

Virtual Archaeologist: Assembling the past

Georgios Papaioannou *IEEE Member*,
Evaggelia-Aggeliki Karabassi *IEEE Member*,
Theoharis Theoharis
University of Athens

Abstract This paper describes a semi-automatic system for the reconstruction of archaeological finds from their fragments. Virtual Archaeologist is a system that uses computer graphics to calculate a measure of complementary matching between scanned data and employs optimization algorithms in order to estimate the correct relative pose between fragments and cluster those fragments that belong to the same entity.

Keywords: 3D matching, depth buffer, restoration, global optimization

Reconstruction of archaeological monuments from fragments or parts found at the archaeological sites, is a tedious task requiring many hours of work from the archaeologists and restoration personnel.

In case of large constructions, such as the Parthenon at the Acropolis of Athens, the restoration process takes considerably longer due to the large mass of the fragments. In order to test large building blocks against others for potential matching, archaeologists and site architects have to move the cumbersome stones sometimes 50 meters or more away from their original locations using cranes. Archaeological reconstruction is further hindered by missing or deteriorated fragments due to erosion or impact damage.

Up to now, computers have aided archaeologists by providing tools for the digitization and archiving of artifacts¹, visualization and virtual manipulation of 3D or 2D scanned objects, visual representation of historical sites through VR², image processing and restoration of frescos³.

However, not much work has been conducted toward the automatic reconstruction of complete objects from arbitrary fragments. Existing algorithms focus on the reconstruction of vases and rely either on classification of certain qualitative features of the fragments, as in⁴, or comparison of the broken surface boundary curves to match and align the vase pieces⁵. The

first method assumes that the structure of the final, complete object is known a priori and fragments have to be extensively labeled and categorized beforehand. The second completely disregards the interior of the broken surface and is therefore restricted to thin-walled objects.

The Digital Michelangelo team¹ is currently investigating approaches to assemble the Forma Urbis Romae, a marble map of ancient Rome, from 1,163 fragments. The team is planning to face the problem as a jigsaw puzzle based on broken surface border signatures, while exploiting additional features of the fragments, such as thickness or marble veining.

In the general case, the reconstruction of arbitrary objects from their fragments can be regarded as a 3D puzzle, taking into account the following considerations:

- Parts (fragments) have arbitrary shape.
- The shape and number of the final objects is unknown.
- We have an arbitrary number of fractured faces per fragment.
- Some fragments may be missing.
- Surfaces are probably flawed or weathered.
- There are no strict assemblage rules.

In this article we present a complete method, encapsulated in our Virtual Archaeologist system, for the full reconstruction of



Archaeological reconstruction involves tedious and expensive efforts due to the large number of fragments (more than 10,000 fragments have been collected only from the Acropolis site) and often their considerable size. Until now reconstruction is manual and the restoration specialist has to practically rely on his/her memory and intuition to discover the relation between various fragments. A tool that allows the quick browsing of the collection and the previewing of possible assemblage configurations is bound to significantly facilitate our work and speed up the whole process.

Nikolaos Toganidis
Chief Architect
Parthenon Restoration Project

archaeological finds from 3D scanned fragments. Virtual Archaeologist has been designed to assist archaeologists in the reconstruction of monuments or smaller finds by providing the means to avoid unnecessary manual experimentation with fragile and often heavy fragments. An automated procedure can not completely replace the archaeology expert but provides a useful estimation of valid fragment combinations, as well as a tool for accurately measuring the matching between fragments.

In Virtual Archaeologist, we regard the reconstruction problem from a general, geometric point of view that relies on the broken surface morphology to determine correct matches between fragments. This approach is

abstracted from specific object information but is versatile enough to exploit any other data available. A brief preliminary sketch of the underlying algorithms used in Virtual Archaeologist has been presented in ⁶.

