



Predictive digitization, restoration and degradation assessment
of cultural heritage objects



D5.4 – First Evaluation Report

| | |
|---|---|
| Grant Agreement Number | 600533 |
| Project Acronym | PRESIOUS |
| Project Title | PREdictive digitization, reStoration and degradatIOn assessment of cultUral her-itage objectS |
| Funding Scheme | Collaborative Project (STREP) |
| First Evaluation Report | |
| Name, title and organisation of the scientific representative of the project's coordinator | T. Theoharis, Professor, NTNU |
| Telephone | +47 73591447 |
| Email | theotheo@idi.ntnu.no |
| Project Website Address | www.presious.eu |
| Dissemination Level | PU |

Address
Sem Sælands vei 5, Gamle fysikk 3rd floor
Gløshaugen, NTNU, NO-7491 Trondheim,
Norway

<http://presious.eu>

Contact person
Karelle Gilbert-Soni
karelle@ime.ntnu.no
+ 47 73591481

Deliverable Identification Sheet

| | |
|-------------------------------------|--|
| Grant Agreement number | 600533 |
| Project Acronym | PRESIOUS |
| Project Title | PREdictive digitization, reStoration and degradatIOn assessment of cultUral her-itage objectS |
| Funding Scheme | Collaborative Project (STREP) |
| Contractual Delivery Date | M24 |
| Deliverable Number | D5.4 |
| Deliverable Name | First Evaluation Report |
| Type | Document (L ^A T _E X) |
| Deliverable Version | 2.4 |
| Status | Final |
| Associated WP / Task | WP5 |
| Author(s) | <p>NTNU: Theoharis Theoharis, Christian Schellewald, Panagiotis Perakis</p> <p>AUEB-RC: Anthousis Andreadis, Georgios Papaioannou, Pavlos Mavridis</p> <p>ATHENA RC: Michalis Savelonas, Konstantinos Sfikas, Ioannis Pratikakis, Fotis Arnaoutoglou</p> <p>UKON: Ivan Sipiran, Robert Gregor, Tobias Schreck</p> <p>BREUCKMANN GMBH: Dirk Rieke-Zapp</p> |
| Project Officer | Philippe Gelin |
| Abstract | This document contains the first evaluation report |
| Circulated to Participants | 28. February, 2015 |
| Read by Participants | 3. March, 2015 |
| Approved by General Assembly | 10. March, 2015 |

Document Revision History

| Date | Version | Author/Editor/Contributor | Summary of Changes |
|------------|---------|---|--|
| 09.01.2015 | 1.0 | Theoharis Theoharis | Very first version |
| 28.01.2015 | 1.1 | Ivan Sipiran, Tobias Schreck, Robert Gregor | Circulated structure WP4 |
| 16.02.2015 | 1.5 | Ivan Sipiran, Tobias Schreck, Robert Gregor | Description of experiments T4.2 |
| 17.02.2015 | 1.6 | Anthousis Andreadis, Georgios Papaioannou | Description and evaluation of experiments for T4.1 |
| 18.02.2015 | 1.7 | Ivan Sipiran | Final merging WP4 |
| 19.02.2015 | 1.8 | Theoharis Theoharis | Merging of partner contributions from WP3, WP4, WP5 |
| 26.02.2015 | 2.0 | Michalis Savelonas, Fotis Arnaoutoglou, Konstantinos Sfikas, Ioannis Pratikakis | Description of partial retrieval experiments in WP2 and PDP-related material for WP5 |
| 26.02.2015 | 2.2 | Pavlos Mavridis | Description of model reshaping experiments in WP2 |
| 27.02.2015 | 2.3 | Theoharis Theoharis | Incorporation of entire WP2 and changes in WP3 |
| 28.02.2015 | 2.4 | Christian Schellewald, Panagiotis Perakis | Incorporation of final WP3 |

Contents

| | | |
|----------|--|-----------|
| 1 | Evaluation of WP2 | 7 |
| 1.1 | Partial retrieval (T2.1) | 7 |
| 1.2 | Model reshaping (T2.2) | 15 |
| 1.2.1 | Evaluation of rigid-registration | 15 |
| 1.2.2 | Evaluation of non-rigid deformation | 20 |
| 1.2.3 | Evaluation of the mesh blending stage | 25 |
| 1.3 | Preliminary evaluation of the complete WP2 pipeline | 26 |
| 1.4 | Deviations and corrections with respect to WP2 evaluation plan | 30 |
| 1.5 | WP2 items remaining to be evaluated in D5.8 | 30 |
| 2 | Evaluation of WP3 | 30 |
| 2.1 | Evaluation Data | 30 |
| 2.1.1 | Data from the Cultural Heritage Sites | 30 |
| 2.1.2 | Accelerated Erosion Experiments | 32 |
| 2.2 | Measurement modalities | 35 |
| 2.2.1 | 3D Geometry Measurements | 35 |
| 2.2.2 | QEMSCAN | 35 |
| 2.2.3 | X-CT | 38 |
| 2.2.4 | 3D Microscopy | 39 |
| 2.2.5 | Petrography | 39 |
| 2.3 | WP3 experiments actually performed and results | 41 |
| 2.3.1 | Estimation of amount of erosion between Erosion Cycles | 41 |

| | | |
|----------|--|-----------|
| 2.3.2 | Experimental Results for E-GE Experiments | 42 |
| 2.3.3 | Experimental Results for E-GESC Experiments | 48 |
| 2.4 | Deviations and corrections with respect to WP3 evaluation plan | 48 |
| 2.5 | WP3 items remaining to be evaluated in D5.8 | 49 |
| 3 | Evaluation of WP4 | 51 |
| 3.1 | Reassembly Experiments | 51 |
| 3.1.1 | Pairwise Matching Experiments | 52 |
| 3.1.2 | Reassembly of a Single Object | 60 |
| 3.1.3 | Reassembly of Mixed Objects | 65 |
| 3.1.4 | Reassembly of Heavily Damaged Fragments | 67 |
| 3.2 | Evaluation of Object Repair | 71 |
| 3.2.1 | Dataset | 71 |
| 3.2.2 | Experiments | 72 |
| 3.3 | Deviations and corrections with respect to WP4 evaluation plan | 82 |
| 3.4 | WP4 items remaining to be evaluated in D5.8 | 83 |
| 4 | Evaluation of WP5 | 83 |
| 4.1 | Virtual Repair and Measurement Workspace Experiments | 84 |
| 4.1.1 | System-wide Features | 85 |
| 4.1.2 | Collection Management Tools | 87 |
| 4.1.3 | Object Erosion | 92 |
| 4.1.4 | Shape-guided Reassembly | 93 |
| 4.1.5 | Object Repair | 97 |

| | | |
|----------|--|------------|
| 4.1.6 | Missing parts prediction | 99 |
| 4.2 | Predictive Digitization Platform | 99 |
| 4.3 | Deviations and Corrections with Respect to WP5 Evaluation Plan . . | 100 |
| 4.4 | WP5 Items Remaining to be Evaluated in D5.8 | 101 |
| A | Parameters for the reshaping module (WP2) | 102 |
| B | Benchmark of Fragmented Objects | 103 |

This document describes the first evaluation of the outcomes of WP2, WP3, WP4 and WP5 based on the Evaluation Plan (D5.1).

1 Evaluation of WP2

WP2 evaluation comprises experiments for the assessment of the individual WP2 modules and the integrated system for predictive digitization, as described in the evaluation plan D5.1.

1.1 Partial retrieval (T2.1)

Datasets

Experiments for the evaluation of the partial retrieval module are based on two publicly available benchmark datasets. The first dataset has been used in SHREC 2013 track for large scale partial 3D object retrieval [SMB*13]. The target set has been created from 360 shapes, organized into 20 classes of 18 objects per class. The process of range scan acquisition from the objects of the target set has been simulated in order to obtain a set of partial views. This process results in 7200 queries, associated with varying levels of partiality.



Figure 1: Samples of the SHREC 2013 partial retrieval benchmark [SMB*13].

The second benchmark dataset used for evaluation is related to the CH domain and consists of 3D pottery models originating from the Virtual Hampson Museum collection (<http://hampson.cast.uark.edu>). It is publicly available and has already been presented in D2.2, as well as in PRESIOUS published articles. The dataset consists of 384 models classified to 6 distinct geometrically defined classes (bottle, bowl, jar,

effigy, lithics and others), which can further be divided in 23, more precisely defined, sub-classes. 21 partial queries have been artificially created by slicing and cap filling complete 3D models. The partial queries used in our experiments have a reduced surface compared to the original 3D object, which is associated with 25% partiality. In the case of this dataset, apart from experiments with artificially created queries, additional experiments are performed with real queries, obtained with Breuckmann Optoscan scanner as well as with Microsoft Kinect for Windows V2 sensor. For the creation of the real queries, derivative vessels (replicas) were constructed with the help of a local potter, based on subjective and objective criteria. The process of constructing replicas has been described in D1.1, whereas the sets of real queries, either acquired with Breuckmann Optoscan scanner or Kinect V2 sensor, are described in detail in D5.3.



Figure 2: Example 3D models of the pottery dataset used (<http://hampson.cast.uark.edu>).

Experimental evaluation is based on precision-recall (P-R) plots and five quantitative measures: nearest neighbour (NN), first tier (FT), second tier (ST), mean average precision (MAP), and discounted cumulative gain (DCG). MAP is used for evaluation on SHREC, whereas DCG is used for evaluation on Hampson pottery, in order to enable comparisons with previous methods. All results presented are based on the current versions of Fisher and Panoramic retrieval methods. Details on the methodological differences of these versions, as compared to the ones presented in D2.2 will be provided in D2.4.

Correctness

It has been observed that by separately employing local and global shape similarity, the retrieval performance in SHREC 2013 is significantly lower (FT approximately equal to 15% and 18%, respectively) than the one obtained by the latest version (FT 28%) of the Fisher method, verifying that complementary information is derived from these parallel processes.

Table 1 presents the retrieval performance, as quantified by NN, FT, ST and MAP,

which was obtained by the current version of the Fisher and Panoramic methods, as well as by five state-of-the-art methods, on SHREC 2013 partial retrieval benchmark. It can be observed that the Fisher method achieves the highest performance with respect to all metrics. Figure 3 illustrates the average P-R scores for all retrieval methods. It should be noted that the results presented for the state-of-the-art methods are the ones presented in a recent comparative study [SMB*14].

| Method | NN | FT | ST | MAP |
|-------------------|--------|--------|--------|--------|
| Fisher | 0.3856 | 0.2772 | 0.2135 | 0.2851 |
| Panoramic | 0.3535 | 0.2290 | 0.1808 | 0.2455 |
| SBR-2D-3D | 0.2717 | 0.2016 | 0.1426 | 0.1754 |
| SBR-VC | 0.3218 | 0.2065 | 0.1638 | 0.2199 |
| Data-aware | 0.3457 | 0.2495 | 0.2088 | 0.2836 |
| Polar spin images | 0.0931 | 0.0809 | 0.0768 | 0.0968 |
| SQFD | 0.3108 | 0.2043 | 0.1576 | 0.1978 |

Table 1: The results of the Fisher and Panoramic methods, along with 5 state-of-the-art retrieval methods on SHREC 2013 partial retrieval benchmark.

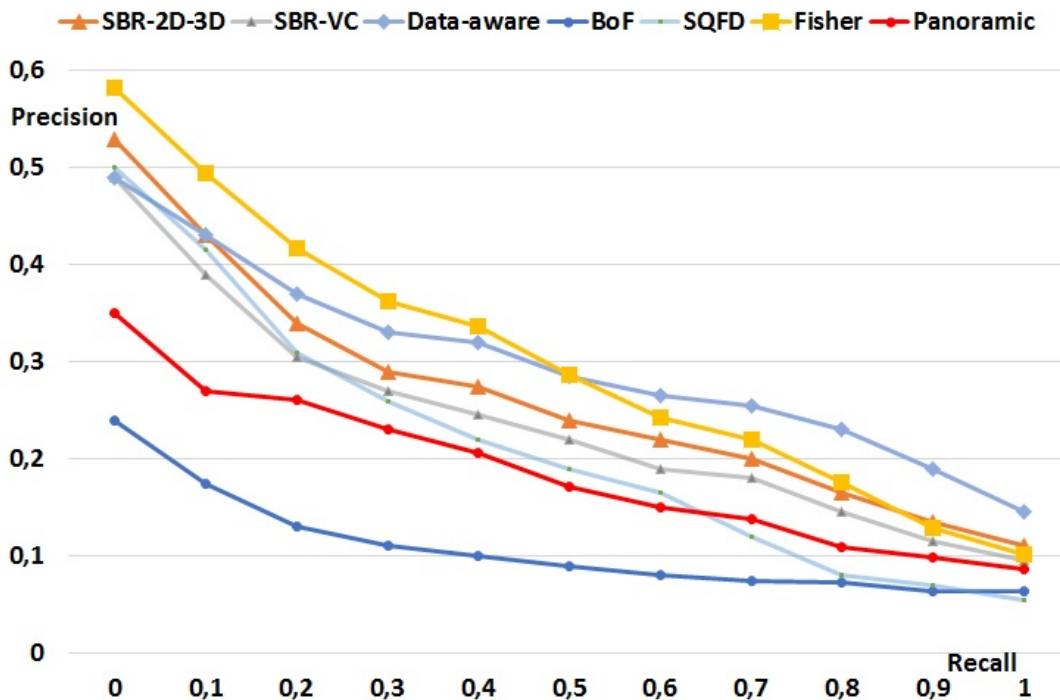


Figure 3: Average P-R for Fisher, Panoramic and 5 state-of-the-art retrieval methods on SHREC 2013 partial retrieval benchmark.

Table 2 presents the retrieval performance, as quantified by NN, FT, ST and DCG, which was obtained by Fisher and Panoramic methods on the Hampson pottery dataset, when using the set of 21 artificial partial queries. Both classifications of the dataset, on 6 and 23 classes, are considered. It can be observed that both methods achieve high retrieval performance. The Fisher method in particular, achieves NN

| Method | NN | FT | ST | DCG |
|--------------|-------|-------|-------|-------|
| Fisher-6 | 1.000 | 0.492 | 0.702 | 0.827 |
| Fisher-23 | 0.952 | 0.460 | 0.642 | 0.778 |
| Panoramic-6 | 0.838 | 0.457 | 0.685 | 0.787 |
| Panoramic-23 | 0.676 | 0.476 | 0.640 | 0.753 |

Table 2: Retrieval results of the Fisher and Panoramic methods on the Hampson pottery dataset with artificial queries.

close to perfect. Fig. 4 illustrates the average P-R scores for both retrieval methods. Interestingly, the Fisher method obtains slightly higher precisions in the case of the coarser, 6-class, classification, whereas the Panoramic method clearly obtains higher precisions in the case of the more fine-grained, 23-class, classification. Figure 5 illustrates example ranked lists obtained in the case of the Hampson pottery dataset. It should be noted that the Fisher method only requires raw point cloud information.

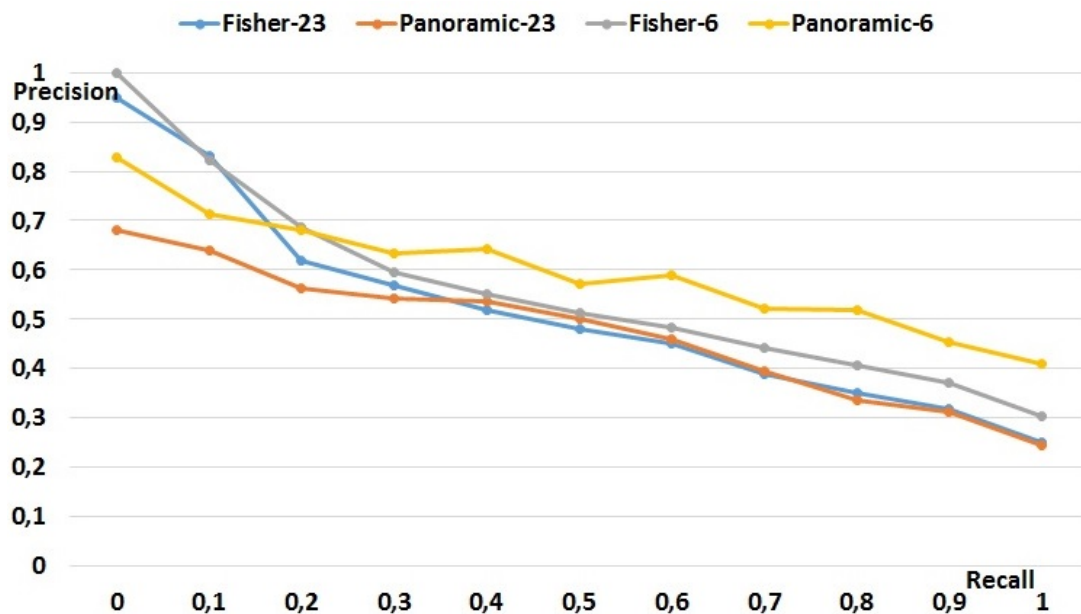


Figure 4: Average P-R for all retrieval methods applied on the Hampson pottery dataset, for artificial partial queries.

Experiments were also performed with 25 real queries obtained by using the Breuckmann Optoscan scanner to digitize vessel derivatives, specifically constructed to resemble objects from the Hampson pottery dataset (D1.1). Table 3 presents the retrieval performance, as quantified by NN, FT, ST and DCG, which was obtained by Fisher and Panoramic methods. All metrics are calculated for both the classification of the Hampson pottery dataset in 6 geometrically defined classes, as well as on 23, more precisely defined, sub-classes. Fig. 6 illustrates the average P-R scores associated with the classification in 6 and 23 classes. It can be observed that the Panoramic method slightly outperforms the Fisher method in the case of 6-class classification, whereas in the case of 23-class classification the difference in performance increases.



Figure 5: Example ranked lists obtained by the proposed 3D object retrieval method in the case of the Hampson pottery dataset. Examples of partial queries are shown in the upper row, whereas the respective top-6 objects retrieved are shown below.

Another set of experiments is based on 20 real queries, obtained by using Microsoft Kinect V2 for Windows sensor to digitize derivative vessels. Table 4 presents the retrieval performance, once more quantified by NN, FT, ST and DCG, which was obtained by Fisher and Panoramic methods. All metrics are calculated for both the classification of the Hampson pottery dataset in 6 geometrically defined classes, as well as on 23, more precisely defined, sub-classes. Fig. 7 illustrates the average P-R scores associated with the classification in 6 and 23 classes. The Panoramic method clearly achieves higher retrieval performance, whereas the retrieval performance of the Fisher method is rather low.

Table 5 and Fig. 8 provide a quantitative assessment of the enhancement in retrieval accuracy obtained by the latest versions of the Fisher (Fisher Hybrid) and the Panoramic (Panoramic-HD) methods, as compared to the retrieval accuracy of the previous versions presented in D2.2. Both names Fisher Hybrid and Panoramic-HD

| Method | NN | FT | ST | DCG |
|--------------|-------|-------|-------|-------|
| Fisher-6 | 0.641 | 0.351 | 0.554 | 0.712 |
| Fisher-23 | 0.524 | 0.199 | 0.321 | 0.562 |
| Panoramic-6 | 0.663 | 0.363 | 0.606 | 0.733 |
| Panoramic-23 | 0.554 | 0.352 | 0.538 | 0.624 |

Table 3: Retrieval results of the Fisher and Panoramic methods on the Hampson pottery dataset with 25 real queries obtained by Breuckmann Optoscan scanner.

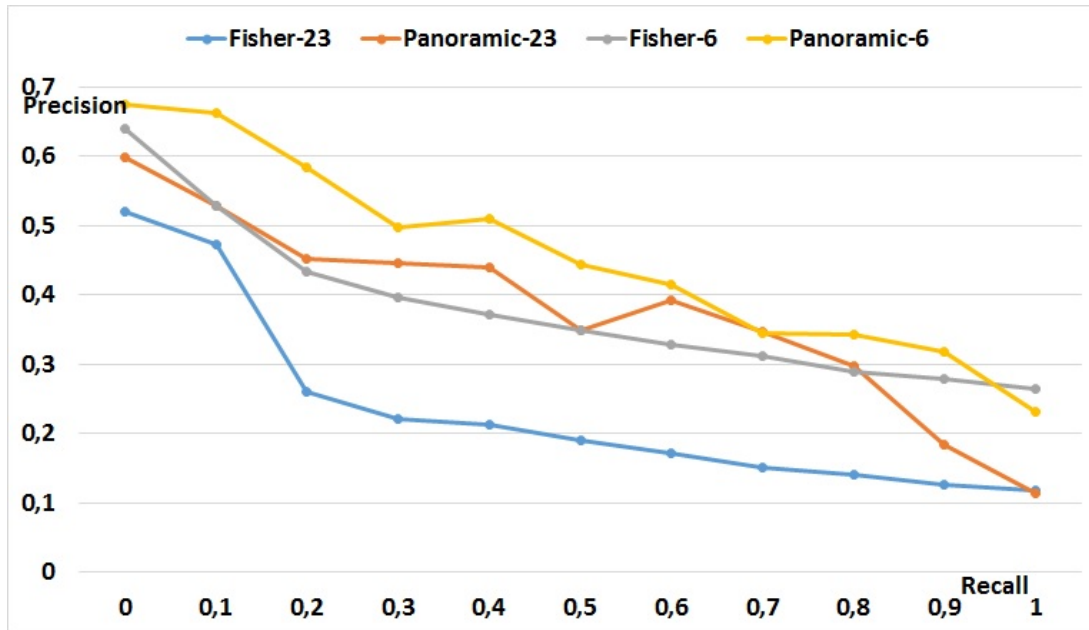


Figure 6: Average P-R for all retrieval methods applied on the Hampson pottery dataset, for 25 queries obtained by Breuckmann Optoscan scanner.

will be explained in D2.4. This comparison is performed on the Hampson pottery dataset, with 21 artificial queries and 23-class classification. A clear improvement in retrieval performance is evident for both methods presented.

Overall, it can be noted that Fisher-Hybrid obtains the highest retrieval accuracy in SHREC 2013, whereas the two methods obtain comparable retrieval accuracy in the Hampson pottery dataset for artificial queries and queries obtained with Breuckmann Optoscan scanner. Finally, Panoramic-HD obtains higher retrieval accuracy for queries obtained with Microsoft Kinect V2 sensor. In the latter case, the retrieval accuracy obtained with the Fisher-Hybrid method is still low. A possible explanation of this might be that parameterization is still suboptimal for this type of queries. This will be the subject of experimentation to follow.

Subjective quality

| Method | NN | FT | ST | DCG |
|--------------|-------|-------|-------|-------|
| Fisher-6 | 0.266 | 0.244 | 0.431 | 0.620 |
| Fisher-23 | 0.153 | 0.084 | 0.197 | 0.460 |
| Panoramic-6 | 0.484 | 0.329 | 0.558 | 0.522 |
| Panoramic-23 | 0.353 | 0.296 | 0.513 | 0.562 |

Table 4: Retrieval results of the Fisher and Panoramic methods on the Hampson pottery dataset with 20 real queries obtained by Microsoft Kinect V2 for Windows sensor.

