-assisted if we want to produce a good-quality model); and how to manage efficient rendering of very large models, supporting also the integration of multi-media data to the 3D mesh with the classical hyperlink approach.

One issue on which we are now focusing is how to manage high-resolution sampling of color data (i.e. hundreds of high-res digital images) on a high-resolution 3D model. Especially in the field of restoration of Cultural Heritage, curators need to manage huge photo sampling, and how to map and render at interactive rates all those data is still a serious problem.

Acknowledgements We acknowledge the financial support of the project EU IST-2001-32641 "ViHAP3D".

References

[1] F. Bernardini and H. E. Rushmeier, "The 3D Model Acquisition Pipeline," *Computer Graphics Forum*, vol. 21, no. 2, pp. 149–172, March 2002.

[2] B. Curless and S. Seitz, "3D Photography," in *ACM SIGGRAPH 00 Course Notes, Course No. 19*, 2000.

[3] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, "The Digital Michelangelo Project: 3D scanning of large statues," in *SIGGRAPH 2000, Computer Graphics Proceedings*, ser. Annual Conference Series. Addison Wesley, July 24-28 2000, pp. 131–144.

[4] F. Bernardini, H. E. Rushmeier, I. Martin, J. Mittleman, and G. Taubin, "Building a Digital Model of Michelangelo's Florentine Pieta'," *IEEE Comp. Graphics & Applications*, vol. 22, no. 1, pp. 59–67, Jan-Febr. 2002.

[5] R. Fontana, M. Greco, M. Materazzi, E. Pampaloni, L. Pezzati, C. Rocchini, and R. Scopigno, "Three-dimensional modelling of statues: the minerva of arezzo," *Journal of Cultural Heritage*, vol. 3, no. 4, pp. 325–331, 2002.

[6] M. Pollefeys, L. J. V. Gool, M. Vergauwen, F. Verbiest, and J. Tops, "Image-based 3d acquisition of archeological heritage and applications," in *VAST 2001 Conference Proc.*, D. Arnold, A. Chalmers, and D. Fellner, Eds. Athens, Greece: ACM Siggraph, Nov. 28-30 2001, pp. 255–261.

[7] J. Stumpfel, C. Tchou, T. Hawkins, P. Debevec, J. Cohen, A. Jones, and B. Emerson, "Assembling the sculptures of the parthenon," in *VAST 2003*, A. C. D. Arnold and F. Niccolucci, Eds. Bighton, UK: Eurographics, Nov. 5-7 2003, pp. 41–50.

[8] C. Baracchini, A. Brogi, M. Callieri, L. Capitani, P. Cignoni, A. Fasano, C. Montani, C. Nenci, R. P. Novello, P. Pingi, F. Ponchio, and R. Scopigno, "Digital reconstruction of the arrigo vii funerary complex," in *VAST 2004*, N. S. Y. Chrysanthou, K. Cain and F. Niccolucci, Eds. Bruxelles, BE: Eurographics, Dec. 7-10 2004, pp. 145–154.

[9] M. Garland and P. Heckbert, "Surface simplification using quadric error metrics," in *SIGGRAPH 97 Conference Proceedings*, ser. Annual Conference Series. Addison Wesley, Aug. 1997, pp. 209–216. [Online]. Available: http://www.cs.cmu.edu/afs/cs/user/garland/www/quadrics/quadrics.html

[10] H. Hoppe, "New quadric metric for simplifying meshes with appearance attributes," in *Proc. IEEE Conference on Visualization (VIS99)*. New York: ACM Press, Oct. 25–28 1999, pp. 59–66. [Online]. Available: http://research.microsoft.com/∼hoppe/

[11] P. Cignoni, C. Montani, C. Rocchini, and R. Scopigno, "External memory management and simplification of huge meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 4, pp. 525–537, 2003.

[12] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, "Adaptive tetrapuzzles: Efficient out-of-core construction and visualization of gigantic multiresolution polygonal models," *ACM Trans. on Graphics (SIGGRAPH 2004)*, vol. 23, no. 3, pp. 796–803, 2004.