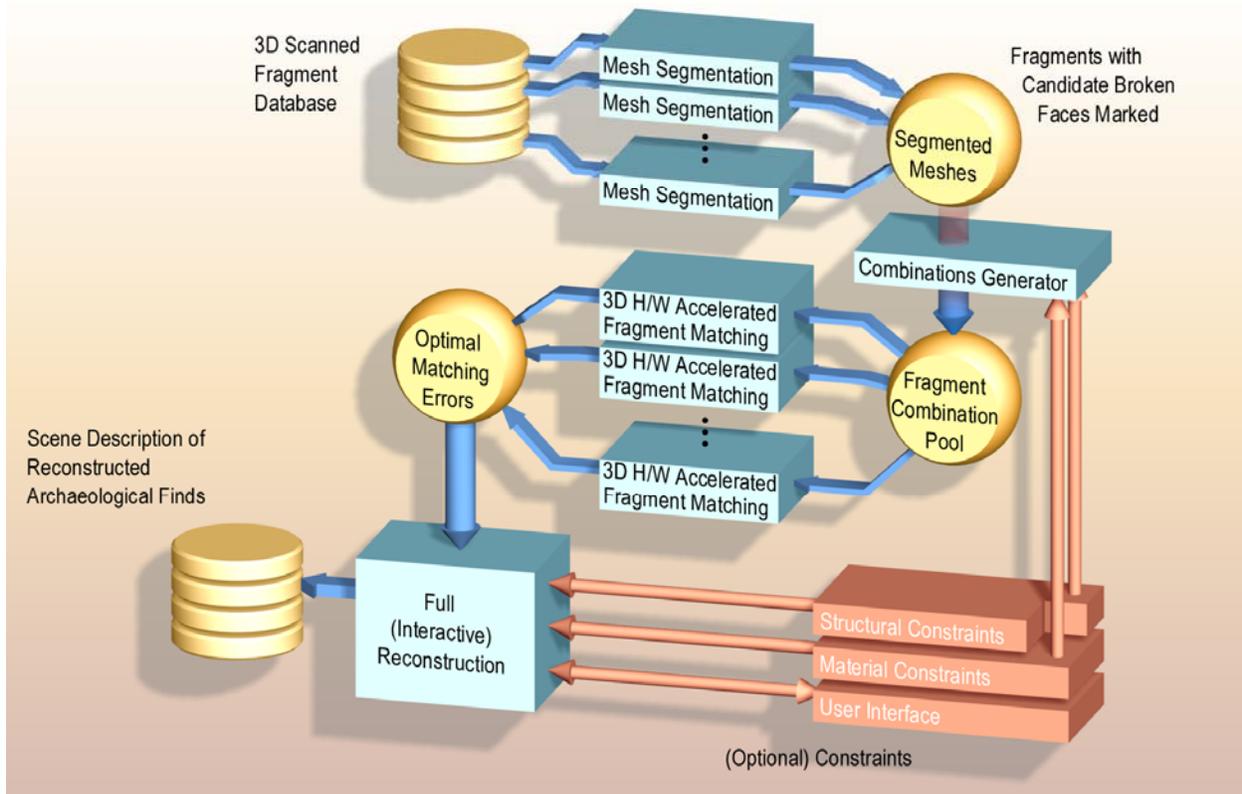
The procedure involves the automatic detection of candidate fractured faces, the one-by-one fragment matching and the final assemblage (clustering) of the fragments into complete or partially complete entities. The only input data required by our system are the polygonal meshes of the original fragments. These meshes are commonly acquired with a 3D scanner or digitizer though modeling and curve interpolation may be required in cases where only blueprints of cut-sections of the fragments are available as part of the standard archiving procedure. An optional set of constraints, such as material or structural fragment attributes which are often available, significantly improve the overall accuracy and performance. Human intervention is not required but can clarify the final reconstruction result by interactively fine-tuning the final clustering and pose of the fragments.

Method Overview

The reconstruction is divided into three main stages (Figure 1). The first stage is the *mesh segmentation*. In order to minimize the search space when trying to “glue” two fragments together, we wish to restrict the matching between potentially “interesting” sides of the fragments. Based on the observation that fractured surfaces, even weathered ones, tend to be more rough and jagged, we segment each mesh into *facets* and only mark as candidates for matching those facets that exhibit relatively high coarseness.