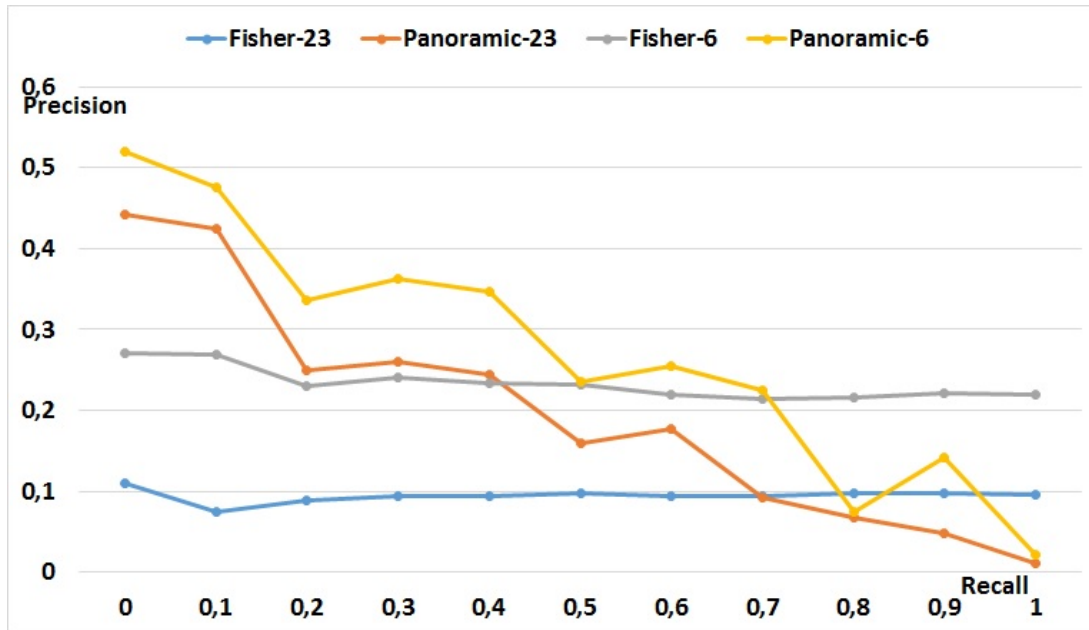


Figure 7: Average P-R for all retrieval methods applied on the Hampson pottery dataset, for 20 queries obtained by Microsoft Kinect V2 for Windows sensor.

A cultural heritage expert has inspected the retrieval results in the case of the experiments with the Hampson pottery dataset, and verified that the ranked lists obtained provide models similar to the partial query.

Speed

Both retrieval methods have been developed on a hybrid Matlab/C++ architecture. The experiments have been performed on an Intel Core i7 workstation, operating at 3.5 GHz with 16 GB of RAM.

The Fisher method requires 57433 ms for the offline calculation of Gaussian mixture models on the Hampson pottery dataset. For each 3D model, local descriptor calculation requires 781 ms, Fisher vectors calculation requires 2749 ms, distance matrix calculation requires 482 ms and the calculation of the ranked list requires 29 ms. In total, the online process for each 3D model requires 4004 ms. It can be observed that the modules associated with Fisher encoding are the most time consuming, which

| Method | NN | FT | ST | DCG |
|----------------|-------|-------|-------|-------|
| Fisher-D2.2 | 1.000 | 0.438 | 0.637 | 0.764 |
| Fisher-Hybrid | 0.952 | 0.460 | 0.642 | 0.778 |
| Panoramic-D2.2 | 0.619 | 0.416 | 0.626 | 0.721 |
| Panoramic-HD | 0.676 | 0.476 | 0.640 | 0.753 |

Table 5: Retrieval results of the current versions of the Fisher and Panoramic methods (Fisher-Hybrid and Panoramic-HD, respectively), as well as of the versions presented in D2.2, on the Hampson pottery dataset with artificial queries.

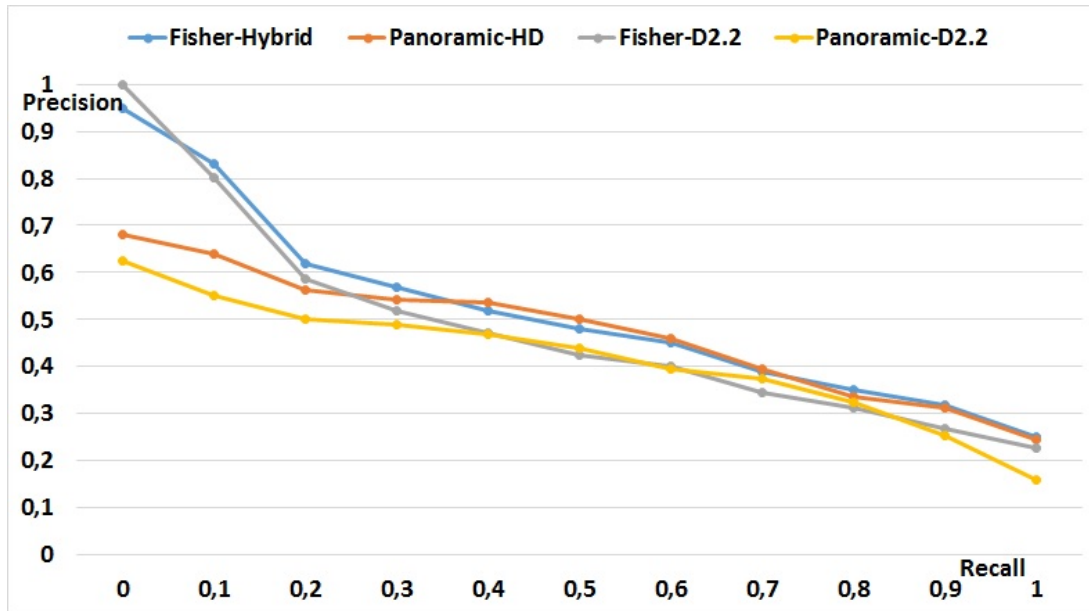


Figure 8: Average P-R for the current versions of the retrieval methods (Fisher-Hybrid and Panoramic-HD) as well as of the versions presented in D2.2, on the Hampson pottery dataset, for 25 artificial queries.

can partly be explained by the fact that these modules are currently implemented in Matlab.

The Panoramic method requires 4256 ms for the offline calculation of the bag-of-visual-words model. For each 3D model, panoramic view extraction requires 356 ms and the calculation of the ranked list requires 1756 ms. In total, the online process for each 3D model requires 2112 ms.

Robustness

Both retrieval methods are robust against the field-of-view of the partial query employed, in the sense that the obtained retrieval accuracy, as quantified by means of NN, FT, ST and DCG, is affected by less than 5%, for a partiality factor ranging from 25% to 40%.

1.2 Model reshaping (T2.2)

In this section we evaluate the fitting and reshaping module, in terms of several qualitative and quantitative measures. This module consists of a pipeline of three stages. The first stage aligns rigidly the input scan with the retrieved object (rigid registration), the second stage deforms the retrieved template in order to better match the input (non-rigid deformation), and the final stage combines the template and input datasets in order to produce the final predicted object. As described in the evaluation plan (D5.1), we will evaluate each one of these stages separately, to ensure that each component works as expected, before proceeding with the evaluation of the complete predictive digitization system.

The evaluation results throughout this section reflect our progress until the second year of the project and we expect that further improvements will be made as we continue the development of our algorithms.

1.2.1 Evaluation of rigid-registration

Registration methods search for the transformation that optimally aligns two or more surfaces. In this pipeline stage, we focus on the *pairwise* registration of a *source* to a *target* surface using a *rigid* transformation, which consists of a rotation and a translation. For more details on our rigid registration approach please refer to D2.2.

Correctness. The first experiment evaluates the correctness (or accuracy) accuracy of our alignment approach. In real-world alignment scenarios, it is often difficult or impossible to properly define a *ground truth* solution, therefore in this experiment we use two synthetically generated datasets. In particular, we took an existing model of an Owl and we created two partially overlapping scans. Using this procedure, the correct, ground truth solution is known a priori and the accuracy of the alignment can be found by computed the *Root Mean Square* (RMS) error between the registered mesh and the original one.

In this experiment, the source and target scans have 154K and 155K vertices, respectively. For the global registration stage, a uniform selection of 3% of the source surface sample points is used. Our measurements in Figure 9 indicate that even without ICP refinement, our approach reliably aligns the source and target surfaces and Sparse ICP [BTP13] effectively converges to the ground truth solution after just two iterations. The accuracy of the registration depends on the parameter p of the ℓ_p -norm used in our optimization framework. As expected, the registration error decreases with the parameter p . In practice, when the optimization is based on the ℓ_2 -norm or ℓ_1 -norm, the alignment is skewed and the optimizer did not converge to the ground truth. However, when an ℓ_p -norm is used, for sufficiently low values of p , we get the desired alignment. It is worth noting that the processing time for the ℓ_0 -norm is increased, compared to the ℓ_p relaxations, because this norm creates a larger number of local minima in the solution space, hence the ESA optimizer re-

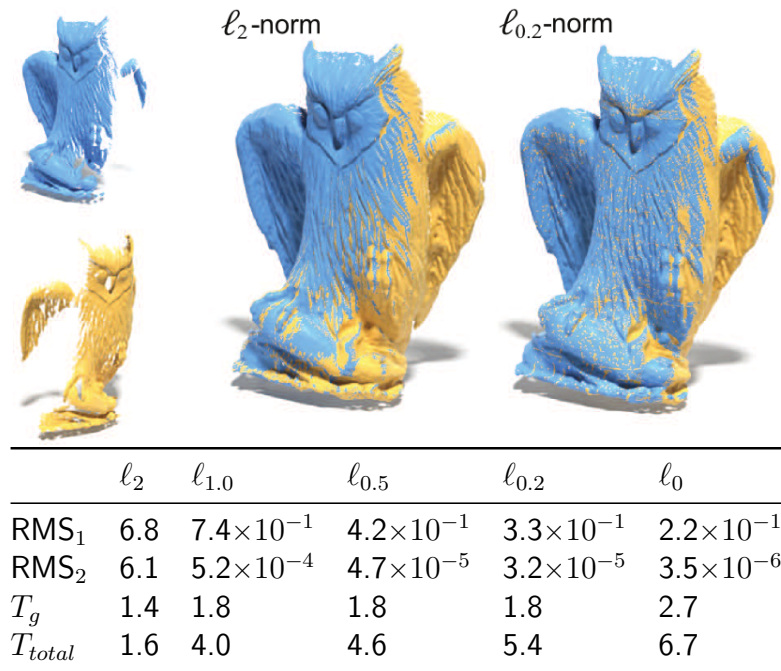


Figure 9: The accuracy of our registration procedure w.r.t. different values of p on a synthetic registration problem with a well-defined ground truth solution. We report the RMS error before (RMS_1) and after (RMS_2) Sparse ICP refinement, the global registration time for the first two levels of our pipeline (T_g) and the total alignment time (T_{total}) after Sparse ICP refinement. Timings are in seconds.

quires more iterations to find a good approximation of the global optimum. Still, the total processing time is not heavily influenced by the actual norm that is used, and our method remains both accurate and efficient for most values of p . Due to the high efficiency of our approach, in most experiments we have used the $\ell_{0.2}$ -norm, even though the improvement over $\ell_{0.4}$ or $\ell_{0.5}$ is relatively small. The reported *Root Mean Square* (RMS) error measurements are based on the Euclidean point-to-point distance of the aligned source surface and the reference model. The bounding box diagonal of the owl model is 1200 units (we report it since the RMS error depends on the scale of the objects).

Scalability and Speed. Our second experiment evaluates the runtime performance with relation to the number of points in the input scans. Since our method uses a pre-computed distance field representation (VDB) of the target surface, the number of target points does not influence the runtime performance. For this reason, we only measure the performance in relation to the number of source points. Our measurements in Table 6 indicate that our method scales roughly linearly with relation to the number of points in the source surface. It is also worth noting that alternative data structure for the distance queries, such as k-D tree and ANN, perform much worse than VDB for large datasets. This experiment was performed on an *Intel Core i7-3820* CPU at 3.6GHz with 4 cores and when possible, all algorithms use 8 threads to take advantage of hyper-threading.

| Data Structure | T_p | 154K | 77K | 38K | 9K | 2K |
|----------------|-------|-------|-------|------|-----|------|
| k-D Tree | 0.01 | 890.6 | 236.8 | 67.6 | 8.5 | 2.59 |
| ANN | 0.01 | 78.4 | 22.8 | 9.1 | 2.4 | 1.51 |
| VDB | 1.9 | 25.5 | 9.3 | 4.1 | 1.8 | 1.37 |

Table 6: Global registration time in seconds when using a discretized distance field representation (VDB), Approximate Nearest Neighbor (ANN) and the standard k-D tree data structure, w.r.t. the number of source surface samples. T_p : The time required for the creation of the corresponding data structure in seconds for the largest dataset tested.

Robustness. In the next experiment we evaluate the robustness of our registration to noise and outliers. Please note that in this experiment we include both source and destination outliers. The results in Figure 10 indicate that our method is able to provide a reliable registration, even when the input data have a significant amount of noise and outliers. In such cases, computing reliable local descriptors is extremely challenging without pre-filtering the input data. In Figure 11 we also compare the robustness of our method with Sparse ICP. Sparse ICP is known to be robust only to source outliers, and in the presence of target outliers the optimizer is trapped in an undesired local minimum. In contrast, our method does not exhibit this problem and converges to the globally optimum solution.

In Figure 12, we evaluate our rigid registration approach in a dataset with a large number of outliers and we also perform a comparison with Sparse ICP. In this case, the initial coarse alignment is computed with a few iterations of the RANSAC-based 4-PCS method [AMCO08]. The partial scans in this example are densely sampled and consist of more than 450K. This dataset poses several challenges to an alignment algorithm. First, the metallic chain (beside the book) reflects the incoming light from the scanner and creates a number of outliers. Furthermore, the background of this scene was not removed and many outliers are inherently created at the edges of the scanner’s field of view. Even in this case, our algorithm provides a very precise alignment. In contrast, the optimizer in the original Sparse ICP method is trapped in a local minimum.

Subjective evaluation. We perform a subjective evaluation by visually inspecting the results of our algorithm in a number of additional datasets. Figure 13 demonstrates additional alignment results in challenging registration problems that involve small overlap, holes or many local minima. In all cases our method provides a highly reliable registration. Figure 14 demonstrates alignment results using synthetic partial scans from the Virtual Hampson Museum dataset. In all cases the alignment is highly reliable and converges to the globally optimum solution.

Limitations. Figure 15 shows a representative dataset where our featureless registration algorithm is expected to fail. In such dataset, the alignment algorithm should focus on the alignment of salient features, such as the inscribed anchor in this particular example, instead of minimizing the overall geometric distance of the two scans.

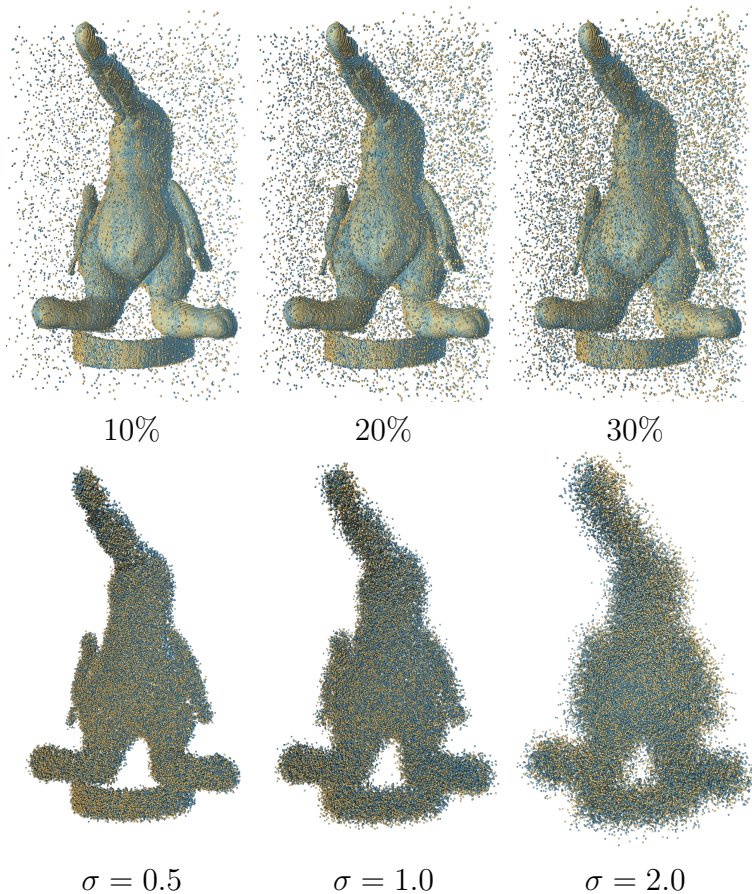


Figure 10: Registration of two partially overlapping scans of the Coati model with various degrees of outliers (top) and synthetically added zero-mean gaussian noise (bottom) with variance σ^2 . The number of outliers is measured as a percentage w.r.t. the number of input points. One unit roughly corresponds to 1% of the bounding box diagonal. The results are shown without any ICP refinement. The Coati model is from Weise et al. [WLVG07].

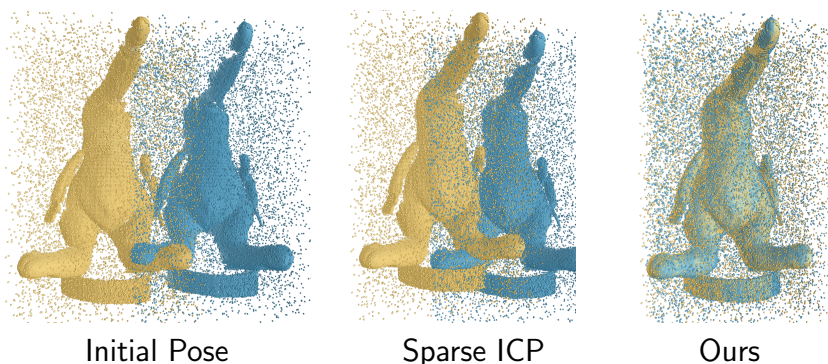


Figure 11: A comparison of our method with Sparse ICP in terms of robustness to source and destination outliers. Left: Initial position of the two surfaces. Middle: The original Sparse ICP method is known to be robust only to source outliers and in the presence of destination outliers the optimizer gets stuck to a suboptimal local minimum solution. Right: Our approach successfully aligns the two datasets, avoiding undesired local minima.

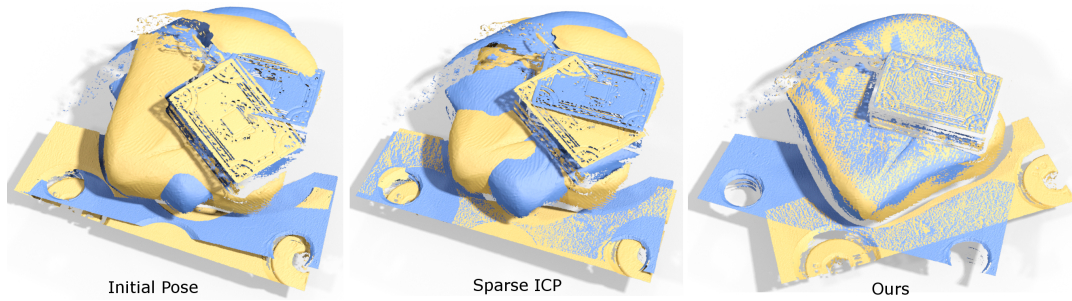


Figure 12: Rigid registration of two partially overlapping scans of a cultural heritage object. In this challenging registration problem, outliers inherently occur due to the partial overlap of the scans, especially at the edges of the scanner's field of view. Furthermore, reflected light on polished surfaces is misread by the sensor, introducing additional outliers. Left: The input scans in their initial pose. Middle: Pairwise registration using Sparse ICP. The optimizer in this case is trapped in a local minimum, failing to align the book at the middle of the scene. Right: Pairwise registration using our method provides the desired alignment. 3D dataset courtesy of Breuckmann GmbH.

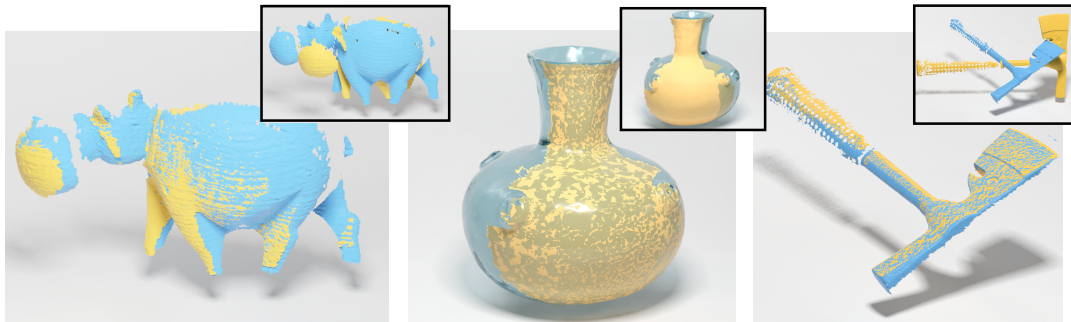


Figure 13: Rigid registration of various datasets computed with our method. Left: Registration of two partial scans with small overlap. Dataset from Mellado et al. [MAM14]. Middle: A registration problem with many local minima, due to the rotational symmetry of the pottery object. Right: Registration of two partially overlapping scans of an axe. The insets show the initial position of the objects. The axe 3D dataset is courtesy of Breuckmann GmbH.

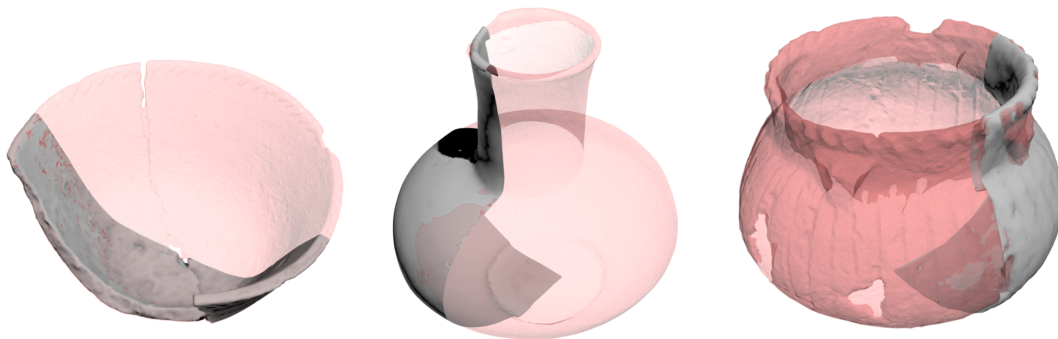


Figure 14: Alignment results using synthetic partial scans from the Virtual Hampson Museum dataset. The synthetic partial scans are shown as opaque and the complete template object is shown as transparent red.

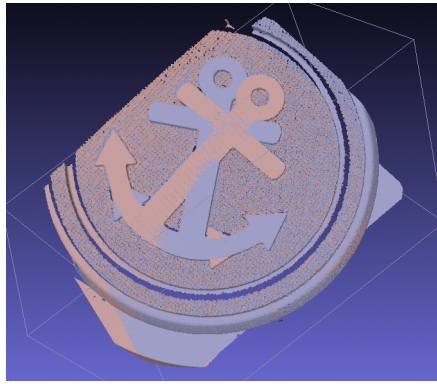


Figure 15: A failure case for our feature-less registration approach. Our algorithm aligns the overall shape of the two partial scans, but doing so, fails to align the inscribed anchor sign at the middle. A feature-based alignment algorithm can properly handle this case, by detecting and prioritizing the alignment of salient features. 3D dataset courtesy of Breuckmann GmbH.

In the context of the reshaping, we will use our registration method for part-in-whole registration problems, such as the ones shown in Figure 14, therefore this limitation of the rigid registration algorithm will not pose a problem in our predictive digitization pipeline.

Conclusion. The evaluation of the rigid registration component demonstrates that the developed methodology provides correct and reliable registration in a variety of challenging registration problems that involve partially overlapping surfaces, it is highly robust to noise and outliers, it is scalable and outperforms previous state-of-the-art registration algorithms.

1.2.2 Evaluation of non-rigid deformation

The second stage of the pipeline computes a *non-rigid* local deformation of the source object (retrieved template object) in order to better match the target one (input scan). This procedure is often referred to as *non-rigid registration* in the bibliography. For more details on the developed method please refer to D2.2.