[13] S. Rusinkiewicz and M. Levoy, "QSplat: A multiresolution point rendering system for large meshes," in *Comp. Graph. Proc., Annual Conf. Series (SIGGRAPH 00)*. ACM Press, July 24-28 2000, pp. 343–352.

[14] M. Botsch, A. Wiratanaya, and L. Kobbelt, "Efficient high quality rendering of point sampled geometry," in *Proceedings of the 13th Eurographics Workshop on Rendering (RENDERING TECHNIQUES-02)*, S. Gibson and P. Debevec, Eds. Aire-la-Ville, Switzerland: Eurographics Association, June 26–28 2002, pp. 53–64.

[15] M. Callieri, P. Cignoni, F. Ganovelli, C. Montani, P. Pingi, and R. Scopigno, "Vclab's tools for 3d range data processing," in *VAST 2003*, A. C. D. Arnold and F. Niccolucci, Eds. Bighton, UK: Eurographics, Nov. 5-7 2003, pp. 13–22.

[16] K. Pulli, "Multiview registration for large datasets," in *Proc 2nd Int.l Conf. on 3D Digital Imaging and Modeling*. IEEE, 1999, pp. 160–168.

[17] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Comp. Graph. Proc., Annual Conf. Series (SIGGRAPH 96)*. ACM Press, 1996, pp. 303–312.

[18] M. Callieri, P. Cignoni, and R. Scopigno, "Reconstructing textured meshes from multiple range rgb maps," in *7th Int.l Fall Workshop on Vision, Modeling, and Visualization 2002*. Erlangen (D): IOS Press, Nov. 20 - 22 2002, pp. 419–426.

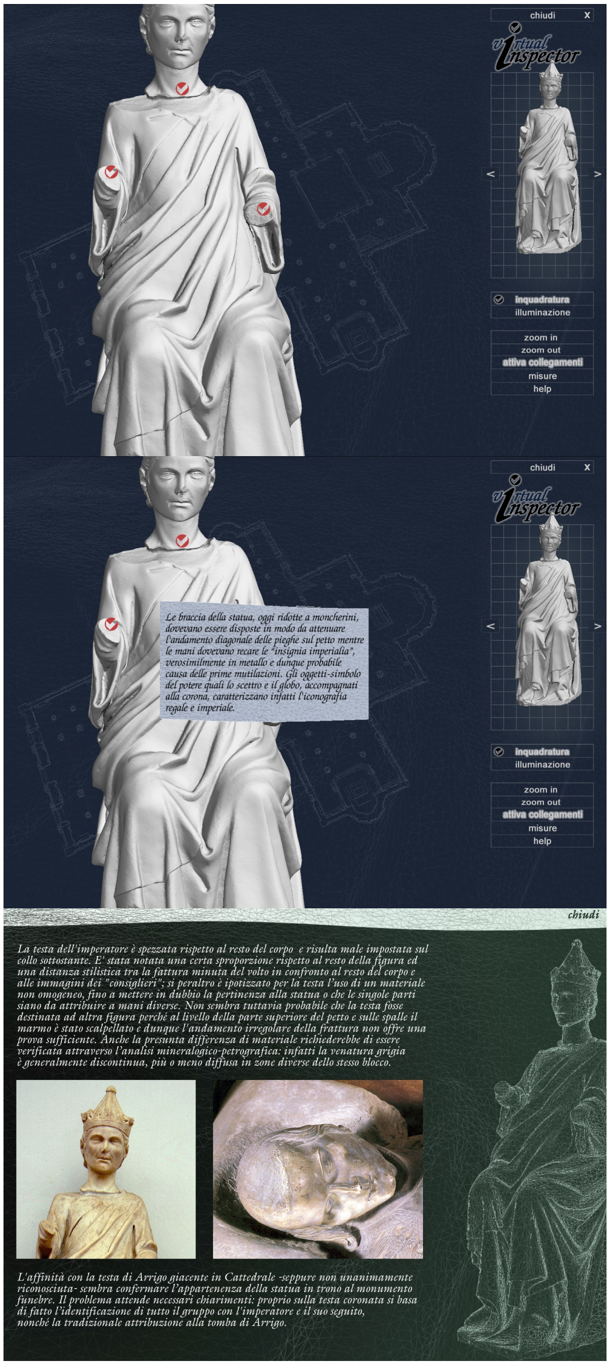
[19] R. J. Campbell and P. J. Flynn, "A Survey Of Free-Form Object Representation and Recognition Techniques," *Computer Vision and Image Understanding*, vol. 81, no. 2, pp. 166–210, 2001.