In the second stage (*fragment matching*), after having generated all valid fragment-pair combinations, we estimate the relative pose for all fragment pairs and all candidate facets at which a *matching error* is minimized. Each pair of fragments is examined in order to determine the relative orientation that corresponds to the best fit and therefore the minimal matching error per facet pair. Note that not all possible pairs of fragments enter this process, as many are discarded due to incompatible material or target structure, if such information is available

Virtual Archaeologist



1 Virtual Archaeologist architecture and data flow.

beforehand (see optional constraints in Figure 1).

The matching error estimator utilizes hardware accelerated 3D rendering of the fragments and operates on the depth buffers produced. In our application, rendering is not used just for visualization, but actively participates in the matching process. We present more details on this matter later on.

When all optimal pairwise (fragment facet by fragment facet) matching error values have been calculated and stored in a look-up table, the third stage (full reconstruction) selects those fragment combinations that minimize a *global reconstruction error*. This reconstruction error equals the sum of matching errors of a given set of fragment pairs.

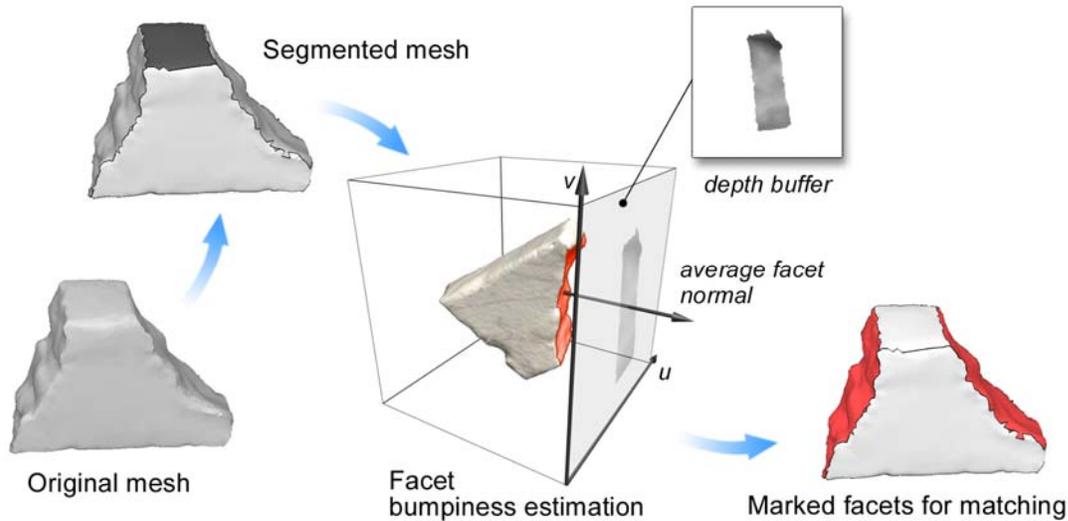
External constraints may contribute to this stage as well, in order to reduce the time needed to produce a correct fragment clustering by

eliminating a large number of combinations.

Both the mesh segmentation and fragment matching stages are off-line operations. Each time we add a new fragment mesh to the object database, it is segmented and the matching error values with respect to all other fragments are calculated and stored. An important reason why these procedures are not interactive is that although the segmentation of a mesh lasts no more than 2 seconds, fragment matching can take up to 30 seconds per facet pair. We should mention at this point that all times in this paper were measured on a Pentium III/450MHz equipped with a TNT2 Ultra/185MHz graphics accelerator.

Mesh Segmentation

A fragment mesh is first partitioned into areas of adjacent nearly coplanar polygons, corresponding to the *facets* of the object (Figure



2 Mesh segmentation and detection of broken sides.

2). This surface segmentation process is accomplished, using a simple region-growing algorithm⁷. The process begins with an arbitrary polygon. Neighboring polygons are classified to the same region if their normals do not deviate from the average region normal by more than a predefined threshold, otherwise a new region is formed.

During the region growing process, small surface regions may be created within larger ones. As it is desirable to partition the mesh into “crude” facets, we apply a region merging stage to clean up the facets and eliminate small erroneous regions. A region is regarded as “small” if it covers less than 5% of the entire mesh surface area. The elimination of the insignificant regions is achieved by iteratively assigning the polygons of small surface areas to large adjacent regions.