Correctness. Non-rigid registration algorithms are often used when scanning moving subjects. We will use such dataset, that consists of a closing hand, in our first experiment, where we evaluate the correctness of our approach. We perform this evaluation by both measuring the RMS Euclidean distance between the source and the target surface before and after the non-rigid deformation and by visually inspecting the results. To make the registration problem even more challenging, we have synthetically created a number of holes and missing parts in the target surface. As shown in Figure 16, rigid registration cannot perfectly align the two meshes, since the geometry has been deformed between the two scans. In contrast, our non-rigid registration approach successfully deforms the source surface in order to match the target one. At the same time, the visual inspection of the results indicate that all

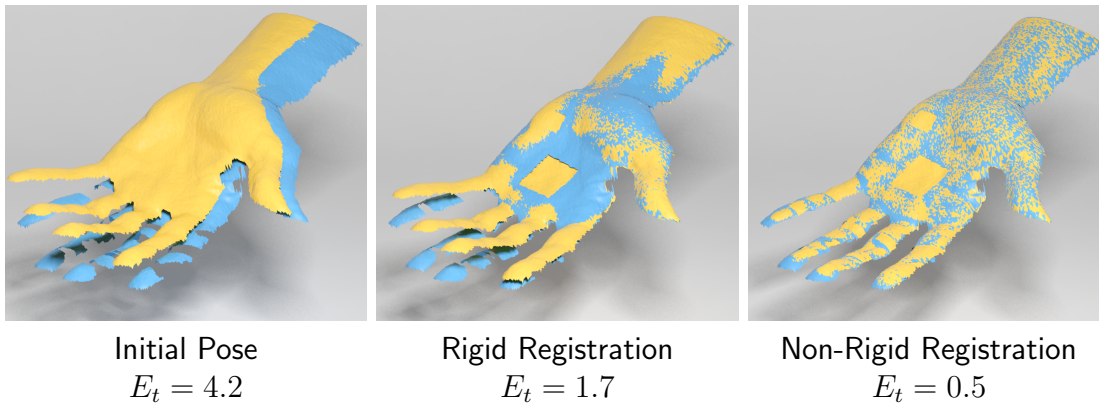


Figure 16: A challenging alignment problem that involves a closing palm. Since the palm is deforming, rigid registration cannot fully align the two models. The result of the rigid registration, shown at the middle, is used for the initialisation of a non-rigid deformation that successfully aligns the two meshes. The holes in the target point cloud, which appear as uniform yellow regions in the resulting non-rigid registration, were synthetically generated to make the alignment problem more challenging. The reported RMS error includes the holes (we did not use a cut-off threshold). Dataset from Weise et al. [WLVG07].

parts of the target surface with missing geometry are filled with information from the template object.

Scalability. In the next experiment we evaluate the scalability of our approach in terms of the input size. Our algorithm computes the desired deformation by solving a $3N \times 3N$ linear system, where N is the number of points in the source (template) point cloud. Therefore, the runtime complexity of our approach is proportional to the number of points in the source surface.

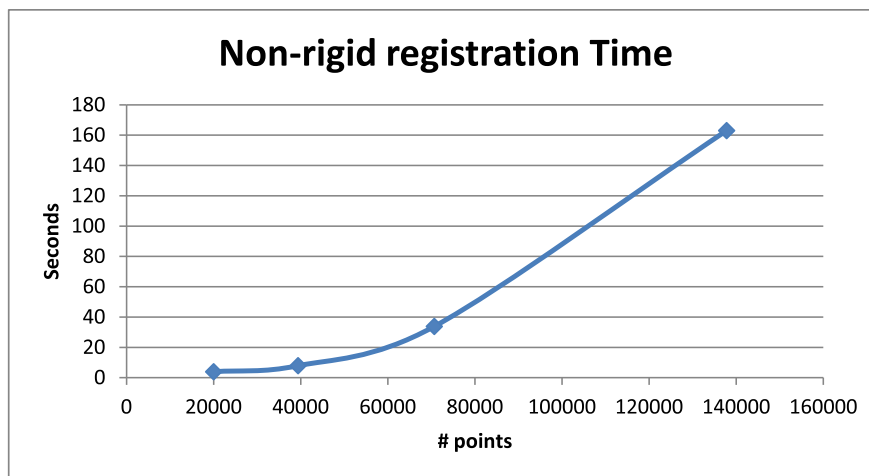


Figure 17: The performance of our non-rigid registration approach w.r.t. the number of points in the retrieved template object. The plot reflects the quadratic relation between the input points and the execution time. This is to be expected, since the computational complexity of our method is $O(n^2)$.

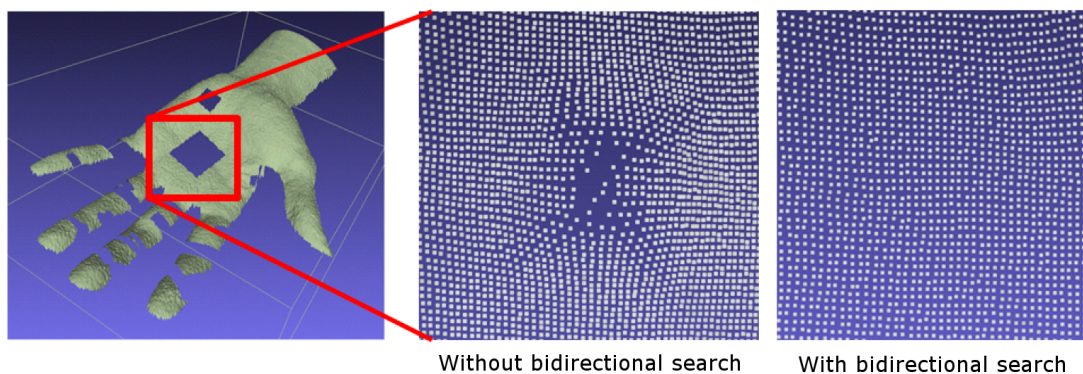


Figure 18: Our method avoids undesired distortions in areas of missing geometry using a bidirectional search heuristic. Without this heuristic, point in the template surface are moved towards the target one, created undesired stretching and distortion in the deformed object, as shown at the middle.

To better handle dense point clouds, we perform a uniform poisson-like sub-sampling of the source point cloud. The deformation is then computed using the low-resolution point cloud and then transferred back to the dense one. The sub-sampling is computed very efficiently using Approximate Nearest Neighbor (ANN) queries in a few milliseconds, but it can also be pre-computed for each template model. Figure 17 measures the performance of the non-rigid registration with respect to the size of the source point cloud. From this chart it is clear that without the subsampling, our method would be impractical for large datasets.

Robustness. In this experiment we evaluate the robustness of our approach to imperfections in the input scans, such as holes and other missing geometry. Our method includes a *bidirectional search* heuristic in order to handle these cases without introducing any distortion. The hand in Figure 16 includes several missing parts that were properly handled by our approach. Figure 18 demonstrates the prediction of the missing part, with and without the *bidirectional search* heuristic. We observe that without this heuristic, the surface points are skewed towards the target surface, creating unwanted distortion. The use of this heuristic by our method avoids this problem.

Subjective evaluation. We perform a subjective evaluation by visually inspecting the results of our algorithm in a number of additional datasets. The experiment in Figure 19 shows how our non-rigid registration method can be used to fill the missing parts in an incomplete scan of a marble statue. While the back of the original statue was impossible to scan, due to close proximity to a wall, it was possible to scan the back of a replica. The information in the replica was used to complete the missing parts in the original statue. We observe that after deformation, the quality of the registration in the overlapping regions is remarkable and all missing parts were filled with information from the replica object, without any noticeable discontinuities.

Non-rigid deformation methods are often used when scanning human subjects. In the

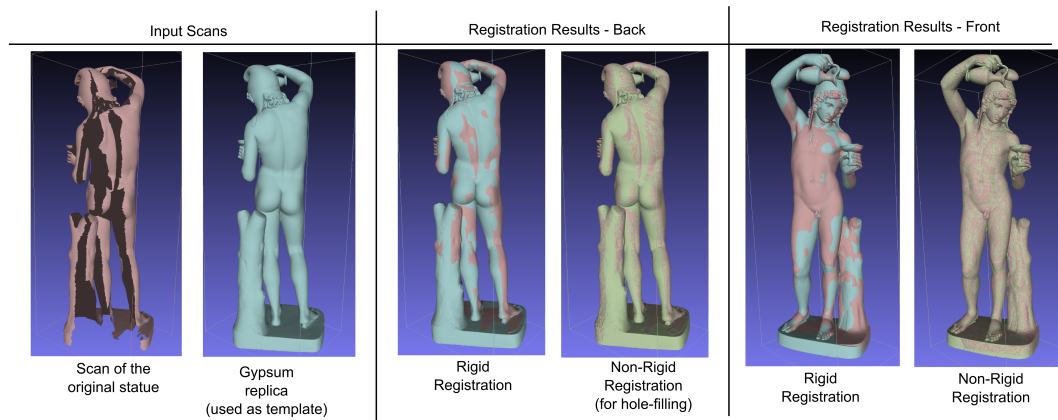


Figure 19: Scan prediction using non-rigid registration. The back of the original statue was close to a wall, so it was not possible to properly digitize and many parts are missing in the final scan. However, it was possible to scan a gypsum replica of the statue. In this experiment, this replica was used to fill the missing parts on the original statue. 3D dataset courtesy of Breuckmann GmbH.

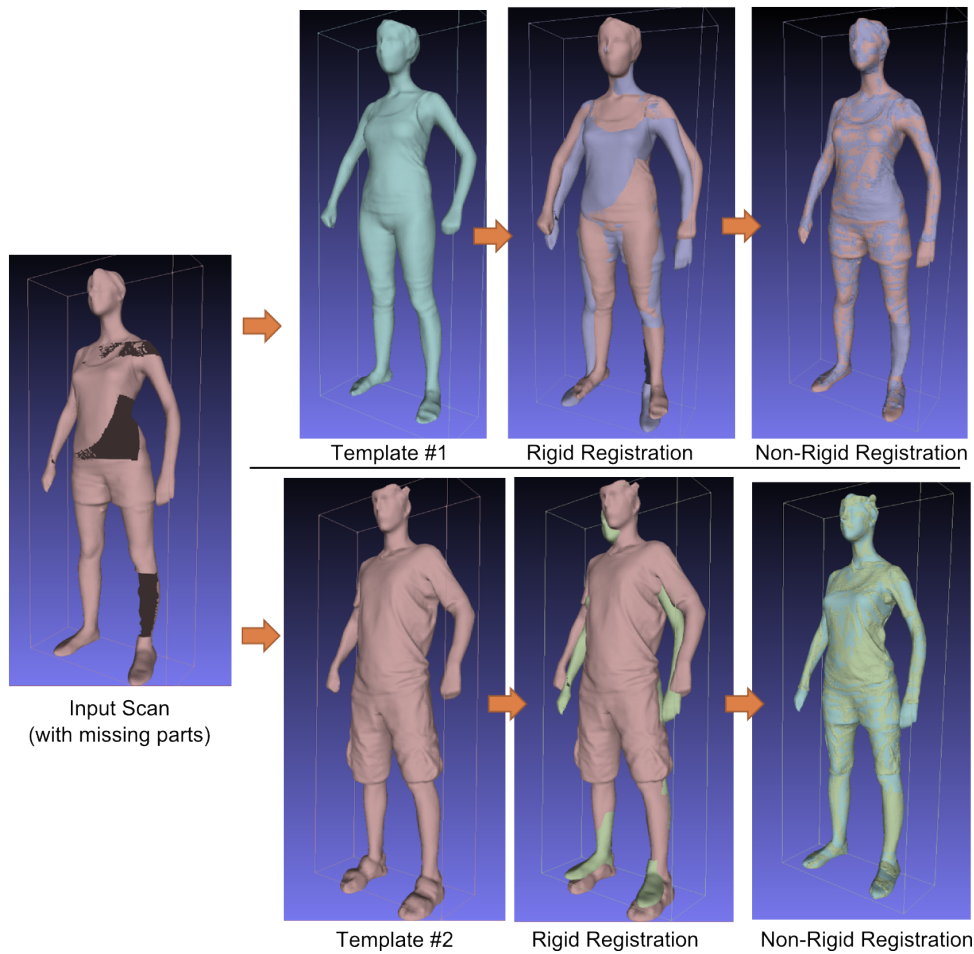


Figure 20: In this experiment, predictive scanning using non-rigid registration is used to complete the missing parts of the input female model. We demonstrate prediction using two different template objects. The first template is more similar to the input model and for this reason the prediction is more reliable. Dataset from [IPOS14, CI11].

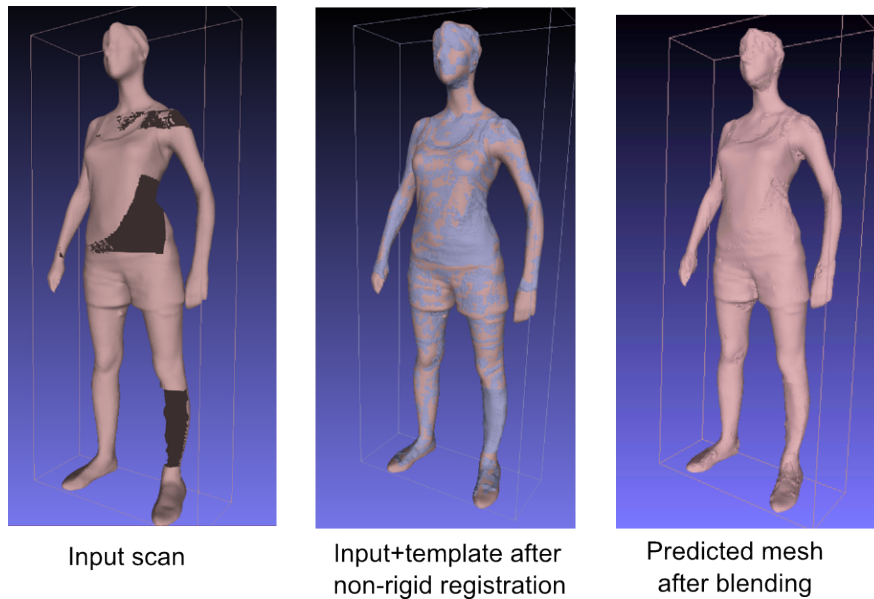


Figure 21: The mesh blending stage of our prediction pipeline merges the two registered meshes in order to produce the final predicted scan. The reference object is shown at Figure 20. Dataset from [IPOS14, CI11].

next experiment, shown in Figure 20, the input 3D scan of a female model has several missing parts. We demonstrate prediction results using two different template objects. The first template is the scan from another female model, with similar shape as the input one. The second template belongs to a man, therefore it has more differences when compared to the input scan. In both cases, our non-rigid registration approach successfully aligns the template meshes with the input scans and completes the missing parts. However, we observe that when the first reference object is used, we get better prediction results. Therefore, we conclude that using a suitable reference object that matches as much as possible to the input scan is critical when performing prediction using this approach.

Limitations. Our non-rigid registration approach is based on closest-point correspondences. A number of heuristics is used in order to improve the reliability of the correspondences and to discard invalid ones, but in all cases, the accuracy of the results depends on the suitability of the reference object. It is worth noting most (if not all) non-rigid registration methods that are based on closest-point correspondences have a similar limitation.

Conclusions. We have evaluated our non-rigid registration approach in a number of datasets that involve deforming objects with missing geometric parts and holes. Our experiments indicate that our method handles these cases robustly and efficiently under the assumption that the reference object and the input scan are sufficiently similar.

1.2.3 Evaluation of the mesh blending stage

The purpose of the mesh blending stage is to merge the source and target surfaces into one mesh. In regions where there is no overlap between the two surfaces, the blending operation simply keeps the surface (either from source or target) that covers this region of space. When there is overlap, three possible choices exist:

- Keep only the target surface (input scan): used in prediction/completion applications.
- Keep only the source surface (reference object): used to replace a low-res input with a high detail template.
- Average the source and target: used to merge partial scans in 3D data acquisition.

All these modes can be trivially implemented and do not require much testing. In our workflow, where we want to predict the missing information in an accurate but incomplete input scan, we use the first option. The continuity of the predicted mesh after the mesh blending stage is guaranteed by the rigidity constraints of the non-rigid registration stage. The current version of mesh blending requires approximately 20-30 ms.

Visual inspection and subjective evaluation. Figure 21 shows the mesh blending stage for the dataset in Figure 20. Visual inspection of the results indicates that this step works as expected, without introducing any undesired discontinuities.

Conclusions. The mesh blending methodology that has been developed sufficiently covers the WP2 requirements.

1.3 Preliminary evaluation of the complete WP2 pipeline

In this section we perform a preliminary evaluation of the complete predictive digitization platform. Even though some modules are still under development, this evaluation is still educational in order to investigate the efficiency of the developed methodology, highlight potential failure cases and assess the parts of the pipeline that require improvement.

Datasets and Testing Methodology. Our predictive digitization framework requires a database of already digitized objects. For this database we have used the objects in the *Virtual Hampson Museum* collection. In order to create queries with partial scans for this database, we have created a number of real-world replicas that resemble (but not exactly match) some of the objects in this collection (see Section 1). Subsequently, these replica objects were scanned, and we have used partial scans from this scanning session as queries, in order to evaluate our predictive digitization framework. We will refer to the objects in this dataset by their IDs (ex. HM_442).

Metrics. For a predicted object O_{pr} , we compute the total prediction error e against a ground truth object O_{gt} as:

$$e(O_{pr}, O_{gt}) = \frac{1}{2} \cdot (RMS(O_{pr}, O_{gt}) + RMS(O_{gt}, O_{pr})),$$

where $RMS(X, Y)$ measures the point-to-point Root Mean Square Euclidian distance from object X to Y . This bidirectional error measurement is necessary in order to better handle cases where one object is a subset of the other. Before the distance measurements, the two objects are aligned using Sparse ICP.

Accuracy. To measure the accuracy of the pipeline we use the RMS distance metric that was described in the previous paragraph. Aside from the quantitative evaluation, we will perform a qualitative one, by visually inspecting the predicted objects. The exact parameters used in these experiments for the reshaping module are shown in Appendix A.

In the first experiment we use as query a partial scan from the replica of the HM_1265 object. The ground truth in this case is the complete replica of the HM_1265 object and not the one in the database. Figure 22 shows the first 4 retrieved objects according to their rank and the resulting predicted model when each one of them is used as a template for prediction. The result of the rigid registration is also shown, in order to assess the amount of deformation required by each case. In this experiment, the first retrieved object is the actual (not the replica) HM_1265 object. However, a quick inspection of the rigid registration results reveals that the proportions and the exact curvature of the replica object and the real one do not match. The non-rigid registration algorithm deforms the replica in order to better match the real object. Figure 22 also shows prediction results with retrieved template objects with lower rank. In this case, as expected, the best prediction is achieved when the the highest ranking template object is used. It is worth noting that the third retrieved template

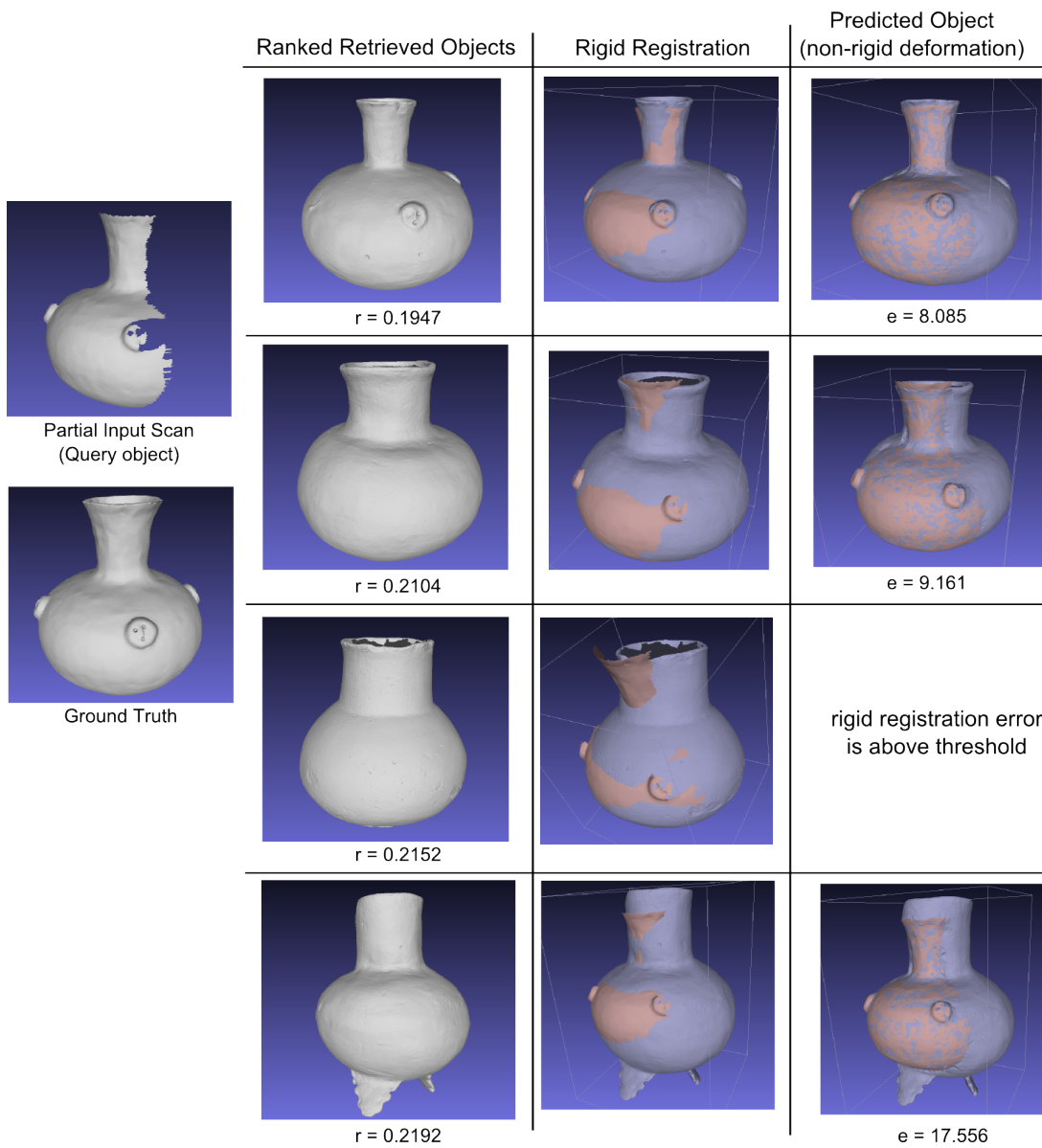


Figure 22: Predictive scanning results using the proposed pipeline. The query object is a partial scan from the replica of the HM_1265 object. Prediction results are shown for the first four template objects by rank. For the third retrieved object, the rigid registration algorithm cannot properly align the objects, thus a prediction is not possible. We report the ranking score of each template object (r) and the RMS error of the predicted object against the ground truth (e). In the rigid registration and prediction results, with red color is shown the query object and with blue the deformed template.

object cannot be properly aligned with the input scan, therefore a prediction is not possible in this case. The results of this experiment indicate that the quality of the prediction depends from the relevance of the retrieved template object.

In Figure 23 we repeat the same experiment using as query a partial scan from the replica of the HM_333 object. In this case, the actual HM_333 object is not in the top retrieved objects, therefore the accuracy of the prediction by the reshaping module is lower. We also observe that the minimum prediction error was achieved with the third retrieved object, and not the first one. However, this is still a good result since the actual object is in the top-3 of the ranked list of 384 objects and can be selected by the user. This capability of manual involvement is supported by the integrated environment.

Discussion. Close inspection of the results reveals a number of common defects in the results. In particular:

Distortion. The predicted scans exhibit some distortion artifacts. The reason for this is the low polygon count of the templates (from 8k to 20k points). Since we are displacing points in order to create the predicted scans, better results can be achieved when the high-resolution versions of the templates are used, at the expense of an increase in the execution time.

Lack of symmetry. Pottery objects are known to be roughly symmetrical. However, the partial scans in our test have information from just one side of the scan. The reshaping module properly deforms the part of the template object that overlaps with the partial scan, but has no information about the rest of the object, thus creating non-symmetrical results. A potential solution to this problem is the use of self similarity, or symmetry, in our deformation algorithm, but the challenging part is to keep the algorithms general.