[20] A. Fasano, P. Pingi, P. Cignoni, C. Montani, and R. Scopigno, "Automatic registration of range maps," ISTI – C.N.R., Pisa, Italy, Tech. Rep. ISTI-2004-TR (Submitted paper), Sept. 2004.

[21] P. Cignoni, G. Impoco, and R. Scopigno, "Closing gaps by clustering unseen directions," in *Shape Modeling International (SMI) 2004 Conf. Proc.*, Genova, Italy, June 23-26 2004, pp. 307–317.

[22] M. Callieri, A. Fasano, G. Impoco, P. Cignoni, R. Scopigno, G. Parrini, and G. Biagini, "Roboscan: An automatic system for accurate and unattended 3d scanning," in *3DPVT'04: Second Int. Symp. on 3D Data Processing, Visualization and Transmission*. IEEE Comp. Soc., Sept. 6-9 2004, p. (in press).

[23] R. Borgo, P. Cignoni, and R. Scopigno, "An easy to use visualization system for huge cultural heritage meshes," in *VAST 2001 Conference Proc.*, D. Arnold, A. Chalmers, and D. Fellner, Eds. Athens, Greece: ACM Siggraph, Nov. 28-30 2001, pp. 121–130.



Le braccia della statua, oggi ridotte a moncherini, dovevano essere disposte in modo da attenuare l'andamento diagonale delle pieghe sul petto mentre le mani dovevano recare le "insignia imperialia", verosimilmente in metallo e dunque probabile causa delle prime mutilazioni. Gli oggetti-simbolo del potere quali lo scettro e il globo, accompagnati alla corona, caratterizzano infatti l'iconografia regale e imperiale.

La testa dell'imperatore è spezzata rispetto al resto del corpo e risulta male impostata sul collo sottostante. E' stata notata una certa sproporzione rispetto al resto della figura ed una distanza stilistica tra la fattura minuta del volto in confronto al resto del corpo e alle immagini dei "consiglieri"; si peraltro è ipotizzato per la testa l'uso di un materiale non omogeneo, fino a mettere in dubbio la pertinenza alla statua o che le singole parti siano da attribuire a mani diverse. Non sembra tuttavia probabile che la testa fosse destinata ad altra figura perché al livello della parte superiore del petto e sulle spalle il marmo è stato scalpellato e dunque l'andamento irregolare della frattura non offre una prova sufficiente. Anche la presunta differenza di materiale richiederebbe di essere verificata attraverso l'analisi mineralogico-petrografica: infatti la venatura grigia è generalmente discontinua, più o meno diffusa in zone diverse dello stesso blocco.

L'affinità con la testa di Arrigo giacente in Cattedrale -seppure non unanimamente riconosciuta- sembra confermare l'appartenenza della statua in trono al monumento funebre. Il problema attende necessari chiarimenti: proprio sulla testa coronata si basa di fatto l'identificazione di tutto il gruppo con l'imperatore e il suo seguito, nonché la tradizionale attribuzione alla tomba di Arrigo.

Fig. 5. *V*irtual Inspector: the "Arrigo VII enthroned" statue rendered with active hot spots (top); a short popup panel with a short info, describing the missing hand, appears when the mouse passes over the hotspot (middle); an example of an HTML page activated by clicking the hot spot on the neck (bottom).