Having partitioned the fragment mesh into crude facets, we proceed by labeling as potential for matching those facets that exhibit higher coarseness. A facet’s bumpiness can not be accurately measured directly from the original mesh (unless the surface is uniformly sampled) because each facet consists of polygons of arbitrary connectivity and varying area. Instead, we use an image-based bumpiness measure calculated on the *elevation map* of the facet.

The elevation map is essentially a two-dimensional array, representing the distance of

the facet from a plane perpendicular to the average facet normal, measured at equidistant grid points. This map is easy to obtain as it equals the contents of the depth-buffer, when the facet triangles are rendered with the viewing direction parallel to the average facet normal (Figure 2).

The bumpiness of a surface is associated with the rate of elevation variance and can be effectively estimated on the elevation map with an image filter, such as the Laplace image operator.

Obviously, engraved sides of a fragment are inherently bumpy and are therefore marked as well. However, this is not a problem as these facets are incompatible with any other and will therefore produce a high matching error during the next stage. On the contrary, if the broken sides are very smooth, they may not be automatically marked and manual selection is required.

Fragment Matching

In order to examine if two fragments are complementary, we seek the best match between them with respect to their relative pose and calculate a corresponding matching error. This process is repeated for all marked facet pairs of all fragment combinations.

The two fragments are positioned in a way that two of their broken facets are facing each

other (Figure 3), i.e. the average facet normals are aligned. A set of seven pose parameters is adequate for the alignment of the two fragments. More specifically, the first object is allowed to perform a full circle around the axis of alignment (ρ_1), deviate from this axis (ϕ_1, θ_1) by up to 10° and slide along the broken facet (x_1, y_1). The ability to slide is essential in locating potential partial matches between facets of different size or shape. The second object need only diverge from the axis of alignment by up to 10° (ϕ_2, θ_2). Although the last rotations are theoretically redundant, they improve the convergence of the search for the optimal matching.

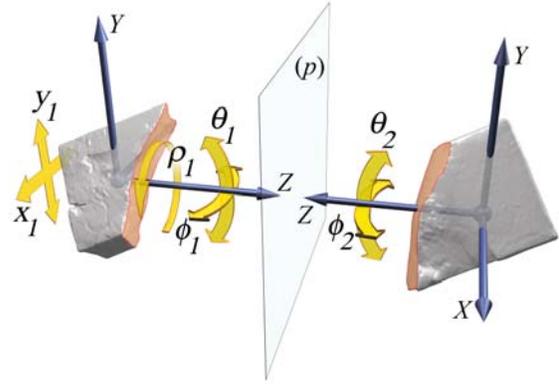
Ideally, if the broken surfaces of the fragments are complementary, the matching error should be zero for a relative pose of the two pieces where they “fit” together. For all other placement configurations or for incompatible fragments, there is a significant matching error between the fragment facets.

A naïve way to estimate the matching error for each set of pose parameters would be to sum all point-to-point distances between corresponding points on the facing surfaces of the two fragments. Unfortunately, due to its dependence on the closest distance between the two fragments, this solution is very sensitive to noisy data. Worse, small variations of the pose parameters produce drastic changes to the resulting error, making this measure unreliable and difficult to optimize.

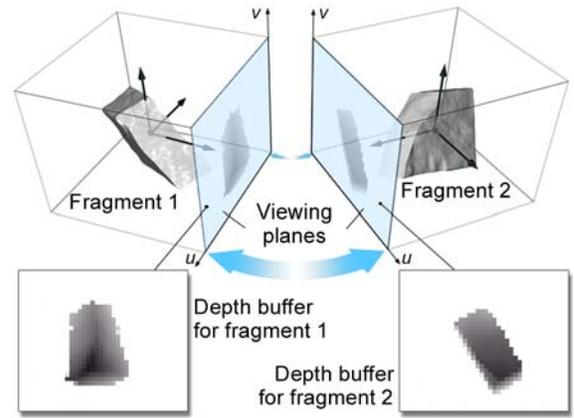
Instead of using the point-to-point distances directly, we work on their derivatives, i.e. the curvature of the surfaces as measured from a uniform grid on a plane (p). (p) is positioned between the fragments and is perpendicular to the original average normal axes (Figure 3). The resulting matching error ε_d for two facets is:

$$\varepsilon_d = \frac{1}{A_s} \iint_S \left(\left| \frac{\partial d_1(u,v)}{\partial u} + \frac{\partial d_2(u,v)}{\partial u} \right| + \left| \frac{\partial d_1(u,v)}{\partial v} + \frac{\partial d_2(u,v)}{\partial v} \right| \right) dS$$

where d_1 and d_2 are the distances of the two surfaces from the separating plane (p), S is the region of overlap between the surface



3 Fragment pose parameters for the matching error calculation.

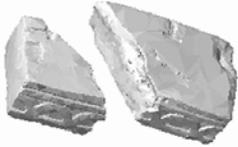


4 Point-to-point distance calculation with the depth buffer.

projections on (p) and A_s is the corresponding area of overlap.

Small isolated surface flaws or sampling errors have a local effect on the resulting error because of its differential form, thus rendering the method tolerant to noisy input data.

In practice, the matching error is measured using the depth buffer. Imagine an observer looking at the two fragments through a viewing plane coincident with the separating plane (p), as illustrated in figure 4. Rendering separately each fragment, we obtain two depth buffers $Z_1(i, j)$ and $Z_2(i, j)$, $i = 1, \dots, N_u$, $j = 1, \dots, N_v$. Surface curvature for corresponding points on the fragments is uniformly sampled and easily

 $\langle e_d \rangle = 0.0019$ 89%	 $\langle e_d \rangle = 0.0024$ 72%	 $\langle e_d \rangle = 0.0031$ 89%
 $\langle e_d \rangle = 0.0020$ 7%	 $\langle e_d \rangle = 0.0056$ 14%	 $\langle e_d \rangle = 0.0051$ 11%

5 Fragment matching examples for scanned objects. The top row presents correct matches along with the corresponding error measures and success rates. The bottom row shows the most frequent erroneous results.

obtained from the derivatives of these two buffers with respect to u and v . The partial derivatives of the continuous error measure are replaced by forward differences on the depth buffers:

$$\Delta_u Z(i, j) = Z(i+1, j) - Z(i, j)$$

$$\Delta_v Z(i, j) = Z(i, j+1) - Z(i, j)$$

The matching error is evaluated as:

$$\varepsilon_d = \frac{1}{A_S} \sum_{(i,j) \in S} (|\Delta_u Z_1(i, j) + \Delta_u Z_2(i, j)| + |\Delta_v Z_1(i, j) + \Delta_v Z_2(i, j)|)$$

In the above equation, S is the set of depth buffer cells where both depth buffers have non-infinite values and A_S is the number of elements in S .

The error is minimized over the set of pose parameters using a global optimization method. In our system, we implemented Enhanced SA, a variation of Simulated Annealing (SA)⁸, which produces good results (average optimum pose detection rate 85%). The details of SA are beyond the scope of this article but the interested reader can refer to any combinatorial optimization or search methods book for details on SA, as well as to⁹ for the Enhanced SA method we adopted for our application. Some representative examples are shown in Figure 5.