Correspondences. In more than one occasion, the upper edge (rim) of the template pot did not align properly with the rim of the partial input scan. The same problem can be often observed at the bottom of the shapes involved. A solution to this problem is to improve the quality of the correspondences during the non-rigid deformation stage. To this end, rich local descriptors should be used (now we use only the normals).

Conclusions. Our preliminary evaluation of the complete WP2 pipeline has allowed us to pinpoint the parts of the pipeline that require improvement. We will focus on these areas in the near future, in order to further improve our results.

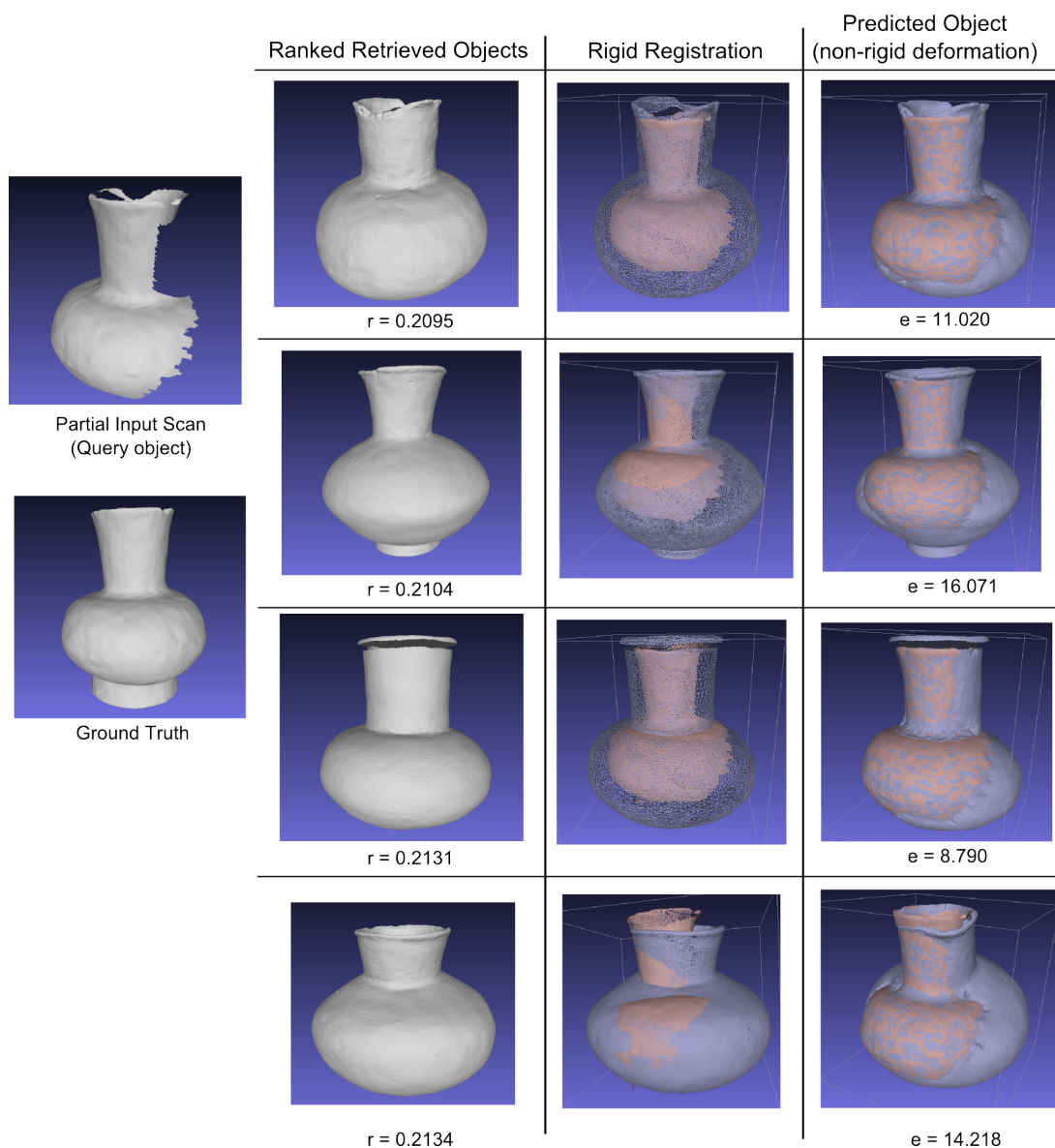


Figure 23: Predictive scanning results using the proposed pipeline. The query object is a partial scan from the replica of the HM_333 object. Prediction results are shown for the first four template objects by rank. We report the ranking score of each template object (r) and the RMS error of the predicted object against the ground truth (e). In the rigid registration and prediction results, with red color is shown the query object and with blue the deformed template.

1.4 Deviations and corrections with respect to WP2 evaluation plan

Generally, there were no major deviations or corrective actions from the WP2 evaluation plan.

For the evaluation of the reshaping module, instead of using the average Euclidean distance (along with min/max) in order to measure the accuracy of our registration methods, as we have described in the evaluation plan, we have used the RMS Euclidian distance between the source and target surfaces. This is a more appropriate metric and is widely used in the relevant bibliography.

1.5 WP2 items remaining to be evaluated in D5.8

In this first evaluation report we have mostly focused on the evaluation of each individual module of our pipeline and we have also provided some preliminary results from the evaluation of the complete pipeline. As we improve our algorithms further, in D5.8 we plan to perform a more comprehensive evaluation of the complete pipeline, which will involve:

- Experiments with varying degrees of partiality.
- Experiments with noisy data from the Microsoft Kinect sensor.
- Comprehensive measurements using additional queries.

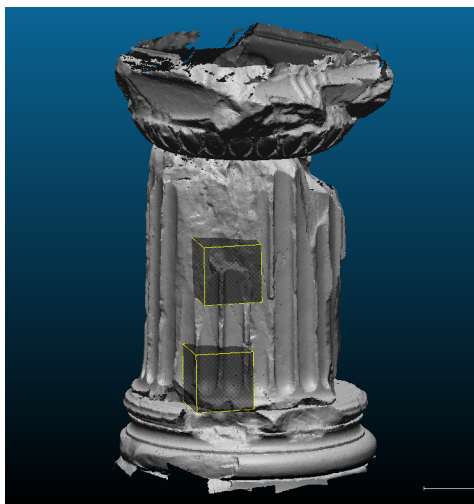
2 Evaluation of WP3

2.1 Evaluation Data

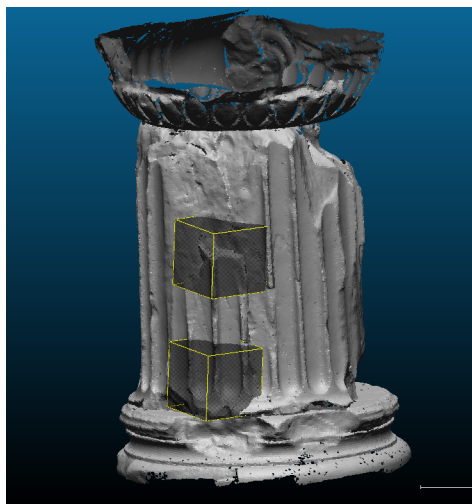
2.1.1 Data from the Cultural Heritage Sites

For the investigation of the erosion mechanisms – that contribute to the degradation of the Elefsis and Nidaros stones – we collected geometry data from the two Cultural Heritage sites. Figure 24 shows a lower resolution mesh of consecutive geometry scans that took place at Elefsis in March 2013 and October 2014. The areas of the Elefsis-column – that are marked with boxes – indicate the patches we selected for illustration of measurements and/or investigations (compare Figure 25).

At the Nidaros cathedral several smaller areas were selected for scanning. This includes two wall parts from the Lectorium (Lectorium East [with Mason Marks] and Lectorium North) and two scans from the inside of the North West and South West

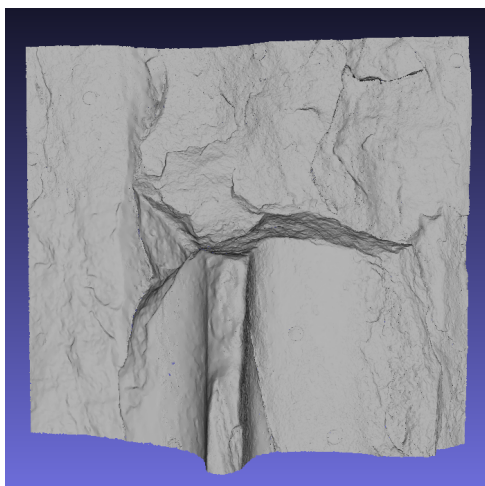


Geometry Mesh (Round 01,
2013-03-20)

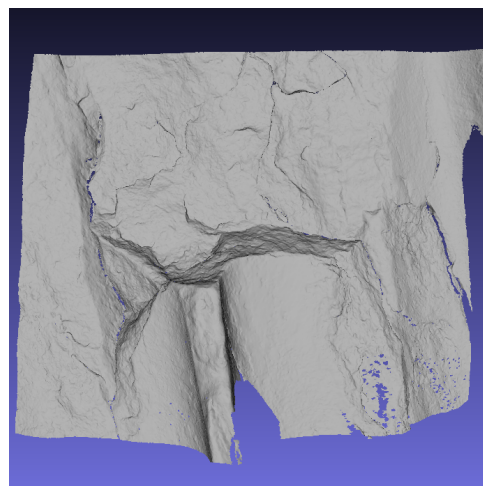


Geometry Mesh (Round 02,
2014-10-14)

Figure 24: Two geometry meshes of the Elefsis-column. This data was acquired in March 2013 and in October 2014. The patches that correspond to the upper box are shown in more detail in Figure 25



Elefsis Patch A (Round 01,
2013-03-20).

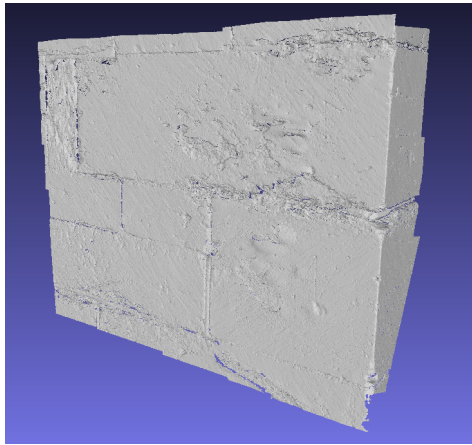


Elefsis Patch A (Round 02,
2014-09-24)

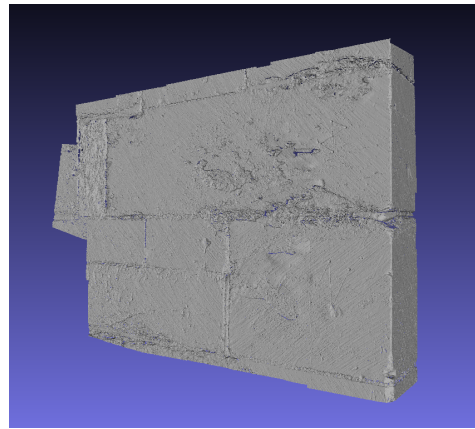
Figure 25: Two Elefsis-column patches of approximately the same area of the column that was scanned at different times. Note that the shown area was scanned with high resolution.

Tower of the cathedral. In Figure 26 we illustrate the geometry scan of the east wall of the Lectorium that contains two mason marks. A close-up view of the areas with the mason marks is depicted in Figure 27

Table 7 summarizes the gathered data sets from the Cultural heritage sites.



Geometry Mesh (Round 01,
2013-04-10)



Geometry Mesh (Round 02,
2014-09-24)

Figure 26: Two geometry meshes of the Nidaros cathedral. This data was acquired in April 2013 and in September 2014. Patches that show the mason marks in more detail are shown in Figure 27

| Geometry 3D Scan (Round 01) | Geometry 3D Scan (Round 02) |
|--|--|
| 20130320_Elefsis | 20141014_Elefsis |
| 20130410_Nidaros_LectoriumEastMasonMarks | 20140924_Nidaros_LectoriumEastMasonMarks |
| 20130410_Nidaros_LectoriumNorth | 20140924_Nidaros_LectoriumNorth |
| 20130410_Nidaros_NorthernWestTower | 20140925_Nidaros_NorthernWestTower |
| 20130410_Nidaros_SouthernWestTower | 20140925_Nidaros_SouthernWestTower |

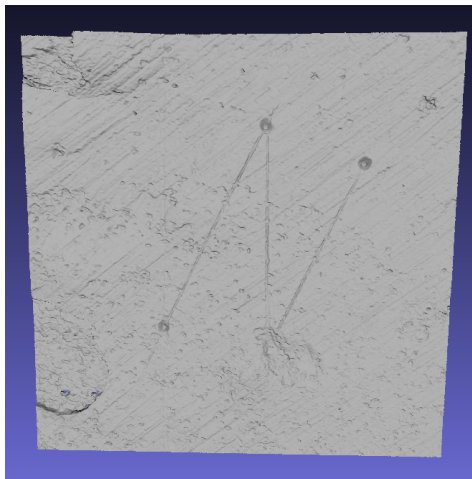
Table 7: Geometry Scans that are available from the two Cultural heritage sites.

2.1.2 Accelerated Erosion Experiments

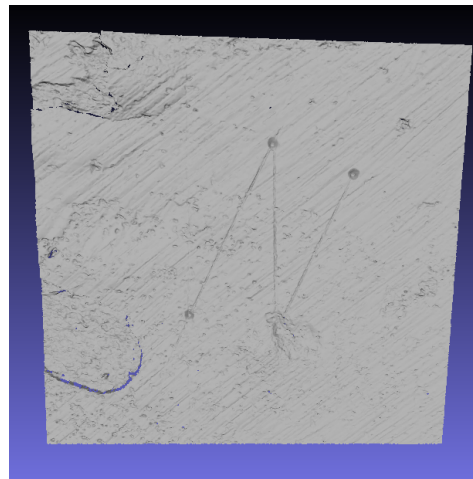
Realizing that the collected geometry data would not be sufficient for the investigation of the erosion mechanisms, we designed accelerated erosion experiments with the aim to observe and measure selected degradation processes on stone slabs. The accelerated erosion experiments themselves are described in more detail in deliverable D1.2 (2nd Annual Project Review). Here we describe these experiments in a brief form with focus on the data obtained from the different measurement methods. Therefore in the next sections we provide a (non-exhaustive) list of the data we acquired so far.

Stone Slabs and Experiment Assignment

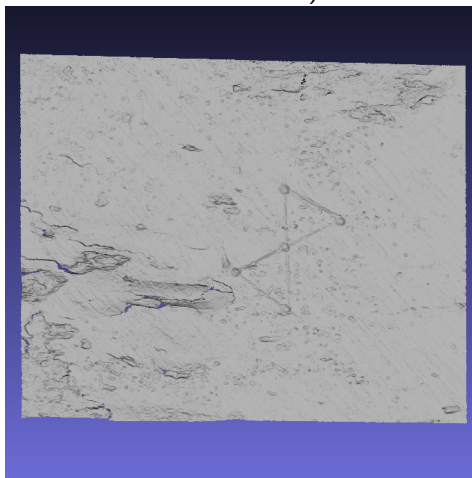
In this section we briefly describe the stone slabs and experiments we created for our accelerated erosion experiments. The stone slabs relate to stones that are similar to the ones used at the two Cultural Heritage sites. First we intended to perform accelerated erosion on just a few larger stone slabs and named them according to their origin (Elefsis, Nidaros) and their suspected stone quality (Good, Bad). It quickly



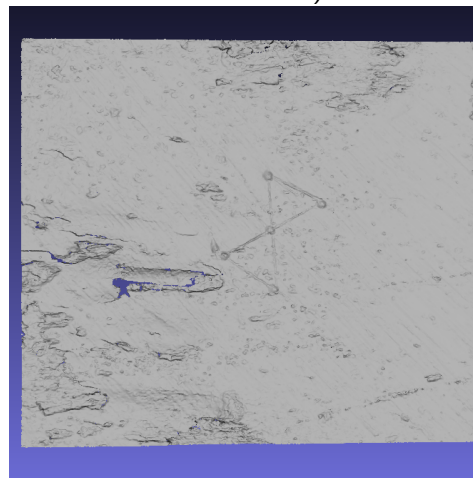
N mason mark (Round 01,
2013-04-10)



N mason mark ((Round 02,
2014-09-24)



X mason mark (Round 01,
2013-04-10)



X mason mark ((Round 02,
2014-09-24)

Figure 27: Patches of two geometry measurement rounds showing the N and X mason marks that are present on the east wall of the Lectorium of the Nidaros Cathedral.

became clear that more stone slabs were necessary for performing accelerated erosion experiments and initial larger stone slabs were subdivided leading to an extended naming scheme of the stone slabs. Experiments we finally decided to perform, include the Salt-effect (Na_2SO_4), the Freeze and Thaw effect and two chemical experiments simulating polluted environments ($\text{H}_2\text{SO}_4(\text{aq})$ and $\text{H}_2\text{SO}_4 + \text{HNO}_3(\text{aq})$). Table 8 lists which stone slabs are assigned to which experiment. Note, the designed erosion chambers and experiments are described in detail in deliverable D1.2 (2nd Annual Project Review).

| | Na ₂ SO ₄ Salteffect | Freeze and Thaw | H ₂ SO ₄ (aq) (A) | H ₂ SO ₄ +HNO ₃ (aq) (B) | Outdoors |
|-----------------------------|--|-----------------|---|---|----------|
| Elefsis.Large_01 | | R01,R02 | | | |
| Elefsis.Large_02 | R01,R02 | | | | |
| Elefsis.Large_03 | | | | R01,R02 | |
| Elefsis.Small_01 | | | R01,R02 | | |
| Elefsis.Small_02 | | | | | T |
| Elefsis.Small_03 | | | | | E |
| Nidaros.Bad.Large_01 | | R01,R02 | | | |
| Nidaros.Bad.Large_02 | R01,R02 | | | | |
| Nidaros.Bad.Large_03 | | | | R01,R02 | |
| Nidaros.Bad.Small_01 | | | R01,R02 | | |
| Nidaros.Bad.Small_02 | | | | | T |
| Nidaros.Bad.Small_03 | | | | | E |
| Nidaros.Good.Large_01 | | R01,R02 | | | |
| Nidaros.Good.Large_02 | R01,R02 | | | | |
| Nidaros.Good.Large_03 | | | | R01,R02 | |
| Nidaros.Good.Small_01 | | | R01,R02 | | |
| Nidaros.Good.Small_02 | | | | | T |
| Nidaros.Good.Small_03 | | | | | E |
| Elefsis.ThinSection_01 | | R01,R02 | | | |
| Elefsis.ThinSection_02 | R01,R02 | | | | |
| Elefsis.ThinSection_03 | | | R01,R02 | | |
| Elefsis.ThinSection_04 | | | | R01,R02 | |
| Elefsis.XRD_01 | | R01,R02 | | | |
| Elefsis.XRD_02 | R01,R02 | | | | |
| Elefsis.XRD_03 | | | R01,R02 | | |
| Elefsis.XRD_04 | | | | R01,R02 | |
| Nidaros.Good.ThinSection_01 | | R01,R02 | | | |
| Nidaros.Good.ThinSection_02 | R01,R02 | | | | |
| Nidaros.Good.ThinSection_03 | | | R01,R02 | | |
| Nidaros.Good.ThinSection_04 | | | | R01,R02 | |
| Nidaros.Good.XRD_01 | | R01,R02 | | | |
| Nidaros.Good.XRD_02 | R01,R02 | | | | |
| Nidaros.Good.XRD_03 | | | R01,R02 | | |
| Nidaros.Good.XRD_04 | | | | R01,R02 | |
| Nidaros.Bad.ThinSection_01 | | R01,R02 | | | |
| Nidaros.Bad.ThinSection_02 | R01,R02 | | | | |
| Nidaros.Bad.ThinSection_03 | | | R01,R02 | | |
| Nidaros.Bad.ThinSection_04 | | | | R01,R02 | |
| Nidaros.Bad.XRD_01 | | R01,R02 | | | |
| Nidaros.Bad.XRD_02 | R01,R02 | | | | |
| Nidaros.Bad.XRD_03 | | | R01,R02 | | |
| Nidaros.Bad.XRD_04 | | | | R01,R02 | |

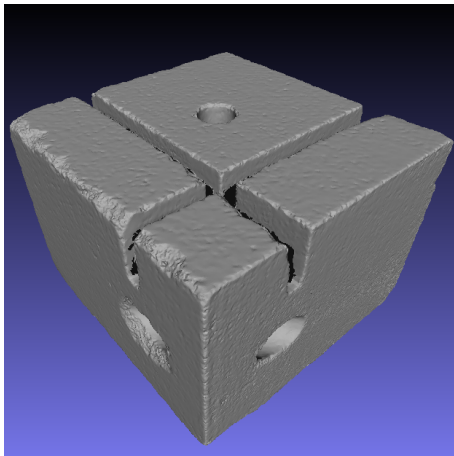
Table 8: Stone slabs along with their assigned experiment. R01 and R02 refer to the round of experiment the stones are in or have finished (R01=Round 01, R02=Round 02). The outdoor locations E and T refer to Elefsis and Trondheim respectively.

2.2 Measurement modalities

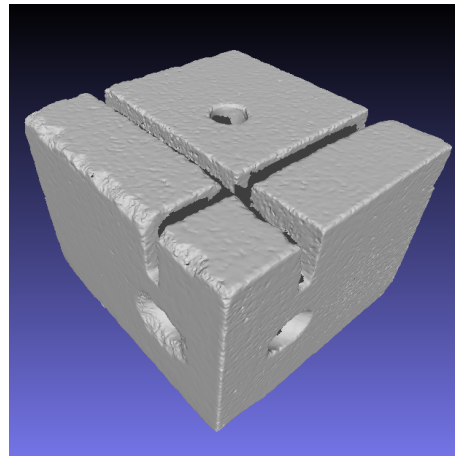
Several different measurement techniques were used to measure and observe the changes that happen on the stone slabs during the accelerated erosion cycles. The measurements consist of 3D Geometry Scans, QEMSCAN (Quantitative Evaluation of Minerals by SCANNing electron microscopy), Micro X-ray CT, 3D microscopy and Petrography. Below we summarize the data sets we currently collected from these measurements and if applicable we show examples of the data for illustration purposes.

2.2.1 3D Geometry Measurements

The 3D scans of the stone slabs result in high resolution surface meshes of the 3D geometry of the stones. An example for the resulting mesh data is shown in 28 and Table 9 lists all collected data sets.



Stone Slab: Elefsis Large 01 (Round 01, 2014-06-02).



Stone Slab: Elefsis Large 01 (Round 02, 2015-01-12)

Figure 28: Two Elefsis-column patches of approximately the same area of the column that was scanned at different times. Note that the shown area was scanned with high resolution.

2.2.2 QEMSCAN

QEMSCAN (Quantitative Evaluation of Minerals by SCANNing electron microscopy) is a technique that uses a Scanning Electron Microscope (SEM) combined with X-ray spectroscopy and a database to obtain accurate mineral maps for a measured stone surface. The gathered QEMSCANS are listed in Table 10.

The results of the QEMSCAN of an Elefsis stone slab is shown in Figure 29. The used labeling of the mineral map is shown in Table 11. In Figure 30 the mineral maps of the same area of the stone slab "Nidaros Bad Large 01" at different measurement rounds are shown and Figure 31 shows similar results for the "Nidaros Good Large

| Geometry Scan (Round 01, 2014-06-02) | Geometry Scan (Round 02, 2015-01-12) |
|--------------------------------------|--------------------------------------|
| Elefsis_Large_01_R01_G3D.stl | Elefsis_Large_01_R02_G3D.ply |
| Elefsis_Large_02_R01_G3D.stl | Elefsis_Large_02_R02_G3D.ply |
| Elefsis_Large_03_R01_G3D.stl | Elefsis_Large_03_R02_G3D.ply |
| Elefsis_Small_01_R01_G3D.stl | Elefsis_Small_01_R02_G3D.ply |
| Elefsis_Small_02_R01_G3D.stl | |
| Elefsis_Small_03_R01_G3D.stl | |
| Elefsis_Small_04_R01_G3D.stl | |
| Nidaros_Bad_Large_01_R01_G3D.stl | Nidaros_Bad_Large_01_R02_G3D.ply |
| Nidaros_Bad_Large_02_R01_G3D.stl | Nidaros_Bad_Large_02_R02_G3D.ply |
| Nidaros_Bad_Large_03_R01_G3D.stl | Nidaros_Bad_Large_03_R02_G3D.ply |
| Nidaros_Bad_Small_01_R01_G3D.stl | Nidaros_Bad_Small_01_R02_G3D.ply |
| Nidaros_Bad_Small_02_R01_G3D.stl | |
| Nidaros_Bad_Small_03_R01_G3D.stl | |
| Nidaros_Bad_Small_04_R01_G3D.stl | |
| Nidaros_Good_Large_01_R01_G3D.stl | Nidaros_Good_Large_01_R02_G3D.ply |
| Nidaros_Good_Large_02_R01_G3D.stl | Nidaros_Good_Large_02_R02_G3D.ply |
| Nidaros_Good_Large_03_R01_G3D.stl | Nidaros_Good_Large_03_R02_G3D.ply |
| Nidaros_Good_Small_01_R01_G3D.stl | Nidaros_Good_Small_01_R02_G3D.ply |
| Nidaros_Good_Small_02_R01_G3D.stl | |
| Nidaros_Good_Small_03_R01_G3D.stl | |
| Nidaros_Good_Small_04_R01_G3D.stl | |

Table 9: Geometry Scans of the stone slabs at different stages of the accelerated erosion experiments.

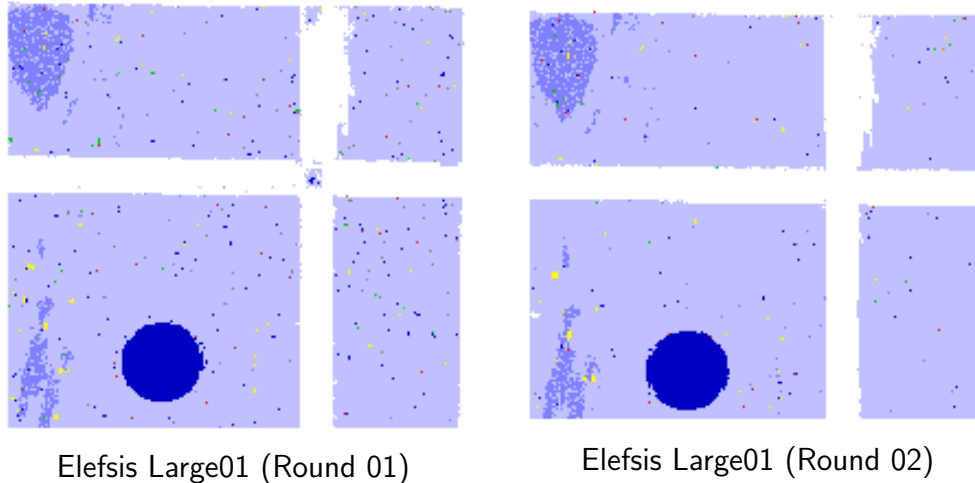
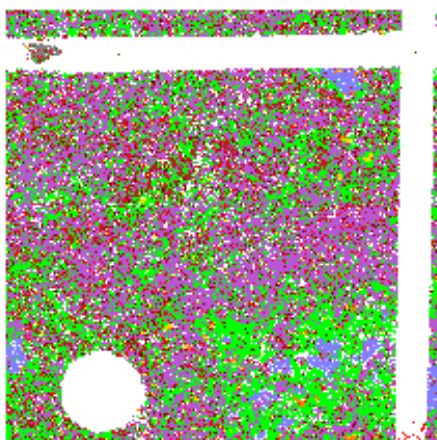


Figure 29: Mineral maps of the same area of the stone "Elefsis Large 01" at two different measurement rounds.

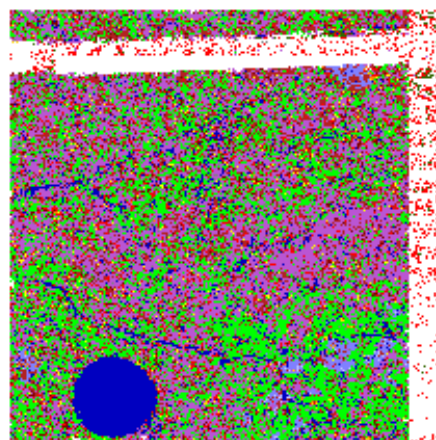
01" stone slab. The same mineral map applies for all the measured stones (cf. Table 11).

| QEMSCAN (Round 01, 2014-07-31) | QEMSCAN (Round 02, 2015-01-12) |
|--------------------------------|--------------------------------|
| Elefsis_Large_01 | Elefsis_Large_01 |
| Elefsis_Large_02 | Elefsis_Large_02 |
| Elefsis_Large_03 | Elefsis_Large_03 |
| Elefsis_Small_01 | Elefsis_Small_01 |
| Elefsis_Small_02 | |
| Elefsis_Small_03 | |
| Elefsis_Small_04 | |
| Nidaros_Bad_Large_01 | Nidaros_Bad_Large_01 |
| Nidaros_Bad_Large_02 | Nidaros_Bad_Large_02 |
| Nidaros_Bad_Large_03 | Nidaros_Bad_Large_03 |
| Nidaros_Bad_Small_01 | Nidaros_Bad_Small_01 |
| Nidaros_Bad_Small_02 | |
| Nidaros_Bad_Small_03 | |
| Nidaros_Bad_Small_04 | |
| Nidaros_Good_Large_01 | Nidaros_Good_Large_01 |
| Nidaros_Good_Large_02 | Nidaros_Good_Large_02 |
| Nidaros_Good_Large_03 | Nidaros_Good_Large_03 |
| Nidaros_Good_Small_01 | Nidaros_Good_Small_01 |
| Nidaros_Good_Small_02 | |
| Nidaros_Good_Small_03 | |
| Nidaros_Good_Small_04 | |
| SideScan_Nidaros_Bad_Small_01 | SideScan_Nidaros_Bad_Small_01 |
| SideScan_Nidaros_Bad_Small_02 | |
| SideScan_Nidaros_Bad_Small_03 | |
| ThinSection_Elefsis_01 | |
| ThinSection_Nidaros_Bad_01 | |
| ThinSection_Nidaros_Good_01 | |

Table 10: QEMSCAN data of the stone slabs.

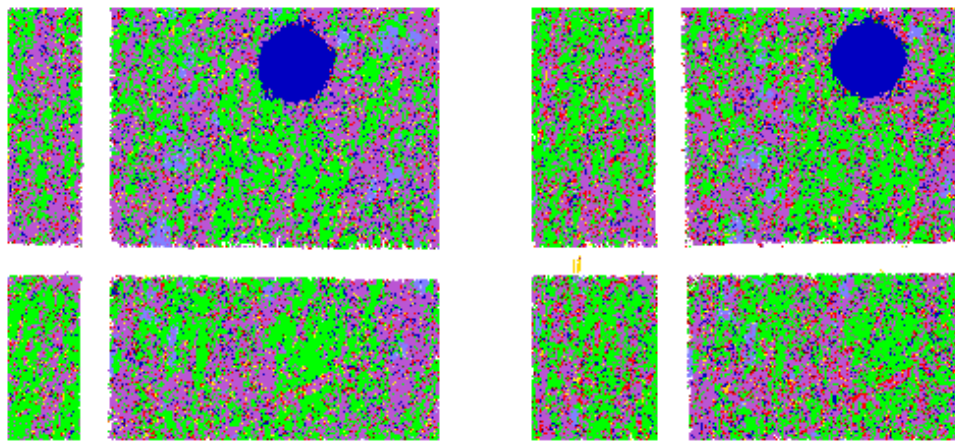


Nidaros Bad Large 01 (Round 01)



Nidaros Bad Large 01 (Round 02)

Figure 30: Mineral maps of an area of the stone slab "Nidaros Bad Large 01" at two different measurement rounds.



Nidaros Good Large 01 (Round 01)

Nidaros Good Large 01 (Round 02)

Figure 31: Mineral maps of an area of the stone "Nidaros Good Large 01" at two different measurement rounds.

| Minerals | Color | RGB-hex | | | RGB-dec | | |
|----------------------|-------|---------|----|----|---------|-----|-----|
| Quartz | | FF | FF | 00 | 255 | 255 | 0 |
| K-Feldspar | | DD | A0 | DD | 221 | 160 | 221 |
| Plagioclase Feldspar | | FF | 69 | B4 | 255 | 105 | 180 |
| Biotite | | 80 | 00 | 00 | 128 | 0 | 0 |
| Illite | | 00 | C0 | 00 | 0 | 192 | 0 |
| Chlorite | | 00 | FF | 00 | 0 | 255 | 0 |
| Smectite | | 33 | 66 | 00 | 51 | 102 | 0 |
| Kaolinite | | C0 | 40 | 00 | 192 | 64 | 0 |
| Glaucanite | | 80 | 80 | 00 | 128 | 128 | 0 |
| Calcite | | C0 | C0 | FF | 192 | 192 | 255 |
| Dolomite | | 80 | 80 | FF | 128 | 128 | 255 |
| Ankerite | | FF | C0 | 80 | 255 | 192 | 128 |
| Siderite | | FF | 80 | 00 | 255 | 128 | 0 |
| Gypsum/Anhydrite | | B2 | 22 | 22 | 178 | 34 | 34 |
| Pyrite | | FF | D7 | 00 | 255 | 215 | 0 |
| Heavy Minerals | | FF | 00 | 00 | 255 | 0 | 0 |
| Altered Mafics | | BA | 55 | D3 | 186 | 85 | 211 |
| Pores | | 00 | 00 | C0 | 0 | 0 | 192 |
| Others | | 80 | 80 | 80 | 128 | 128 | 128 |
| No measurement | | FF | FF | FF | 255 | 255 | 255 |

Table 11: Minerals that appear in the mineral maps of Figure 30, 29 and Figure 31 .

2.2.3 X-CT

Micro computed tomography is a technique similar to the well known CT scans performed at hospitals. It provides x-ray images in 3D for small scale objects with a

high resolution. It provides density information about the inner structure of the stone material and could be helpful when analyzing the 3D pore structure and volume changes of the stones. Table 12 lists the X-CT measurements that are available for the stone slabs.

| X-CT (Round 01) | X-CT (Round 02) |
|------------------------------------|------------------------------------|
| Elefsis_Large_01_20140826_R01 | |
| Elefsis_Large_01_20140918_R01 | Elefsis_Large_01_20150218_R02 |
| Elefsis_Large_02_20140826_R01 | Elefsis_Large_02_20150218_R02 |
| | Elefsis_Large_03_20150218_R02 |
| Elefsis_Small_01_20140826_R01 | Elefsis_Small_01_20150218_R02 |
| Elefsis_Small_02_20140901_R01 | |
| Elefsis_Small_03_20140901_R01 | |
| Nidaros_Bad_Large_01_20140826_R01 | Nidaros_Bad_Large_01_20150216_R02 |
| Nidaros_Bad_Large_02_20140820_R01 | Nidaros_Bad_Large_02_20150216_R02 |
| Nidaros_Bad_Large_03_20140901_R01 | Nidaros_Bad_Large_03_20150218_R02 |
| Nidaros_Bad_Small_01_20140918_R01 | Nidaros_Bad_Small_01_20150218_R02 |
| Nidaros_Bad_Small_02_20140901_R01 | |
| Nidaros_Bad_Small_03_20140901_R01 | |
| Nidaros_Good_Large_01_20140826_R01 | Nidaros_Good_Large_01_20150216_R02 |
| Nidaros_Good_Large_02_20140826_R01 | Nidaros_Good_Large_02_20150216_R02 |
| Nidaros_Good_Large_03_20140901_R01 | Nidaros_Good_Large_03_20150216_R02 |
| Nidaros_Good_Small_01_20140822_R01 | Nidaros_Good_Small_01_20150218_R02 |
| Nidaros_Good_Small_02_20140901_R01 | |
| Nidaros_Good_Small_03_20140901_R01 | |

Table 12: Micro tomography measurements that were performed on the stone slabs.

2.2.4 3D Microscopy

To allow for additional measurements on the surface of the eroded stone slabs, 3D microscopy was employed and provides textural and 3D structure of the measured stones. Illustrations for this type of measurement are given in Figure 32 where the data of three different stone surfaces is shown. A limitation of this data is that only the depth of the surface can be measured and that any concavities that might be present are ignored.

2.2.5 Petrography

Petrography is a method used since the mid 1800 for describing the mineral content and the textural relationships within rocks. A thin transparent slab slice of the stone is observed with a light microscope under plane polarised light of different directions. An example for the obtained data from the Nidaros Good stone type is shown in Figure 33.

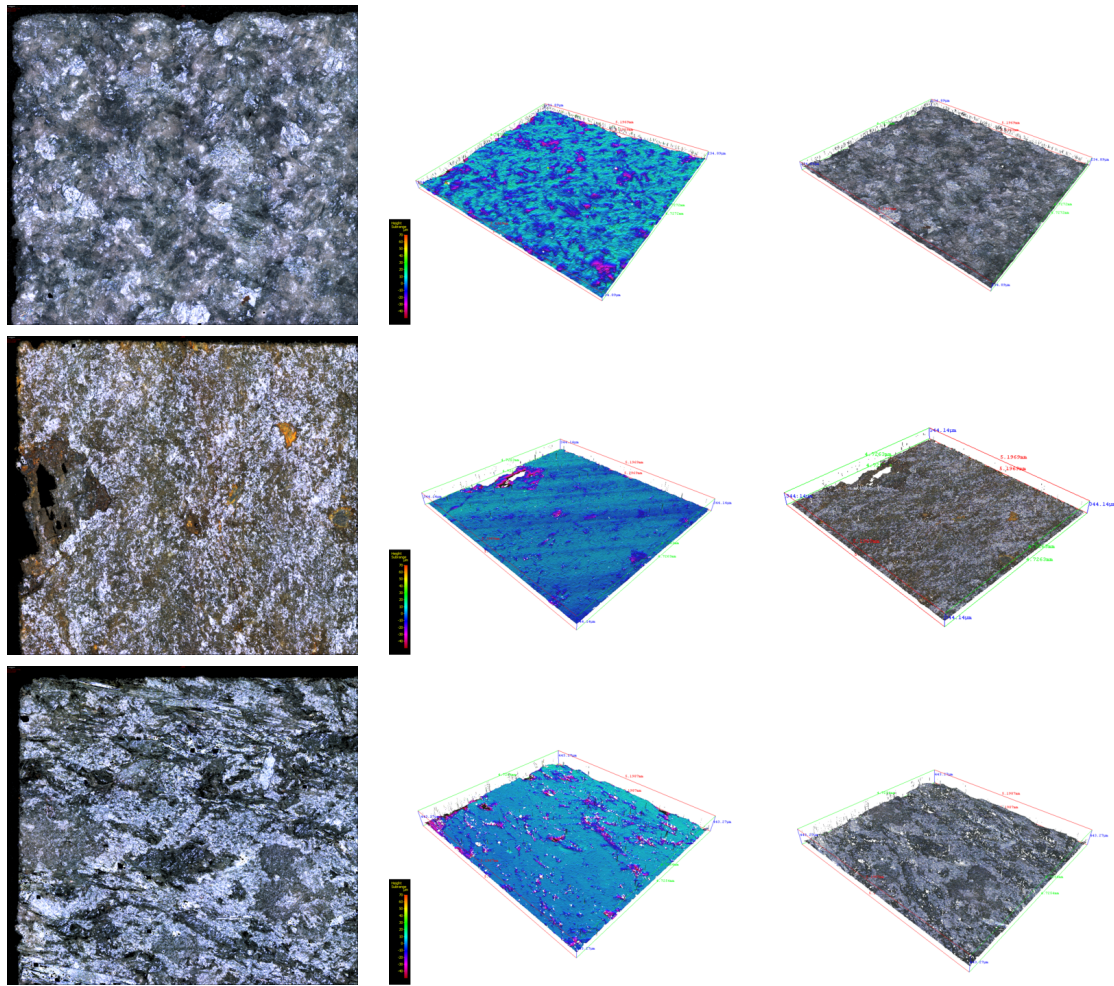


Figure 32: 3D Microscopy images provide surface geometry data along with textural information of the scanned area. Examples from the three stone types are shown: **Top row:** Elefis Large 01, **Middle row:** Nidaros Bad Large 01 **Bottom row:** Nidaros Good Large 01.

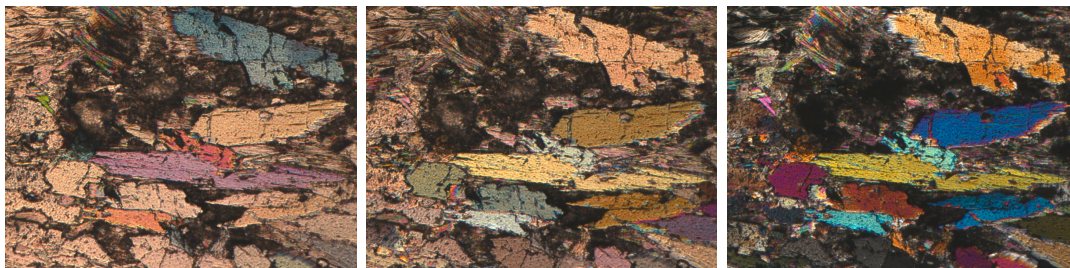


Figure 33: A thin stone slice (Nidaros Good) is illuminated with polarized light that shines through it with different orientations. Depending on the orientation of the polarized light, distinctive crystals of the stone become apparent and appear in different colors and brightness.

2.3 WP3 experiments actually performed and results

2.3.1 Estimation of amount of erosion between Erosion Cycles

A key problem in measuring erosion based on scans made across time is the difficulty in registering these scans. Due to the absence of an external reference frame, a typical registration algorithm, such as ICP, will align the scans so as to minimise the RMS error between them, which is not an ideal solution in case of erosion, see Figure 34 (a). Here is how we handled this problem in the case of the erosion chamber slabs.

We first register the top surface of the slabs using ICP and assume that this registration is sufficient in terms of the X and Y dimensions that define the top surface. The question is by how much to displace the slab in Z in order to accurately describe the erosion effect, see Figure 34 (b). Let us call this necessary displacement ΔZ . This should be equal to the computed mean erosion δ .

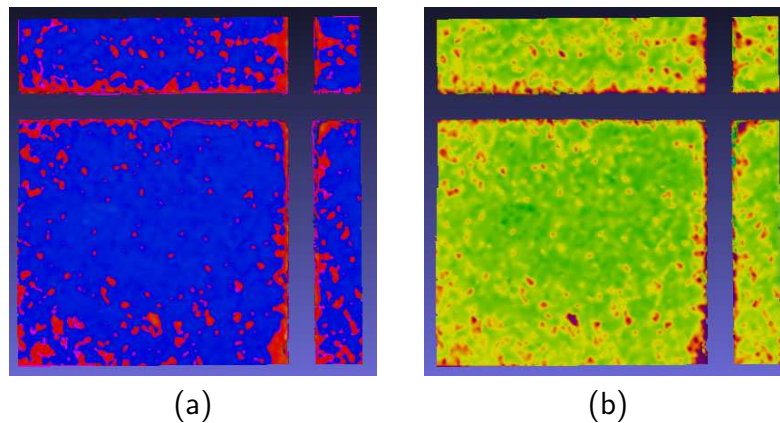


Figure 34: Differential Map of initial to eroded mesh for the frontal surface of the stone slab Elefsis Large 3 (EL3): (a) Slabs registered using ICP (blue indicates positive distances and red indicates negative distances); and (b) Slabs displaced in Z using estimated erosion value (red indicates most eroded areas and blue least eroded areas).

Estimating δ by Cubic Volume Approximation using micro-CT scans:

One way of estimating the mean erosion δ is to assume that the erosion takes place equally on all faces of the slab and that the slab can be approximated as a cube of edge h . Then $\delta = \frac{1}{2}\Delta h$.

Thus, for estimating Δh we use the volumes V_1 and V_2 of the slab before and after erosion respectively and assume that erosion took place equally in all directions and that the volume of the slab is cubical. Then Δh is computed from the slab volumes as $\Delta h = \sqrt[3]{V_2} - \sqrt[3]{V_1} = h_2 - h_1$, where h_1 and h_2 represent the cube edge lengths before and after erosion respectively. The volumes V_1 and V_2 were computed using non-void voxel counting on the micro-CT scans of the slabs.

Estimating δ by Surface Area Approximation using micro-CT scans and surface scans:

A second way of estimating δ is to use the surface areas S_1 and S_2 of the mesh before and after erosion respectively. Assuming that the surface area doesn't change too much we can use the differential equation $\Delta V = S\Delta h$, and $\delta = \Delta h = \Delta V/S$, where $\Delta V = V_2 - V_1$ and $S = S_{avg} = (S_1 + S_2)/2$. The surface areas S were computed using the summation of the triangles area of the scanned mesh. Since the back face of some slabs was not scanned before erosion, S_1 could not be directly computed and we finally assumed that $S \simeq S_2$.

Note that the above two ways of estimating $\delta = \Delta Z$ are based on different measurements (3D scans and micro-CT). We have estimated δ in both ways and the results for the various slabs are quite close to each other, confirming the reliability of the results (see Table 13).

Table 13: Comparison of the proposed erosion computation methods over Elefsis Large 3 (EL3) and Elefsis Small 1 (ES1) slabs: (a) Cubic approximation; and (b) Surface area approximation

| Mean Erosion δ (mm) | | | | | | |
|------------------------------|-----------------|-----------------|----------------------|---------------|------------------------|------------------------|
| Stone | V_1 cm^3 | V_2 cm^3 | ΔV cm^3 | S cm^2 | $\delta^{(a)}$ mm | $\delta^{(b)}$ mm |
| EL3 $H_2SO_4 + HNO_3$ | 10.2961 | 10.1391 | -0.1570 | 29.7689 | -0.0556 | -0.0527 |
| ES1 H_2SO_4 | 10.3216 | 10.1510 | -0.1706 | 29.9718 | -0.0603 | -0.0569 |

2.3.2 Experimental Results for E-GE Experiments

Experiments E-GE-1 and E-GE-2 (geometric on the Nidaros and Elefsis scans) were not performed yet for the following reasons:

- ICP registration is not sufficient. By minimising the overall registration error, it "misses" the possible erosion "common" to all points. For this reason, it may be only possible to measure relative rather than absolute erosion values since there are no external fixed reference points. We are working on alternative registration methods that may be able to exploit invariant points.
- ICP cannot be run directly on large data, such as that acquired from Nidaros and Elefsis, because it has quadratic complexity, even if the data is split in an octree data structure. We are working on overcoming this shortfall by establishing a set of the points/areas that are invariant to erosion.

We have performed the E-GE-3 experiments on the following calcium carbonate slabs / erosion chambers:

- ES1 in H_2SO_4 aqua solution erosion chamber

- EL3 in $H_2SO_4 + HNO_3$ aqua solution erosion chamber

for the following reasons:

- The Gauri erosion model, which has been implemented in the erosion simulator, is only applicable to calcium carbonate stones exposed to acid rain polluted by SO_2 and NO_2 .
- The Nidaros slabs exhibited an unexpected swallowing behaviour in both acid solution chambers; this has not yet been interpreted or modelled.
- No model is (yet) available for freeze-thaw plus rain effect or salt effect.

Furthermore, the environmental parameters of the Gauri model have not yet been related to the chemical parameters of the acid solution erosion chambers (see equation (4) of D3.4). Therefore it was not feasible to run the erosion simulator by direct application of the Gauri environmental parameters supported by the model. We thus opted for the direct application of a computed geometrical erosion offset to the mesh. This offset is computed as shown in section 2.3.1.