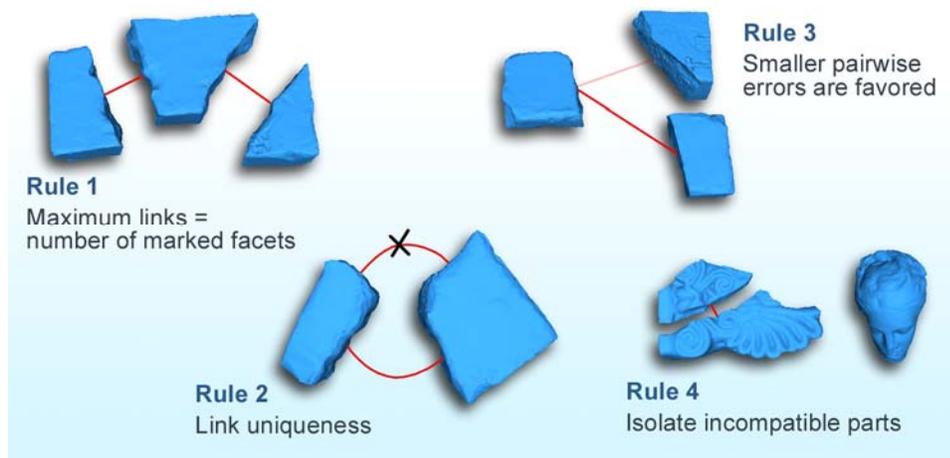
The Full Reconstruction

For the final assemblage of the fragments, we seek to minimize a global *reconstruction error*, which is the sum of the matching errors of the individual combinations currently active. The fragment assemblage is governed by 4 principal rules (Figure 6):

1. A fragment can be linked to as many other objects as the number of its facets marked as broken.
2. The bond between two fragments is unique,
3. Fragment pairs that yield a smaller matching error must be favored.
4. Fragments may exist that do not belong to a valid reconstructed object and must be isolated.

A link between two fragments corresponds to a combination of fragment facets whose matching error and relative pose have been precalculated. The globally optimized assemblage of the fragments is formed by a kernel that generates and rearranges the links in a sub-second time. This is an important feature because the user is able to experiment with the fragments and shape the final result interactively by explicitly joining or separating them.

If the number of fragments in the data set is fairly small (fewer than 30 pieces), the set of fragment combinations that yields the smallest reconstruction error is determined using exhaustive search. As exhaustive search leads to



6 Full reconstruction stage assemblage rules.

exponential increase of execution time, with respect to the number of fragments, for large data sets we use a genetic algorithm to reduce the number of iterations needed (Figure 7).

In the case of the genetic algorithm, the currently active set of combinations is repeatedly mutated by crossing-over combinations. The new set of combinations that is produced may be accepted or rejected according to the new reconstruction error. The algorithm terminates when a set of combinations is fit enough (has prevailed long enough). More details on the specific algorithm may be found in 6.

System Implementation

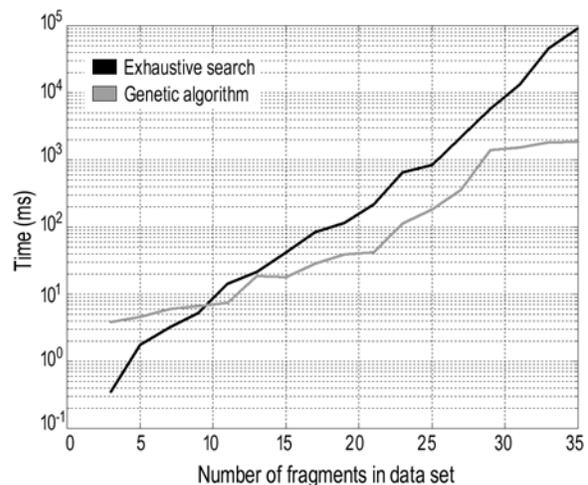
Virtual Archaeologist is implemented in three modules, one for each stage, so that they can work independently.

We perform the preprocessing once, after the polygonal data acquisition and the segmentation information is stored in the mesh file. We used OpenGL for the calculation of the facet elevation maps and the segmentation lasts no more than 2 seconds even for complex models (~90,000 triangles) for 128×128 depth buffer resolution.