2.3.2.1 C-WP3-1. Erosion prediction accuracy against geometric ground-truths

Let us first define some terms:

- δ_{Gauri} represents the average amount of erosion predicted by the Gauri model; this is considered constant for all vertices in the case of a homogeneous stone.
- δ_{Exp} represents the average amount of erosion computed from the measurements taken before/after an erosion experiment (see Table 13).
- $\delta_{Gauri}(\mathbf{v})$ represents the amount of erosion predicted by the Gauri model on a vertex \mathbf{v} .
- $\delta_{Exp}(\mathbf{v})$ represents the amount of erosion computed from the measurements taken before/after an erosion experiment on a vertex \mathbf{v} .

Ideally we should evaluate two things:

- The compliance of the predicted δ_{Gauri} with δ_{Exp} .
- The compliance of the predicted $\delta_{Gauri}(\mathbf{v})$ with $\delta_{Exp}(\mathbf{v})$.

As it was not practical to run the Gauri model (see equation (4) of D3.4) for the reason outlined above, we opted for measuring the compliance of δ_{Exp} with $\delta_{Exp}(\mathbf{v})$, which essentially evaluates the compliance of the geometric part of the erosion model (see equation (1) of D3.4). This was performed by applying the constant value δ_{Exp} to all vertices and comparing it with the per-vertex values $\delta_{Exp}(\mathbf{v})$.

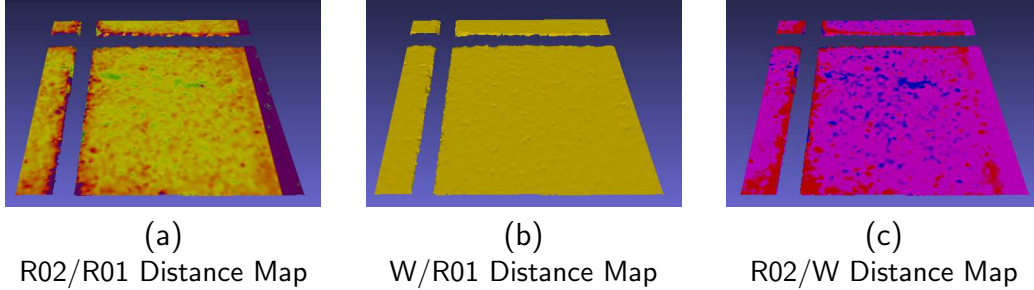


Figure 35: Distance maps between different meshes of the stone slab Elefsis Small 1 (ES1): (a) Distance of Round 1 mesh (R01) from Round 2 mesh (R02) mapped in the interval of $[-0.1mm \sim +0.1mm]$; (b) Distance of Erosion prediction mesh (W) from Round 1 mesh (R01) mapped in the interval of $[-0.1mm \sim +0.1mm]$; and (c) Distance of Round 2 mesh (R02) from Erosion prediction mesh (W) mapped in relative scale (red under-estimation/blue over-estimation).

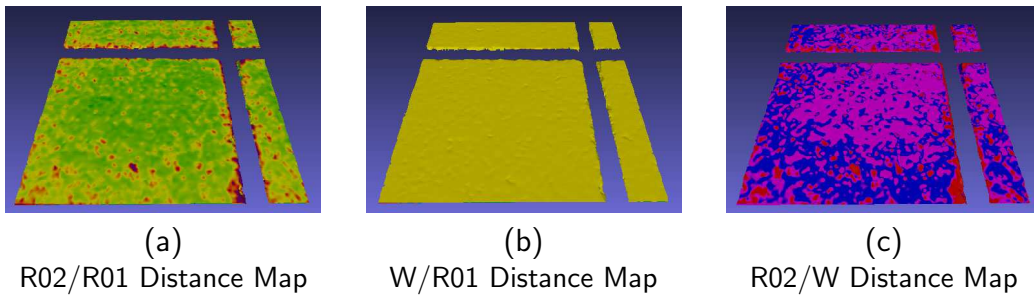


Figure 36: Distance maps between different meshes of the stone slab Elefsis Large 3 (EL3): (a) Distance of Round 1 mesh (R01) from Round 2 mesh (R02) mapped in the interval of $[-0.1mm \sim +0.1mm]$; (b) Distance of Erosion prediction mesh (W) from Round 1 mesh (R01) mapped in the interval of $[-0.1mm \sim +0.1mm]$; and (c) Distance of Round 2 mesh (R02) from Erosion prediction mesh (W) mapped in relative scale (red under-estimation/blue over-estimation).

Hausdorff distance as a metric of stone erosion:

Consider two point sets:

$$M = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_p\}$$

that represents the initial surface of a stone, and

$$T = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_q\}$$

that represents the weathered surface of the same stone, where $\mathbf{m}_i, \mathbf{t}_j \in \mathbb{R}^3$.

The *standard Hausdorff* distance is defined as:

$$D_H(M, T) = \max(D_h(M, T), D_h(T, M)) ,$$

where

$$D_h(M, T) = \max_i (\min_j (\|\mathbf{m}_i - \mathbf{t}_j\|)) ,$$

is the *directed Hausdorff* distance from M to T .

The directed Hausdorff distance expresses the Euclidean distance $\|\mathbf{m}_i - \mathbf{t}_j\|$ of the farthest point of M from any point of T , i.e., the maximum value of the minimum Euclidean distances of the points of M from any point of T .

The *average directed Hausdorff distance* D_{MH} , of an initial stone model M to an eroded stone model T , can be defined, as:

$$D_{MH}(M, T) = \frac{1}{p} \sum_{i=1}^p \min_j (\|\mathbf{m}_i - \mathbf{t}_j\|) , \quad (1)$$

where $\|\mathbf{m}_i - \mathbf{t}_j\|$ is the Euclidian distance between the initial model vertices \mathbf{m}_i and the eroded model vertices \mathbf{t}_j , and p the number of the model vertices. D_{MH} expresses the mean value of the minimum Euclidean distances $\|\mathbf{m}_i - \mathbf{t}_j\|$ of the points of M from any point of T .

The average directed Hausdorff distance D_{MH} can be used as an overall mean erosion measure for the whole stone or a portion of it.

The distance $d_e(\mathbf{m}_i) = \min_j (\|\mathbf{m}_i - \mathbf{t}_j\|)$ can be used as a local erosion measure which expresses at each vertex of the initial model M the distance of the closest vertex of the eroded model T , and is a scalar mapping of the erosion measure at each vertex of the initial stone model M , to which the eroded model T is registered.

Figures 35 and 36 depict the distance maps (i.e the $d_e(\mathbf{m}_i)$) between various meshes of Elefsis Small 1 and Elefsis Large 3, and consequently the experimental erosion measure (Figures 35 (a) and 36 (a)), the predicted erosion measure (Figures 35 (b) and 36 (b)), and the prediction error (Figures 35 (c) and 36 (c)) mapped on the initial meshes.

From the above, the following conclusions can be drawn:

- The simulated erosion is smaller than the observed erosion close to the edges of the mesh.
- The overall geometric erosion prediction is close to the experimentally measured values.

2.3.2.2 C-WP3-3. Erosion prediction geometric robustness The scanned 3D meshes of the slabs include holes and missing data (e.g. cross groove in Figure 37), noise and spikes (no preprocessing has taken place). Despite these, the simulator runs smoothly.

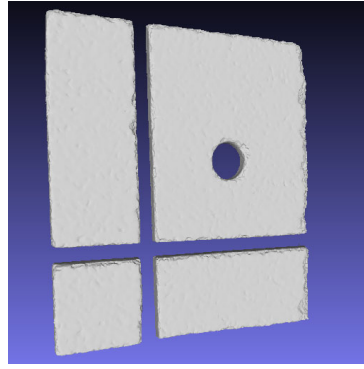


Figure 37: Surface mesh exhibiting holes and missing areas.

2.3.2.3 C-WP3-5. Erosion prediction tolerance against object geometric resolution and scale variations Two issues are involved here. First, whether the simulator runs smoothly on meshes digitised at different resolutions and, second, the sensitivity of the results to changes in mesh resolution.

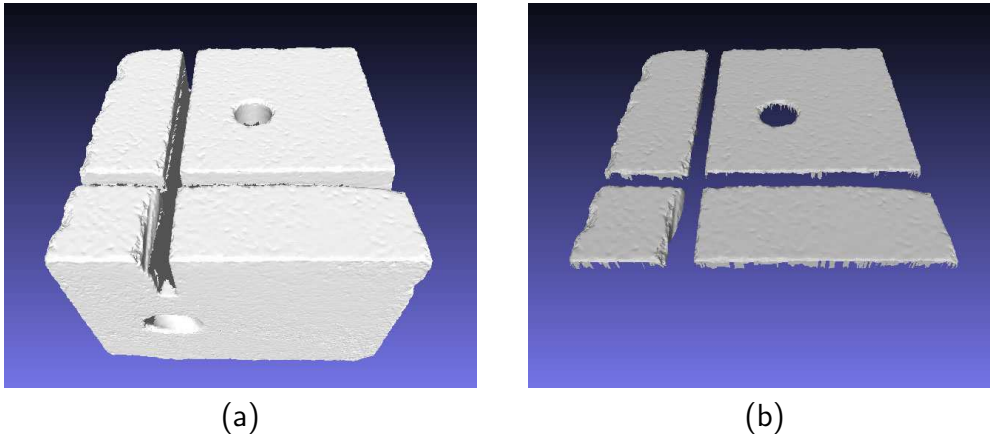


Figure 38: Depiction of mesh of the stone slab Elefsis Large 1 (EL1): (a) the complete scanned irregular mesh; and (b) the regularly resampled frontal area.

For the first case, the simulator was run on various complete irregular meshes acquired at different resolutions with mean-edge-length at $0.060 \sim 0.098\text{mm}$ (see Figures 38 (a) and 39 (a)) and on various regular meshes re-sampled at different resolutions with mean-edge-length at $0.035 \sim 0.050\text{mm}$ (see Figures 38 (b) and 39 (b)) without any problems.

For the second case, we evaluated the sensitivity of the results to changes in the mesh resolution; to this end, we regularly re-sampled the frontal meshes at two different resolutions (Figures 39 (a) and 39 (b)). The result of the erosion applied on a 1024×1024 regularly sampled mesh (mean-edge-length at 0.035mm) and 256×256 regularly sampled mesh (mean-edge-length at 0.140mm) is depicted in Figure 40 as a colored difference map.

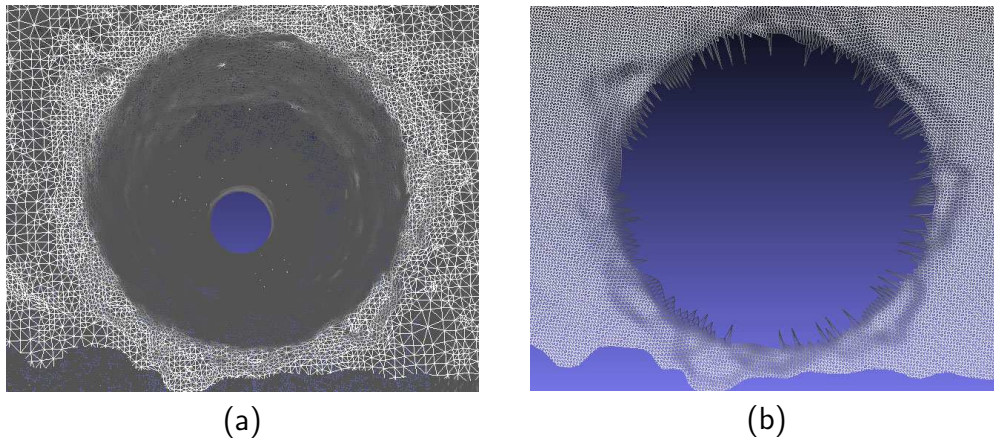


Figure 39: Detail of the mesh of the stone slab Elefsis Large 1 (EL1) near the hole area: (a) the initial irregular mesh; and (b) the regularly resampled area.

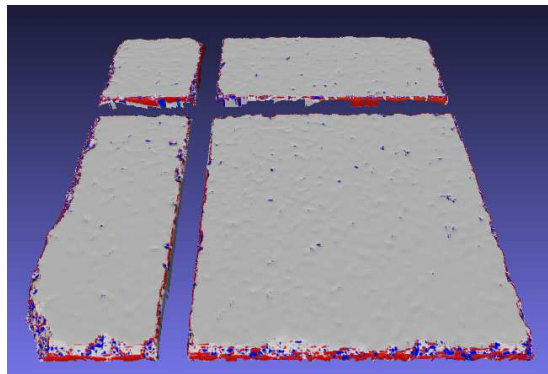


Figure 40: Difference map between the eroded meshes at 1024×1024 sample vertices and 256×256 sample vertices (blue under-estimation/red over-estimation).

2.3.2.4 C-WP3-7. Erosion prediction procedure complexity and absolute processing time evaluation The complexity of the current version of the erosion simulator is linear in the number of vertices of the mesh being eroded. Some typical running times on an Alienware computer (Intel Core i7 @ 3.6 GHz, 16 GB RAM) are given in Table 14.

Table 14: Comparison of the proposed erosion computation running times over Elefsis Large 3 (EL3) and Elefsis Small 1 (ES1) slabs: (a) on the regularly resampled frontal area; and (b) on the whole scanned irregular mesh

| Stone | Running times | | |
|--------------------------|---------------|----------|---------|
| | facets | vertices | t (sec) |
| EL3^(a) | 295,643 | 149,451 | 3.01 |
| EL3^(b) | 1,960,187 | 983,698 | 18.38 |
| ES1^(a) | 278,244 | 140,690 | 2.77 |
| ES1^(b) | 1,296,290 | 652,069 | 12.18 |

The current version of the erosion simulator is therefore near real-time for relatively small meshes, allowing the user to experiment with various scenarios.

2.3.3 Experimental Results for E-GESC Experiments

These involve surface physicochemical data in addition to geometric information. Due to late arrival of the physicochemical data measurements on the slabs, calibration and evaluation of the erosion model using such data has not yet been performed.

2.4 Deviations and corrections with respect to WP3 evaluation plan

The experimental nature of WP3 made the Evaluation Plan (D5.1) look overly optimistic for this WP.

In the course of the second year it became apparent that the differential erosion scans at the Elefsis and Nidaros archaeological sites would be insufficient for model calibration and evaluation purposes, as:

- The problem of registration has not yet been solved in a satisfactory manner (ideally with external reference points) and the little suspected amount of erosion can therefore not yet be reliably estimated. Specifically:
 - Simple ICP registration is not sufficient. By minimising the overall registration error, it "misses" the possible erosion "common" to all points.
 - ICP cannot be run directly on large data, such as that acquired from Nidaros and Elefsis, because it has quadratic complexity.
- No surface chemical data of the eroding surfaces can be provided.

Thus experiments E-GE-1 and E-GE-2 (geometric on the Nidaros and Elefsis scans) were not performed yet, pending solutions to the above problems.

It thus became clear that we should also employ Erosion Chambers. As erosion is a complex process involving many factors, it was decided to build 4 chambers in order to study 4 important factors separately. Although the joint effect of the combination of factors is not guaranteed to be their sum and ideally one should employ erosion chambers with combinations of factors, the approach that we decided upon seemed most practical given our time and financial constraints. For more details on the erosion chambers see D1.2 (Second Annual Report). We initially considered outsourcing the erosion chambers to Sintef, but this approach was ruled out due to suitability and cost considerations. The chambers were thus built in-house as described in D1.2. This process took longer than anticipated and the 4 chambers became operational between

19 September 2014 and 10 November 2014. All chambers were stopped by 4.12.2014 in order to get the second round of measurements in time for the Evaluation Report. The measurement phase ended on 20.2.2015.

In order to potentially study non-linear erosion behaviour, we decided to have 4 one-month long erosion cycles interspersed with 5 measurement points. These cycles are expected to be completed in summer 2015. We also decided upon 6 measurement modalities (geometrical and chemical). However, there was insufficient budget for the measurements, some of which have to be outsourced. The Grant Agreement therefore had to be changed accordingly to reflect this by diverting money originally planned for travel.

The E-GESC experiments (see D5.1 Evaluation Plan) were delayed due to the late arrival of the second round of measurements as mentioned above. However we tried to make the best use possible of the data.

E-GEVC experiments (see D5.1 Evaluation Plan) could not be run as no measuring modality gives sufficient physicochemical data within the volume of the stone slabs. It was hoped that the density data provided by micro-CT could somehow be mappable into volumetric chemical data; unfortunately this is not feasible as there is no 1-1 mapping between density data and chemical composition. It further seems that the densities of the minerals involved are rather homogeneous. The E-GEVC evaluation experiments therefore need to be cancelled.

2.5 WP3 items remaining to be evaluated in D5.8

E-GE1 and E-GE-2 We are still trying to solve the registration problem of the differential scan data in a way that can give us useful information. This data is useful as it represents erosion in real conditions. Avenues we are still exploring are:

- The measurement of relative erosion, i.e. which parts of a surface may have eroded more;
- The use of unconventional distance functions in the registration algorithm, in an attempt to achieve registration based on features such as curvature;
- The manual selection of landmarks that are assumed to have remained invariant and their use to register the surfaces.

E-GE-3 We are aiming to expand this experiment to more effects (freeze-thaw) and stone types (soapstone), if the erosion simulator can be expanded to cover these and as more datapoints become available from the erosion chambers. **E-GESC-2** This part of the evaluation will be performed incrementally as more data points become available from the erosion chambers. **E-GESC-1** This part of the evaluation will

Table 15: Updated WP3 Evaluation Table; ● : done ● : postponed to D5.8
● : cancelled

| | C-WP3-1 | C-WP3-2 | C-WP3-3 | C-WP3-4 | C-WP3-5 | C-WP3-6 | C-WP3-7 | C-WP3-8 |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| E-GE1 | ● | | ● | | ● | | ● | |
| E-GE2 | ● | | ● | | ● | | ● | |
| E-GE3 | ●● | | ●● | | ●● | | ●● | |
| E-GESC1 | ● | ● | ● | ● | ● | ● | ● | ● |
| E-GESC2 | ● | ● | ● | ● | ● | ● | ● | ● |
| E-GEVC1 | ● | ● | | ● | ● | ● | ● | ● |
| E-GEVC2 | ● | ● | | ● | ● | ● | ● | ● |

be performed incrementally as more data points become available from the erosion chambers, assuming that the erosion model can be extended to handle soapstone.

Table 15 summarizes our modified evaluation plan for WP3, based on the original WP3 table.

3 Evaluation of WP4

According to the Evaluation Plan D5.1, the evaluation in WP4 is task specific. Therefore, each task defines its evaluation protocols following the guidelines described in D5.1. Section 3.1 presents the experiments and evaluation related to T4.1 - Object Reassembly. Section 3.2 describes the experiments and evaluation related to T4.2 - Object Repair.

3.1 Reassembly Experiments

In this section we cover the evaluation of the reassembly algorithm, taking into account all of the evaluation criteria described in Evaluation Plan D5.1. For convenience, the mapping between successively more complex experiments and criteria, as detailed in D5.1, is repeated below, in Table 23.

| | General Effectiveness | General Efficiency | Scalability | Discrimination | Robustness | Plausibility | User Input |
|---|-----------------------|--------------------|-------------|----------------|------------|--------------|------------|
| Pairwise Matching | • | • | • | • | • | | |
| Reassembly of a Single Object | • | • | • | • | • | • | • |
| Reassembly of Mixed Objects | • | • | • | • | • | • | • |
| Reassembly of Heavily Damaged Fragments | • | | • | • | • | • | • |

Table 16: Criteria of the reassembly experiments

The pre-processing steps required by our algorithms (*segmentation*, *classification* and *feature curve extraction*) are omitted from this report, mainly because they represent a strictly off-line, preprocessing step, despite the fact that for sake of completeness and overall system cohesion and practicality, VRMW integrates the preprocessing of *new* meshes and allows corrective *re-evaluation* of the segmentation and classification from within the interface. For more information on these stages please refer to D4.2 First Object Reassembly and Repair and D1.2 Periodic Report of Year 2 (feature curve extraction). All experiments were conducted on an Intel Core i7-3820 @ 3.6GHz with 16GB of RAM, running Windows 7 Professional 64bit.

3.1.1 Pairwise Matching Experiments

In this section we focus on the core techniques used for the alignment of two fragments, both in the case when the state of the fragments allows for a contact-based alignment and the case when fractured surfaces are unusable and/or when there is a significant amount of material missing between the pieces. In the first case, there exists sufficient compatible area on the fractured surfaces of the two fragments to lead to a valid alignment and registration is based on the geometry of the fractured facets. In the latter case, the registration is assisted or even entirely guided by feature curves detected on the intact surfaces of the fragments, depending on the compatibility of the fractured facets. This process is explained in detail in D1.2.

3.1.1.1 General Effectiveness In order to evaluate the pairwise matching algorithm we performed numerous tests, an indicative sub-set of which is presented here. Most of the data used are from the archaeological site of Nidaros Cathedral for which and since we could not compare the results to a reference model, we evaluate the quality of the resulting experiments by visual inspection (here we present images). For quantitative analysis we measure the matching score of our objective function (see D1.2) and the penetration between the fragments, for which we provide the measurement procedure below. In order to perform an even better qualitative evaluation of our method, we performed experiments with non-archaeological models, which we also scanned before fracturing in order to have a real reference model (see full datasets in deliverable D5.5 - Pilot Acquisition Data).

Penetration Measurement. To measure a meaningful penetration value, we generate the intersection of the two fragments $f_i \cap f_j$, using constructive solid geometry operations. However, the volume of $f_i \cap f_j$ alone as a quantity is not very descriptive, as it depends on the actual size of the models. On the other hand, the ratio of the intersection volume $f_i \cap f_j$ against the fragments' volume is also misleading, as thin and sizable penetrating segments would report different values. Instead, we measure the area of the projection of volume $f_i \cap f_j$ on a plane P , roughly aligned with the fractured surfaces. The final penetration score we report is the *intersection volume to area ratio*, which accounts for the *mean penetration distance*. P is generated by using PCA analysis of the covered fractured point set (of both fragments) due to the alignment (see Figure 41).

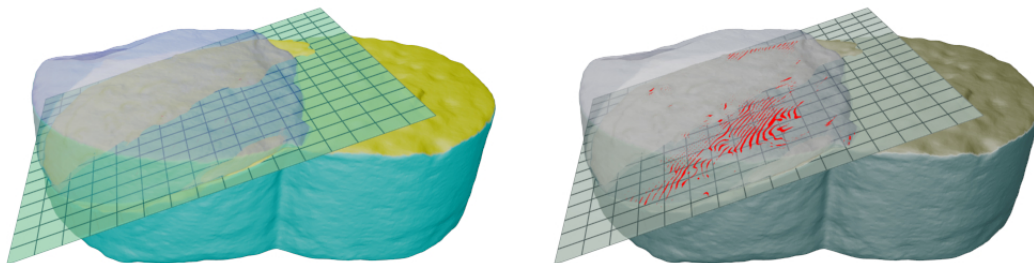


Figure 41: On the left, the plane on which the penetrating volume $f_i \cap f_j$ is projected in order to measure the covered area. On the right, the penetrating volume is shown in red.

Visual Alignment Assessment. Capturing alignment performance for the real archaeological datasets relies on visual inspection, as there are no ground-truth data. Nevertheless, in most cases, inspection of the resulting 3D models of the aligned fragments could easily indicate even slight misalignments, especially when the pairs were inspected as part of a larger fragment cluster, due to the propagation of a potential error to the entire chain of connected parts.

Quantitative Alignment Assessment. To validate our alignment method in a more concise and measurable manner, as explained in the introduction, we performed experiments with non-archaeological models, which we scanned before fracturing. Using the scanned original as a reference model, we generated *reference reassemblies* and measured their RMS *Hausdorff distance* to our resulting reassemblies.

The reasons we measure the RMS from the reference reassembly and not the initial reference model are three-fold. During the fracturing procedure, parts of the object are fractured into tiny pieces that cannot be digitized and would impose an error in the measurements. Furthermore the scanning process of objects introduces errors in the digitized models and thus there is a deviation between the complete model and its fragments that cannot be measured. This error is typically manifested both as jittering of the resulting vertices but also as a slight thickening of the surface after meshing due to the surface reconstruction algorithm. Finally if we were to measure the deviation between a reassembly and the complete object we would have to exclude the fractured facets from the measurements, which would require a perfect segmentation of the fragments, something that is neither trivial to achieve or desired.

Instead we follow a different approach that solves these problems: Initially we register the fragments on the surface of the intact scanned object, using both our registration algorithms and manual constraints, where required (especially for featureless surfaces and symmetrical objects). In both cases, the use of Sparse ICP guarantees the convergence of the registration. The fragments individually registered with the original object comprise the *reference reassembly*. We evaluate the results of our reassembly algorithm by aligning one fragment of the reassembly with the corresponding piece in the reference reassembly and relatively transforming the rest of the reassembled fragments. The deviation of the reassembled fragments from the reference reassembly gives us reliably the quality of the achieved reassembly. An additional benefit from this test procedure is that since we test a fragment against the *same mesh* in a different alignment configuration, there are no deviations introduced due to different meshing of the surfaces, as the compared geometries are piece-wise identical, increasing the reliability of our measurements.

Our experiments are split in two categories. In the first we showcase pairs of fragments with good matching area and minor erosion or missing parts. These are considered the easy cases for our algorithm due to its objective function (minimization of distance between the tested facets). In the second category we show fragment pairs, where facets are not perfectly matched and the use of the ℓ_p -norm in order to achieve good alignment.

Pairs with good matching area. Here we test our pair-wise alignment algorithm with fragments that share good contact areas (Figure 42). In such cases the expectation is to achieve good performance and quality with minimal user interaction. We performed the experiments using a balanced set of parameters, 5000 iterations and 4 restarts in the ESA optimizer using a median norm and ICP refinement as the last step.

From the results in Figure 42 we can conclude that the alignment quality and efficiency of our method is very good with minimal penetration (below the accuracy of the scanner used to obtain the scans). While settings that achieve better performance


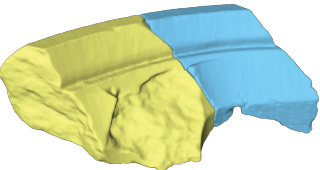
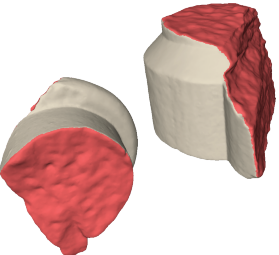
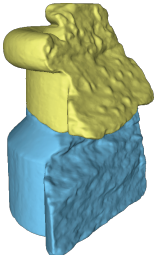
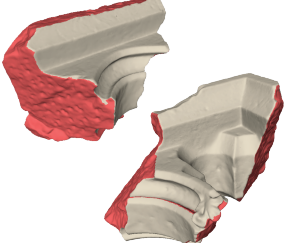
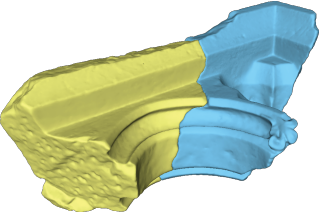
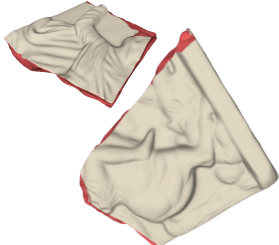
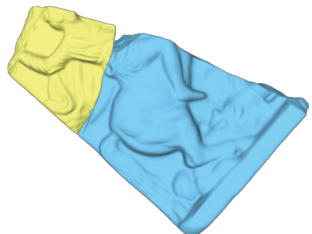
| | Fragments | Pairwise Alignment | Evaluation |
|----------------|---|---|--|
| DoraArch |  |  | ESA : 3.08 sec ICP : 0.077 sec Penetration : 0.32 mm Matching Error : 0.64 I _p -norm : - Pair-Tests : 9 Hausdorff RMS : - |
| DoraColumnBase |  |  | ESA : 4.4 sec ICP : 0.042 sec Penetration : 0.11 mm Matching Error : 0.75 I _p -norm : - Pair-Tests : 12 Hausdorff RMS : - |
| DoraEmbrasure |  |  | ESA : 4.32 sec ICP : 0.04 sec Penetration : 0.35 mm Matching Error : 0.81 I _p -norm : - Pair-Tests : 20 Hausdorff RMS : - |
| Metope1 |  |  | ESA : 1.5 sec ICP : 0.047 sec Penetration : 0.06 mm Matching Error : 0.44 I _p -norm : - Pair-Tests : 6 Hausdorff RMS : 0.75 |

Figure 42: Pairwise alignments for fragments with low deterioration and usable contact area. All the pairwise combinations are explored and the best one is refined using ICP. We report the time spent in the ESA optimization and the ICP refinement, the matching score, whether p-norm is utilized, the total number of pair-wise tests and the Hausdorff distance RMS from the ground truth where possible.

exist, we prefer a more conservative approach that guaranties that the need for the user to perform any parameter readjustment is kept to a minimum.

Eroded Pieces. In Figure 43 we present pairs of fragments that have eroded or chipped matching facets. While an exhaustive set of parameters could lead to the correct alignment eventually, that would be impractical. Instead in such cases, the use of a robust distance metric is preferred, in order to obtain the correct alignment. The use of ℓ_p -norm can solve alignment problems with significant outliers (either by erosion or due to slippage of surfaces). All the results shown were obtained using the same parameters, 5000 iterations and 4 restarts in the ESA optimizer using the ℓ_p -norm with $p = 0.2$, which imposes a small performance impact on the ESA optimization step. Furthermore we apply a 5% trimming of the largest residuals in order to reduce the outliers. The remaining outliers are suppressed by the ℓ_p -norm.

| | Fragments | Pairwise Alignment | Evaluation |
|----------------------|-----------|--------------------|--|
| DoraColumnFoundation | | | ESA : 0.97 sec (0.71) ICP : 0.91 sec Penetration : 0.15 mm Matching Error : 1.24 (2.29) ℓ_p -norm : 0.2 Pair-Tests : 1 Hausdorff RMS : - |
| Tombstone | | | ESA : 13.3 sec (9.74) ICP : 3.1 sec Penetration : 0.25 mm Matching Error : 1.09 (2.6) ℓ_p -norm : 0.2 Pair-Tests : 8 Hausdorff RMS - |
| Lionhead1 | | | ESA : 1.7 sec (1.06) ICP : 0.11 sec Penetration : 0.12 mm Matching Error : 0.9 (1.42) ℓ_p -norm : 0.2 Pair-Tests : 6 Hausdorff RMS : 0.58 |

Figure 43: Pairwise alignments for eroded fragments. The highlighted is the one where the ℓ_p -norm is used. All the pairwise combinations are explored and the best one is refined using ICP. We report the ESA stage time and in parentheses the equivalent time without the robust metric. The time spent in the ICP refinement of the best match, the matching score (in parenthesis the score with the median norm), the value of p, the total number of pair-wise tests, whether feature curves were used and the Hausdorff distance RMS from the ground truth, where available.

3.1.1.2 General Efficiency.

Performance. The optimization step that solves the pairwise alignment is highly non-convex and non-smooth and the optimizer requires a large number of iterations in order to achieve a good approximation of the global optimum. Critical to the efficiency of this process is the cost of distant queries. As detailed in D1.1, our method utilizes *Signed Distance Fields* (SDF) in order to speed-up the most time-consuming part of the algorithm. In Figure 44 we showcase the performance gain in comparison to the *Kd-Tree* and *Approximate Nearest Neighbors* (ANN) data structures. As this test indicates, SDFs perform on average $3.4\times$ faster than the ANN and $21\times$ faster than the Kd-Tree. This result is consistently validated throughout all of our experiments and is due to the complexity of the search queries (see scalability below).

The generation of the SDF is done the first time a fragment participates in a pairwise alignment and stored for future re-use. If a different resolution or clamp distance for the discretization is requested, the SDF is re-evaluated. The cost of the SDF creation depends on the desired resolution, but is between 0.5 and 3.0 sec, the latter for the highest resolution ever used in our experiments. Furthermore, this minimal cost is amortized among subsequent pairwise runs and is therefore negligible.

Indicative cumulative timings for the pairwise registration experiment are shown in Figure 45. The first stage in our alignment pipeline uses a RANSAC-based algorithm to provide a coarse initial alignment of the input surfaces. However, this stage is only triggered when the area of the two fractured facets differs more than 30%, i.e. when we expect a sliding effect of one against the other due to very incompatible size. In the rest of the cases, the two facets are roughly aligned using their centroids and average fracture normals. In the figure examples, only Pair1 and Pair5 enable the RANSAC stage. Pair3 has two small surfaces, resulting in less accurate alignment in the SA stage due to a small area of support. Consequently, the ICP stage requires more iterations to converge to the desired error margin.

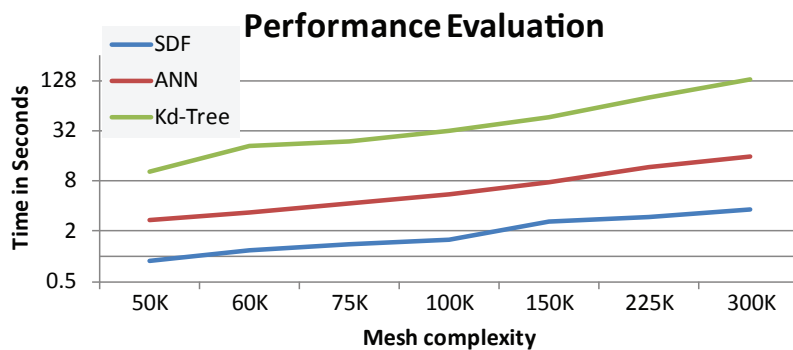


Figure 44: Running time of a single facet pair alignment for: *Signed Distance Fields*, *ANN* and *Kd-Tree* data structures in relation to the geometric complexity of the fragments. The fragments are decimated in different resolutions and the same single facet from each one takes place in the alignment. The graphs are in log scale, since timings are an order of magnitude apart, even for moderately large meshes.

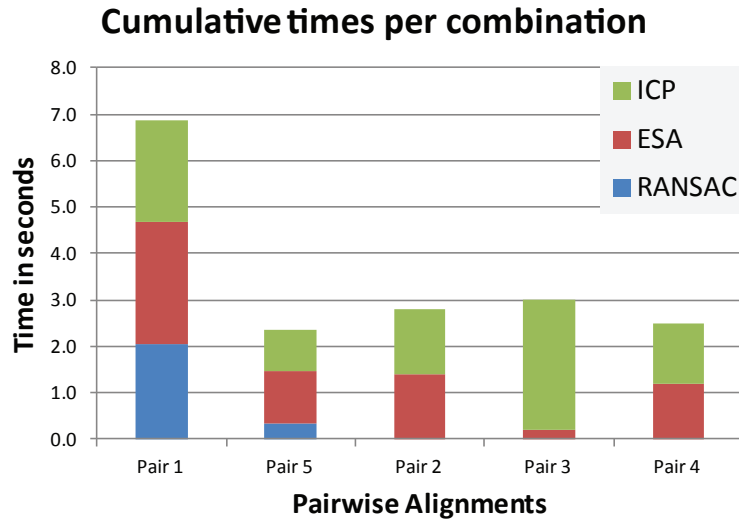


Figure 45: Cumulative timing of pairwise matches for fragments with a diverse set of fractured facets. Pair1 consists of facets with a significantly larger number of samples than the rest of the combinations. Also notice that the RANSAC stage is only activated for Pair1 and Pair5.

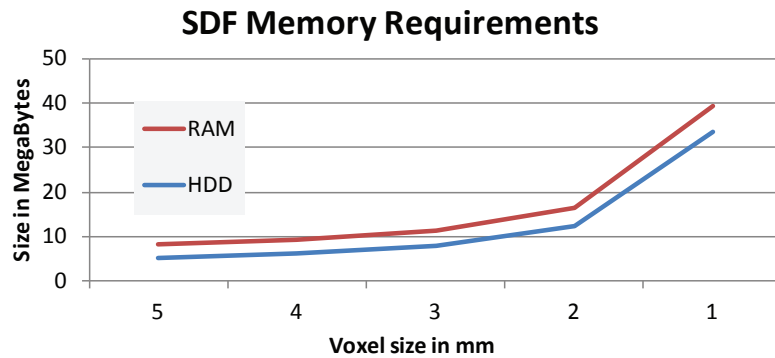


Figure 46: Memory requirements of the *SDF* data structure both in RAM and HDD in relation to their resolutions.

SDF memory Impact. The use of SDFs gives us a performance gain, but on the other hand it introduces additional storage requirements both in memory and persistent storage (SDFs are computed only the first time a fragment is used and are permanently stored for re-use). In Figure 46 we show the storage impact with respect to voxel granularity. Since the SDFs are used only during the ESA optimization and the result is refined using ICP and accurate queries, in most of our experiments we use a 2mm voxel size, without any impact on the final alignment.

3.1.1.3 Scalability. In order to reassemble a fractured object, all pairs of fractured facets belonging to distinct fragments must be explored. Therefore, for two fragments f_i, f_j , with N_i and N_j facets detected as fractured on each one respec-

tively, a total $N_i \times N_j$ combinations must be explored. However, not the entire 3-stage registration pipeline is executed for all combinations. Instead, *the final registration is executed only for the best candidate facet pair.*

The complexity of the alignment depends linearly on the maximum number of iterations allowed in the RANSAC and SA stages. In most of our experiments though, we fix these parameters to a reliable setting, despite the fact that for many fragment pairs fewer iterations are practically required.

As often noted in the bibliography, key to the success of a RANSAC-based method is the selection of wide bases, which are known to be less sensitive to noise. The width of a base is bounded by the diameter of the fracture. In our approach, we select wide bases consisting of three-point congruent sets. Given a randomly selected wide base consisting of three points $\mathbf{v}_i, i = 1, 2, 3$ on the source surface, we extract at most k congruent bases from the target surface, where k is a user-defined parameter. To this end, we first extract k random point pairs on the target surface whose distance is approximately $\|\mathbf{v}_2 - \mathbf{v}_1\|_2$. For this operation we exhaustively inspect all possible point pairs, therefore it has a complexity of $O(n^2)$, n being the number of points in the cloud. Then for each one of the k extracted pairs, we create a base by searching for a third point that forms a triangle, which is approximately congruent with the source one. For the k point-pairs, the cost of this operation is $O(kn)$, therefore, the total cost for the selection of k congruent bases is $O(n^2 + kn)$. Then, a candidate transformation is computed and validated for each one of the extracted bases.

In the Simulated Annealing stage, each cost function evaluation invokes the a point-to-surface distance query multiple times (once per point sample). Obviously, as shown in Figure 44, SDFs scale better than the other data structures, since the complexity of the queries is $O(1)$, while for both the ANN and k-d tree, the cost of the query increases as the search data structure height grows. Additionally, since the SDFs offer constant time queries, neither the granularity of the discretization has an impact on the method's performance. Obviously, the time for all distance evaluation methods increases proportionally to the number of point samples on one fragment for which we query their distance to the other surface.

3.1.1.4 Robustness

Initial Alignment. We show in the presented results that our methodology is robust and in contrast to other approaches that depend on an initial alignment, no assumptions were made about the fragment's initial position, as for each pair test the initial alignment was generated algorithmically (using either PCA or RANSAC).

Input Geometry Type. Furthermore, the reassembly method is robust with regard to the input representation and can operate both on mesh geometries and point-clouds with or without normal vectors, as they are generated on the fly. Additionally,

there is no restriction with regard to the shapes participating in the registration being manifold, a useful characteristic, when the entire fragment cannot be scanned (see for example the non-public Larnaka data set in D1.2 report).

Scale & Resolution Invariance. Our optimization step is scale-invariant and this is showcased in the presented experiments, where small and large fragments are used in reassemblies, using the same parameters. On the other hand, the matching error used in our experiments depends on the actual units of the model and one should not mix and compare the matching errors of fragment-pairs expressed in different units. This is logical, since the matching error involves distance measurements, which in order to work well in mixed puzzles, must not be normalized.

3.1.1.5 Discrimination. The discrimination ability of our matching score can be evaluated from Figure 47, where we see that the correct alignment achieves the lowest matching error among all pairwise facet tests, with a wide percentile difference from the next candidate solution.

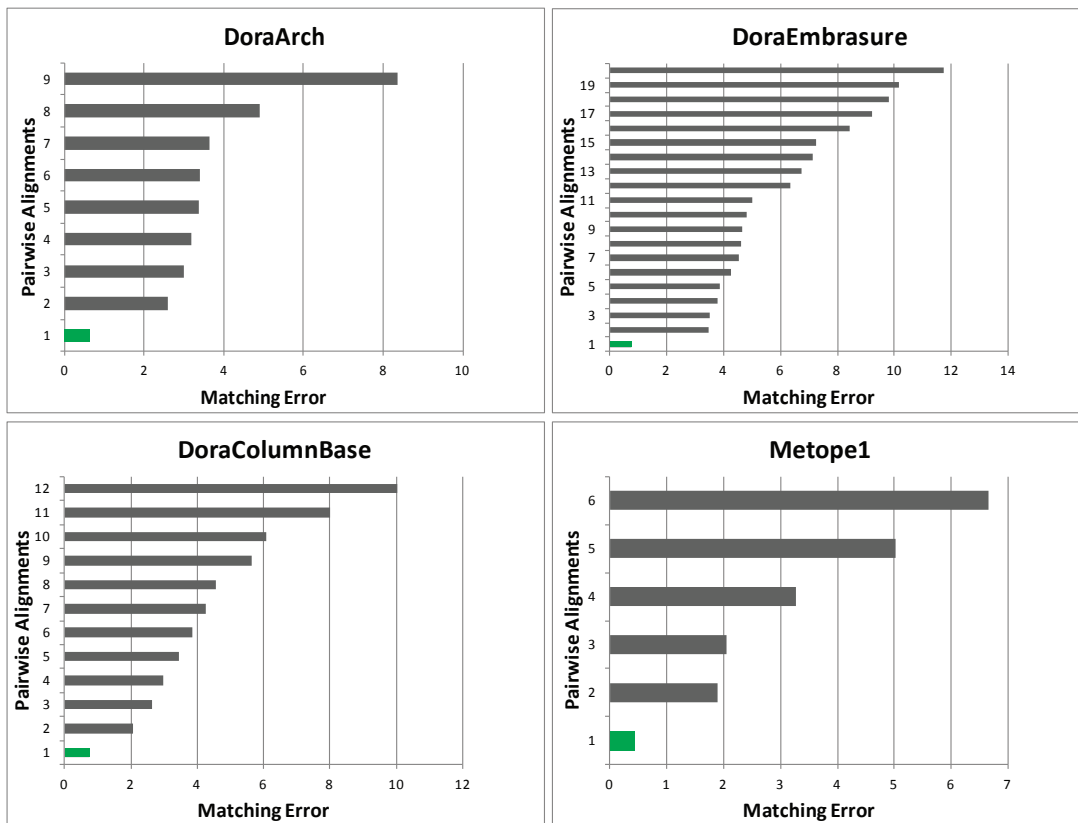


Figure 47: The matching scores for each pair-wise alignment for the experiments shown in Figure 42.

Despite the fact that the use of an ℓ_p -norm with $0 < p < 1$ helps handle slippage and outlier problems, it is not the default setting, as it reduces the discrimination of the matching error. Especially for low values of p , the penalty induced by the outliers becomes very small, and the slope of the objective function is reduced greatly. This is demonstrated in Figure 48, where the results of the matching error for the examples of Figure 43 are presented (fragments with slippage, missing parts and erosion). Notice the diminished difference of the best match (lowest error) with respect to the rest of the results, when the ℓ_p -norm with $0 < p < 1$ is used instead of a standard RMS distance.

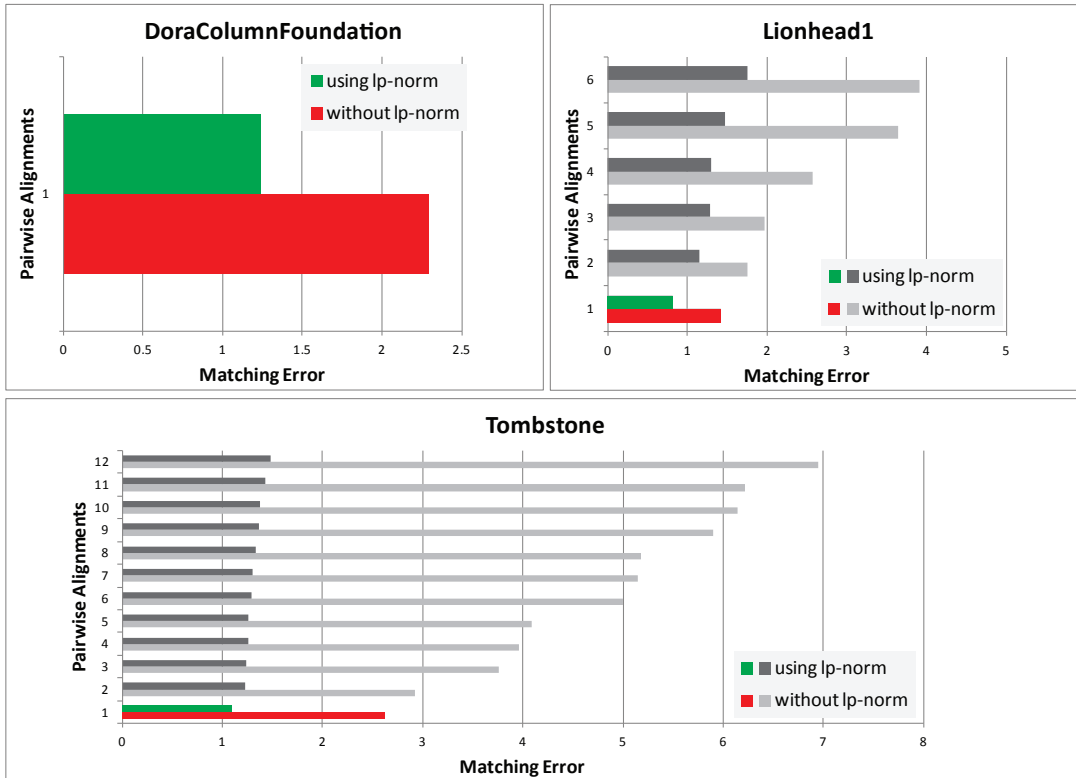


Figure 48: The matching scores for each pair-wise alignment for the experiments shown in Figure 43.

3.1.2 Reassembly of a Single Object

In this section we perform experiments with multiple fragments from a single object. Initially, all facet-to-facet combinations of distinct objects are computed and using Kruskal’s algorithm we find the *Minimum Spanning Forests* of a graph, where fragments are the vertices and pairwise matches the edges.

3.1.2.1 General Effectiveness & Robustness Since in this section fragments of a single object are used we expect a single island of fragments. The robustness of our matching score is evaluated here as its capability to properly group the fragments correctly without leaving isolated islands in the solution.

Quality Assessment. The quality of the single-object reassembly is evaluated similarly to the pair-wise matching; when no ground truth original object is available (i.e. for actual archaeological datasets), visual inspection of the resulting models confirms the correct shape and helps detect severe skewing of the alignment. Quantitatively, again we measure the average matching error and penetration depth. For test sets with ground truth data, we additionally measure the RMS Hausdorff distance against the reference reassembly, with the procedure explained in Section 3.1.1.