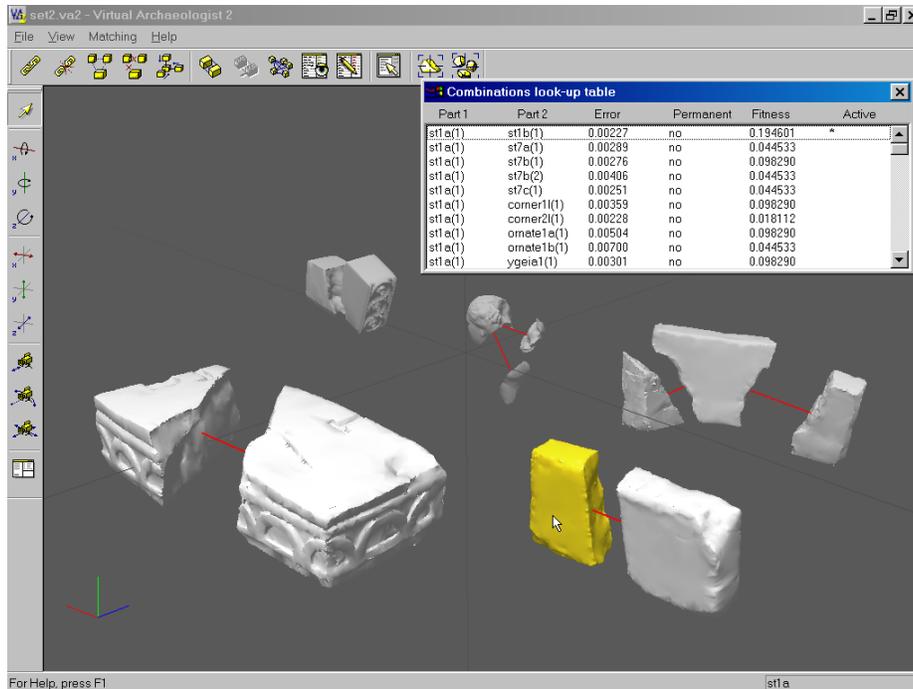
The fragment matching module uses hardware accelerated OpenGL rendering for the error estimation achieving an average of 10 matching error samples per second. Depth buffer resolution can be adjusted to trade off accuracy for computation speed, according to the graphics hardware rendering throughput. Typical

dimensions for the depth buffer in this application are between 100×100 and 256×256 pixels. Matching lasts between 15 and 40 seconds per facet pair, depending on the mesh triangle count. As fragment matching is computationally expensive, it is invoked incrementally to update the file of matching errors when a new piece is added to the collection.

The interactive full reconstruction is performed in a 3D environment as depicted in figure 8. This allows the user to edit the final result by providing standard manipulation functionality (fragment position and rotation), as well as additional tools via the user interface. These include the linking and separation of



7 Full reconstruction stage execution time.



8 The virtual Archaeologist desktop in action.

fragments, the real-time matching error measurement between two fragments and the automatic pose calculation for the reconstructed objects.

Case Studies

The use of range scanners has been only recently introduced to field archaeology; archaeological sites, such as the Athens Acropolis, do not yet have the necessary equipment to create databases of scanned finds. We have tested Virtual Archaeologist with 3D-digitized plaster scale models of objects, mostly building block replicas, and ceramic pot fragments.

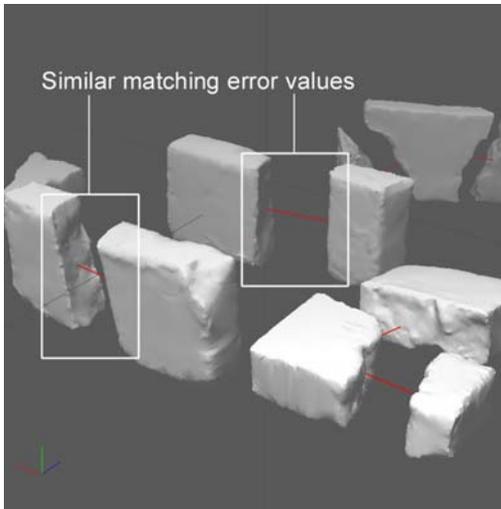
The majority of the fragments were not well preserved and the broken surfaces were smoothed out. Even on sharp edged pieces, protrusions have been deliberately chipped off during the experiments, to simulate deterioration. In general, we had no fragments that could match perfectly and this is a fact that is evident in all archaeological finds. The fragment collections we have used, ranged from 3 to 35 pieces.

Unsupervised reconstruction (no user intervention) resulted in correct reconstruction

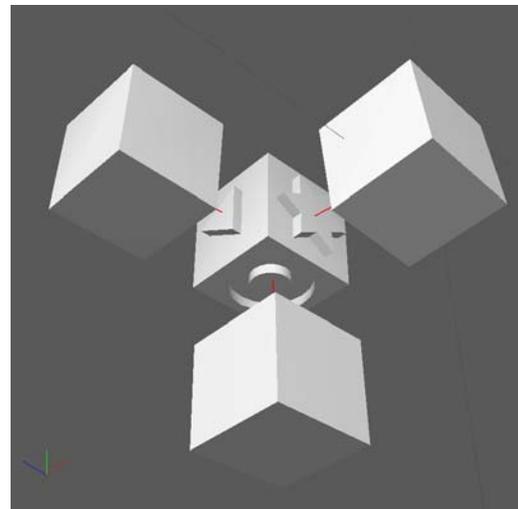
for 90% of the original objects. We used a simple, yet effective constraint in the final reconstruction stage, the broken surface area similarity. If the area of the broken facets differs by more than 20%, the system prohibits this combination. The existence and the parameters of this constraint can be adjusted from the user interface.

In experiments where many surfaces were similar to each other and the matching error variations small, some fragments failed to match. Practically, it was sometimes difficult even for a human to match these fragments (Figure 9). Moreover, as the automatic assembling algorithm does not have knowledge about the expected shape of the reconstructed object, in some cases the fragment combinations that minimized the matching error did not correspond to valid objects. Manual linking of the fragments resolved the above ambiguities, during the interactive reconstruction stage.

To demonstrate our system's ability to perform 3D assemblage based only on geometric information, without any constraints, we include the small 3D-puzzle example of figure 10. The pieces of this puzzle were produced with a modeling package and Virtual Archaeologist



9 Reconstruction example with possible ambiguities.



10 Simple 3D puzzle: a non archaeological case study.

matched them perfectly.

Acknowledgements

This work was partially funded by the National and Kapodistrian University of Athens, Research Grant #70/4/3241.

We would like to thank Nikolaos Toganidis (Chief Architect, Parthenon Restoration Project) for motivating this work.

References

1. M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, D. Fulk, "The Digital Michelangelo Project: 3D Scanning of Large Statues", Computer Graphics (Proc. SIGGRAPH 2000), New Orleans, USA, July 2000, pp. 131 – 144.
2. E. Berndt, J. C. Teixeira, "Cultural Heritage in the Mature Era of Computer Graphics", Computer Graphics and Applications, 20(1), 2000.
3. A.D. Kalvin, A. Remy, L.J. Castillo, K. Morla, E. Nolasco, J. Prado, V. Fernandez, R. Franco, and G. Wiese, "Computer-Aided Reconstruction of a pre-Inca Temple Ceiling in Peru," Proc. Computer Applications in Archaeology (CAA 97), University of Birmingham, England, April 1997.
4. R. Sablatnig, C. Menard, and W. Kropatsch, "Classification of Archaeological Fragments

Using a Description Language," Proc. Eusipco 98, vol 2, Rhodes, Greece, 1998, pp. 1097-1100.

5. G. Ucoluk, I. H. Toroslu, "Automatic Reconstruction of Broken 3-D Surface Objects", Computers and Graphics, Vol. 23, No. 4, August 1999, pp. 573-582.
6. G. Papaioannou, E. A. Karabassi, T. Theoharis, "Automatic Reconstruction of Archaeological Finds – A Graphics Approach", Proc. International Conference on Computer Graphics and Artificial Intelligence '2000, Limoges, France, 2000, pp. 117-125.
7. G. Papaioannou, E. A. Karabassi, T. Theoharis, "Segmentation and Surface Characterization of Arbitrary 3D Meshes for Object Reconstruction and Recognition", Proceedings of the International Conference on Pattern Recognition '2000, IEEE, 2000, pp. 734-737.
8. S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi. "Optimization by Simulated Annealing". Science, 220(4598), pp. 671-680, 1983.
9. P. Siarry, G. Berthiau, F. Durbin, J. Haussy, "Enhanced Simulated Annealing for Globally Minimizing Functions of Many-Continuous Variables", ACM Transactions on Mathematical Software, 23(2), pp 209-228, 1997.