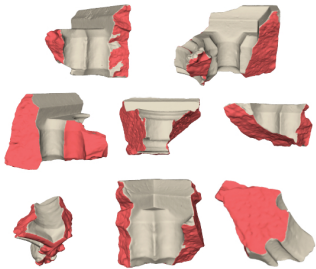

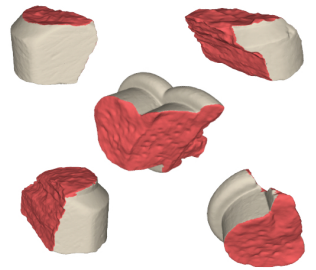
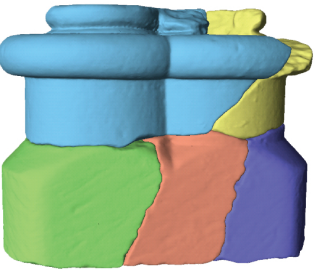
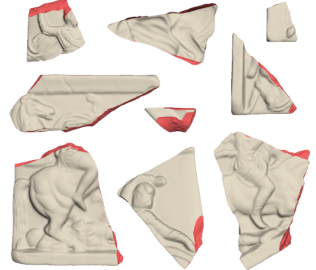

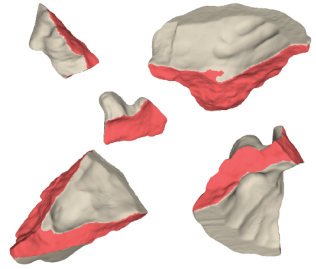
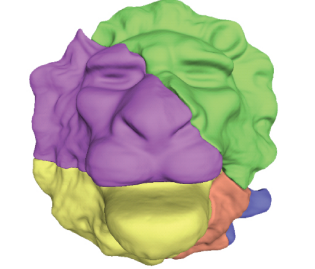
| | Fragments | Reassembled Object | Evaluation |
|----------------|---|---|---|
| DoraEmbrasure |  |  | ESA : 420 sec ICP : 171 sec Penetration : 0.46 mm Mean Error : 1.035 Pair-Tests : 526 Hausdorff RMS : - |
| DoraColumnBase |  |  | ESA : 88 sec ICP : 125 sec Penetration : 0.41 mm Mean Error : 1.17 Pair-Tests : 112 Hausdorff RMS : - |
| Metope1 |  |  | ESA : 100 sec ICP : 150 sec Penetration : 0.2 mm Mean Error : 0.54 Pair-Tests : 233 Hausdorff RMS : 1.14 |
| Lionhead1 |  |  | ESA : 50 sec ICP : 39 sec Penetration : 0.36 mm Mean Error : 0.62 Pair-Tests : 38 Hausdorff RMS : 0.55 |

Figure 49: Object reassembly. All the pairwise combinations are explored and using a graph based approach, the best matches drive the reassembly. We report the total time spent in the ESA optimization and the ICP refinement. The mean pairwise score of all pairs is reported. RMS Hausdorff distance from the ground truth is also given (in mm), where available.

Figure 49 demonstrates some results using single-object collections, two archaeological data sets and two test sets with ground truth data. Measured penetration is below the scanner measurement and meshing software accuracy (0.5mm). The mean error is not indicative of an RMS *distance* between matching fractured surfaces, as it depends on the norm used and also considers measurements across gaps (up to the outlier cut-off limit) for interlocking parts with eroded surfaces.

In the case of multi-part matching, the comparison with the ground truth reference reassembly also measures here (in contrast to pairwise tests) also the error propagation in the chains of connected fragments. Figure 50 shows a comparison of an original object (Metope1 - scanned before breaking) with the reassembled object from the ten resulting fragments after breaking that could be scanned. To demonstrate the quality of the result, we provide the overlaid models after an ICP step. This particular model is ideal to demonstrate the correctness of the multi-part registration, since the fragments have only thin fractures (it is a flat object, max. depth 2.5 cm) and there are at least 3 fragments in a transformation chain, where error can be amplified due to propagation. Nevertheless, the good overlap of the resulting reassembly with the original object proves that the skewing of the result was minimal.

Final Adjustment. A necessary step to obtain optimal results after the clustering of the fragments, is to perform a multi-part ICP to correct any small misalignments that may have occurred due to error propagation across the transformation chains, such as the slight deviations shown in Figure 51 (left). For each fragment, we align the entire set of its fractured facets (treated as a single surface) with set of the fractured facets of all the other fragments (again treated as a single object). This process is repeated for all fragments multiple times, until the reassembly transformations settle (Figure 51 (right)). All multi-part reassembly results shown, utilize this final adjustment step, although in many cases, the resulting reassembly was good enough, even without this step.



Figure 50: Metope1 example. Left: Visual comparison of the scanned object before breaking and the resulting reassembly of the scanned fragments. Right: overlap of the original (blue) and reassembled object (yellow).

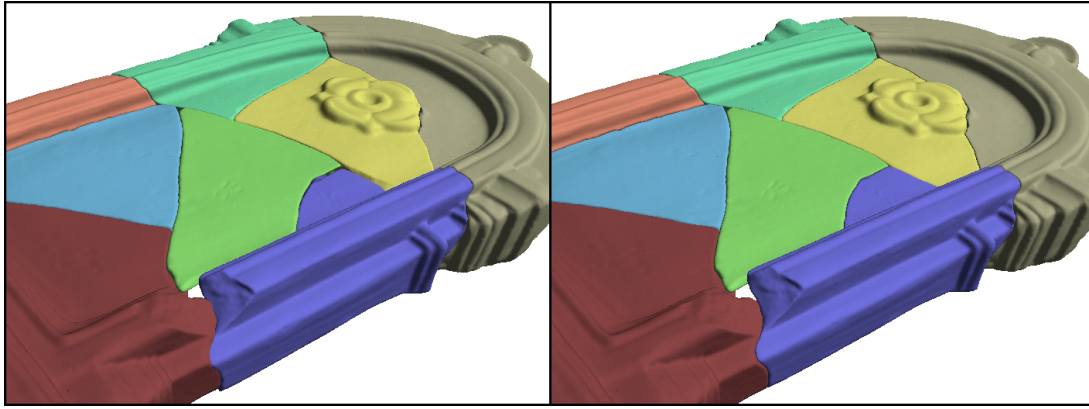


Figure 51: Final fragment adjustment. Left: before, Right: after adjustment.

3.1.2.2 Efficiency & Scalability It is important to study how the combinatorial complexity affects the performance of the reassembly. For example in the case of DoraEmbrasure, despite the efficiency of the pair-wise alignment operations, the reassembly process involves 526 pair-tests and takes 7 minutes. The behavior of the multi-part registration (reassembly) stage in terms of scaling directly depends on the total number of pairwise tests required. For N_f fragments with an average of k fractured facets, the total number of pairwise combinations N_c is:

$$N_c = k^2 \cdot \sum_{i=1}^{N_f-1} i \quad (2)$$

In Table 17 we show some examples of the total time required for the computation of all pair-wise combinations. Clearly, the problem becomes intractable quickly, if pairwise matches need to be re-evaluated on the fly. This simple truth also makes approaches that geometrically combine fragments (union) during the combinatorial multi-part optimization, less attractive for large problems, despite the fact that separate pieces are gradually reduced over time, if not many backtracks occur.

To address the above combinatorial explosion, in the VRMW, and in particular, the *reassembler* server, we utilize a caching scheme, where combinations not forcefully

| | Fragments | Fracture Facets | Pair-wise Combinations (0.5 sec per combination) | Total Time |
|-----------------|-----------|-----------------|---|------------|
| Simple Shape | 4 | 3 | 54 | 0.45 mins |
| Complex Shape | 9 | 5 | 900 | 7.5 mins |
| Nidaros Dataset | 64 | 6 | 72576 | 10 hours |

Table 17: Indicative time required for the computation of all pairwise computations (no data caching). Column *Fracture Facets* shows the average number of facets in the dataset. The "simple" and "complex" shapes are typical statistical representatives of the fragments encountered in all our experiments, while the Nidaros dataset refers to the entire set of fragments from multiple objects scanned at the Nidaros Cathedral.

| | Fragments | Fracture Facets | Next Fragment Pair-wise Combinations (0.5 sec per combination) | Incremental Update |
|-----------------|-----------|-----------------|---|--------------------|
| Simple Shape | 3 | 3 | 27 | 0.23 mins |
| Complex Shape | 8 | 5 | 200 | 1.7 mins |
| Nidaros Dataset | 63 | 6 | 2268 | 19 mins |

Table 18: Time required for the computation of incremental pair-wise computations.

invalidated (by the user) are stored for future re-use. This incremental computation amortizes the cost, which, for the insertion of the i -th fragment f_i in the fragment database, becomes:

$$N_c^{f_i} = k^2 \cdot (i - 1) \quad (3)$$

Table 18 shows some examples of the total time required for the computation of all incremental pairwise registrations, when a new fragment is introduced. Column Fragments indicates the number of fragments already in the collection ($i - 1$), when inserting the last fragment (i).

3.1.2.3 Discrimination & Plausibility. The multi-part registration and clustering of parts is primarily a combinatorial optimization stage, which heavily depends on the quality of results of the pairwise alignment and error measurement calculations. However, isolated results of pairwise matching may produce acceptable pairs that do not lead to a valid overall solution, even though the measured error is low. This can happen if parts of different objects achieve a good score (see mixed puzzles in the next section next), or if parts of the same object exhibit trivial matches due to surface slippage or small area of support. When these particular combinations become part of an active solution in the combinatorial optimization they typically cause penetration of the respective fragments against the overall formed clusters and can be detected and removed. Figure 52 shows a typical example of a reassembly with and without penetration testing.

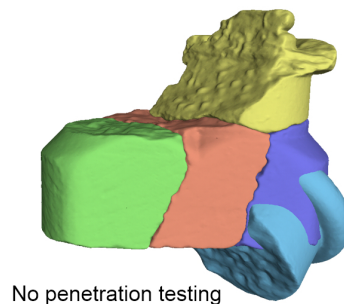


Figure 52: Example of a reassembly without penetration testing. A good match is found between two parts of the DoraColumnBase object, that could lead to the invalid reassembly shown here. When penetration testing is enabled, this configuration cannot occur.

Kruskal's algorithm is modified as follows, in order to incorporate the penetration testing. Whenever a new edge is accepted in the solution, the new fragment node is tested against the sub-tree stemming from the edge's other node. If a penetration is found with the already formed cluster, the edge is penalized by increasing its error (edge weight) and the algorithm is restarted. This effectively solves most penetration problems and at the same time helps the algorithm reject incompatible fragments (or fragments belonging to different clusters, in the case of mixed puzzles). Still, this is not an efficient algorithm and a back-tracking scheme is planned as future work.

For certain puzzles, where fragments had small, smooth fractured surfaces, the pairwise error levels of different part combinations would be comparable. Worse, in these cases, determining a robust pairwise alignment is also hindered, leading to penetrations in the multi-part optimization, even for correct combinations. When such a combination drives a penetration test during the combinatorial stage, the penetration detected increases the error even for a correct match, misleading the algorithm. To resolve such corner cases, manual constraints need to be imposed (user intervention, see below).

3.1.2.4 User Input. In certain hard cases, especially when small fragments with trivial fracture surfaces (e.g. smooth) or a morphology that allows the matching error minimization in many pairwise configurations, a manual filtering of the results is necessary. This comes in the form of a user-guided black- or white-listing of proposed combinations, thus prohibiting or enforcing respectively the joining of two particular facets. This procedure is supported via the reassembler server and will be directly exposed via the graphical user interface in the next version of the VRMW.

Let us mention however that no manual intervention was used in any of the results shown here. All reassemblies were obtained using a fully automatic approach. In the object repair sections that follow, you will notice a reassembly of the Embrasure object with more fragments than the one presented in this section. This extended reassembly was obtained using user constraints to enforce the joining of certain pieces and in particular, certain small protruding decorations, which had a very small and smooth contact surface with the rest of the object.

3.1.3 Reassembly of Mixed Objects

Here we evaluate the performance of our algorithms in the case of mixed puzzles. Fragments from multiple objects are used and in contrast to the former experiment the expected result is multiple clusters of reassembled objects and potentially isolated fragments. In order to achieve that, during the graph construction stage we filter out pairwise matches with score higher than a threshold value E_t , which signify bad alignments. Pairwise alignments could be also directly filtered out due to penetration tests. Here we indirectly cull them off, as the corresponding graph edge error increases due to penetration detection.

The criteria and respective measurements for the mixed puzzle problems are identical to the single-object experiments, with the exception of the discrimination criterion, that affects the plausibility of the results.

3.1.3.1 Plausibility & Discrimination. In Figure 53 we see that in the first two examples our methodology manages to produce the two desired clusters. This is attributed mainly to the good discrimination achieved by the matching error of the fragments. In the third case we see a failure scenario. Using the same threshold E_T here, we see that only 2 out of 4 clusters are found. The 4th cluster (bottom one in figure) consists of two fragments that failed to aligned using the default parameters.

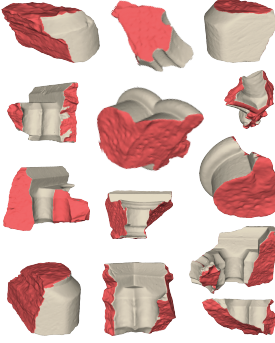
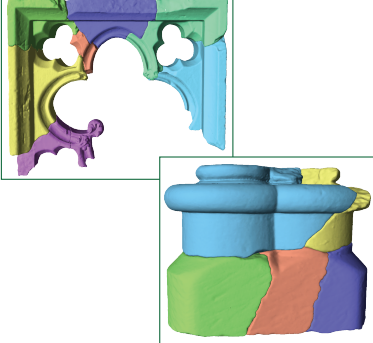
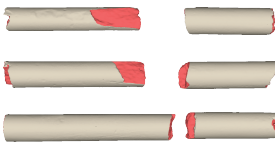
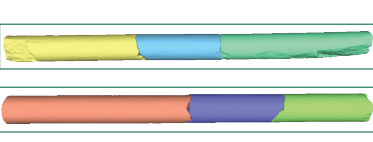
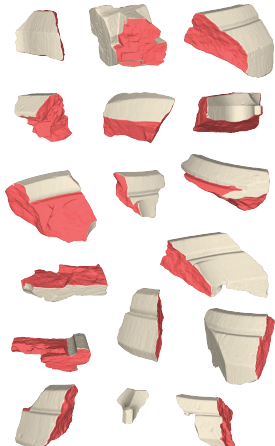
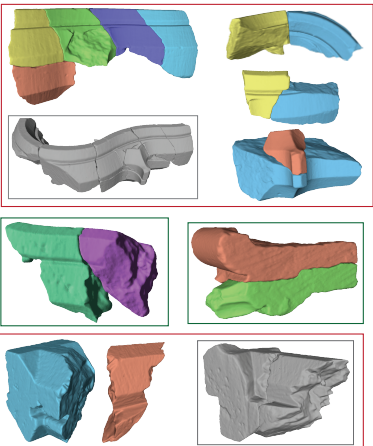
| Fragments | Reassembled Object | Evaluation |
|---|--|---|
|  |  | ESA : 1100 sec ICP : 320 sec Pair-Tests : 1233 E_T : 2.0 Clusters : 2/2 |
|  |  | ESA : 91 sec ICP : 60 sec Pair-Tests : 149 E_T : 2.0 Clusters : 2/2 |
|  |  | ESA : 2100 sec ICP : 525 sec Pair-Tests : 2620 E_T : 1.5 Clusters : 8/4 |

Figure 53: Object reassembly. All the pairwise combinations are explored and using a graph based approach the best matches drive the the reassembly. We report the total time spend in the ESA optimization and the ICP refinement. The total number of pairwise tests and expected/generated clusters. Green boxes denote correctly reassembled clusters, while with red partial or faulty ones. In the second case we visualize the expected cluster (grey-scale object).

In the case of the 1st cluster (top one in figure) our algorithm generated 4 sub-clusters. In such cases user intervention is required as there is no guarantee that an increase of the E_T would produce the correct results. In a realistic scenario, the user would either increase the search space parameters or would enforce/blacklist fragment pairs.

3.1.4 Reassembly of Heavily Damaged Fragments

This is the final case of our evaluation experiments for the reassembly of objects, where we experiment with fragment pairs with large parts of the fractured areas missing. In such cases, contact-based matching approaches fail as the fractured surfaces do not contain enough information for the alignment process. In order to solve these cases we utilize the *feature curves* (*f-curves*) that extend between fragments and are explained in D1.2, along with the revised objective function for this purpose.

3.1.4.1 General Effectiveness and Robustness. In Figure 54, we present pairwise alignment results that were obtained using the f-curves. Without them, the correct alignment cannot be found as there will always exist another transformation that maximizes the contact surface. Here in contrast to the previous test, we use the Sparse ICP (SICP), which is applied on both the feature curves and the contact surface. The use of any other ICP variant skews the results due to the amount of missing data. The first pairwise match is obtained by using a weight of 0.9 for the feature curves and 0.1 for the contact surface, while in the second pair, we perform the ESA optimization only on the feature curves. 5000 iterations and 4 restarts in the ESA optimizer were used, and an ℓ_p -norm with $p = 0.2$. The fourth alignment showcases another use of the method that essentially enhances the robustness, as the matching facets do share a contact area that is almost planar, and by using only the surface contact criterion the alignment is skewed.

3.1.4.2 User Input. We have to note here that that the score of the alignment when feature curves are used, is not compatible with the score when only the matching error on the contact surface is measured. As we can see from the results, the residual error reported might be misleading and without visual inspection one cannot argue on the validity of the results. This leads to the conclusion that for cases as the ones showcased, user inspection of the results is required.

3.1.4.3 Discrimination. Despite that fact, from Figure 55 we see that the use of f-curves greatly enhances the discriminating ability of the matching error.

3.1.4.4 Plausibility. When unreliable contact surfaces mandate the use of the f-curves to align two pieces with a significant bias towards the f-curve-based alignment

(via the corresponding weighting), there is no way to automatically infer the distance along the feature trajectories at which the two pieces should be positioned. Therefore, instead of a single solution, for all practical applications, a solution space is provided instead, where the two fragments are constrained on the trajectories. Our algorithm implementation supports this, being able to obtain a particular instance of the solution space at a given offset. Still, in the results presented here, we choose a marginally non-penetrating distance, since we know that we have no cases with completely disjoint pieces, especially in the multi-part problem as a whole.

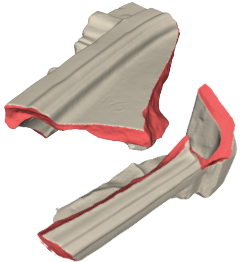
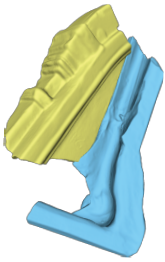
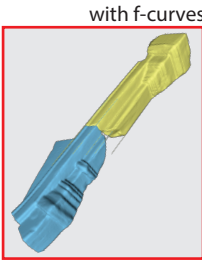
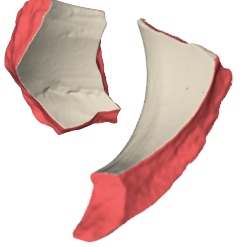
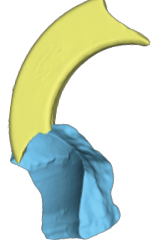
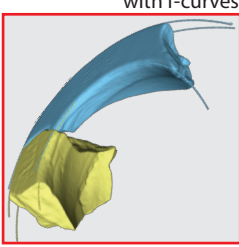
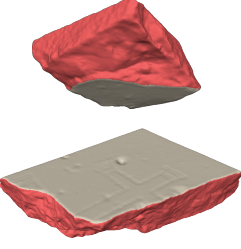
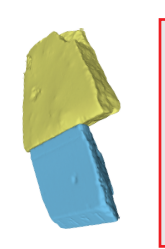
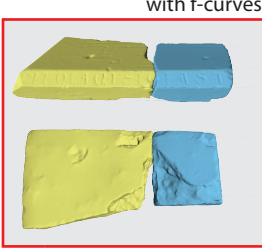
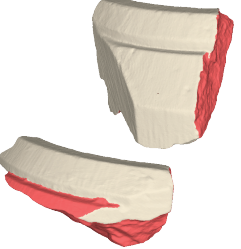
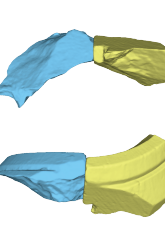
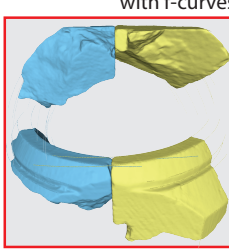
| | Fragments | Pairwise Alignment | Evaluation |
|-----------|---|---|---|
| Fountain |  |   with f-curves | ESA : 26.7 sec (24.5) ICP : 4.7 sec Penetration : 0.1mm Matching Error : 1.9 (1.1) Pair-Tests : 9 Feat. Curves : 0.95 Hausdorff RMS : 0.223 |
| OrnatePot |  |   with f-curves | ESA : 5.75 sec (9.7) ICP : 9.12 sec Penetration : 0.18 mm Matching Error : 1.68 (1.1) Pair-Tests : 9 Feat. Curves : 1.0 Hausdorff RMS : - |
| TombStone |  |   with f-curves | ESA : 22.3 sec (20.7) ICP : 0.15 sec Penetration : 0.26 mm Matching Error : 1.49 (1.2) Pair-Tests : 16 Feat. Curves : 1.0 Hausdorff RMS : - |
| DoraArch |  |   with f-curves | ESA : 27.8 sec (24.1) ICP : 1.12 sec Penetration : 0.18 mm Matching Error : 1.8 (1.1) Pair-Tests : 30 Feat. Curves : 0.5 Hausdorff RMS : - |

Figure 54: Pairwise alignments for fragments with large missing parts. The annotated result (red square) is the one where the feature curves are used. We report the ESA optimization time and in parentheses the equivalent time without the use of external features. The time spent in the ICP refinement of the best match, the matching score (in parenthesis the score without the feature curves), the value of p , the total number of pair-wise tests, the weighting of the feature curves and the Hausdorff distance (RMS) from the ground truth, where available.

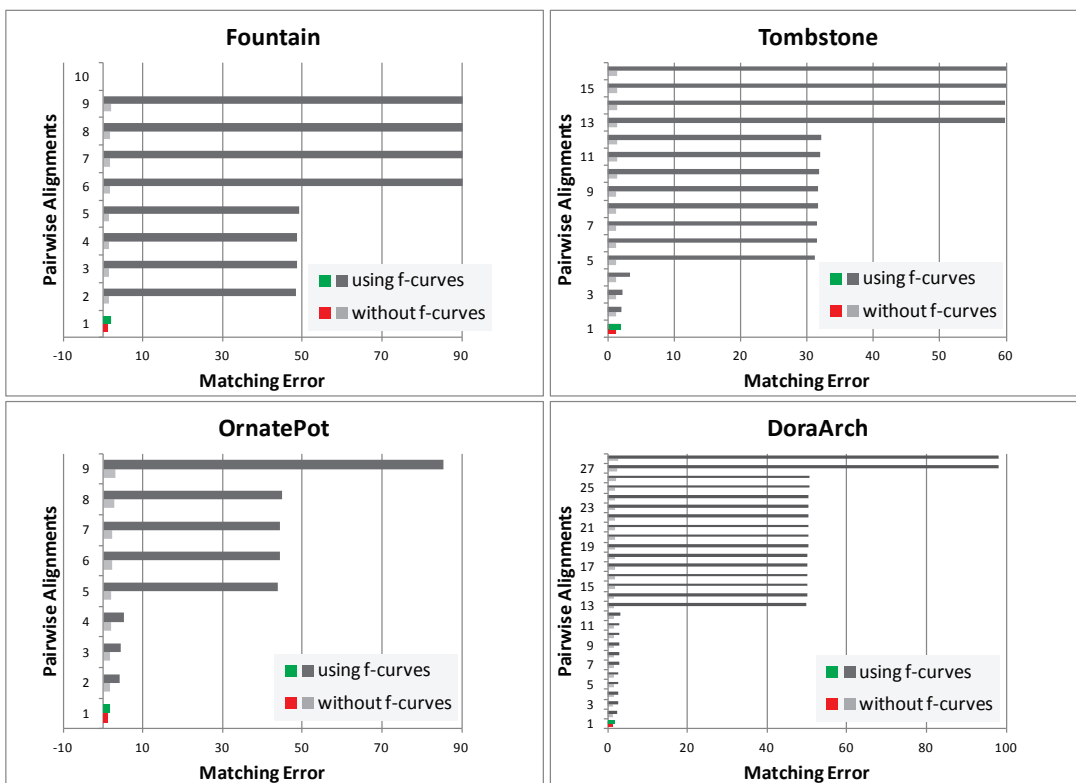


Figure 55: The matching scores for each pair-wise alignment for the experiments shown in Figure 54.

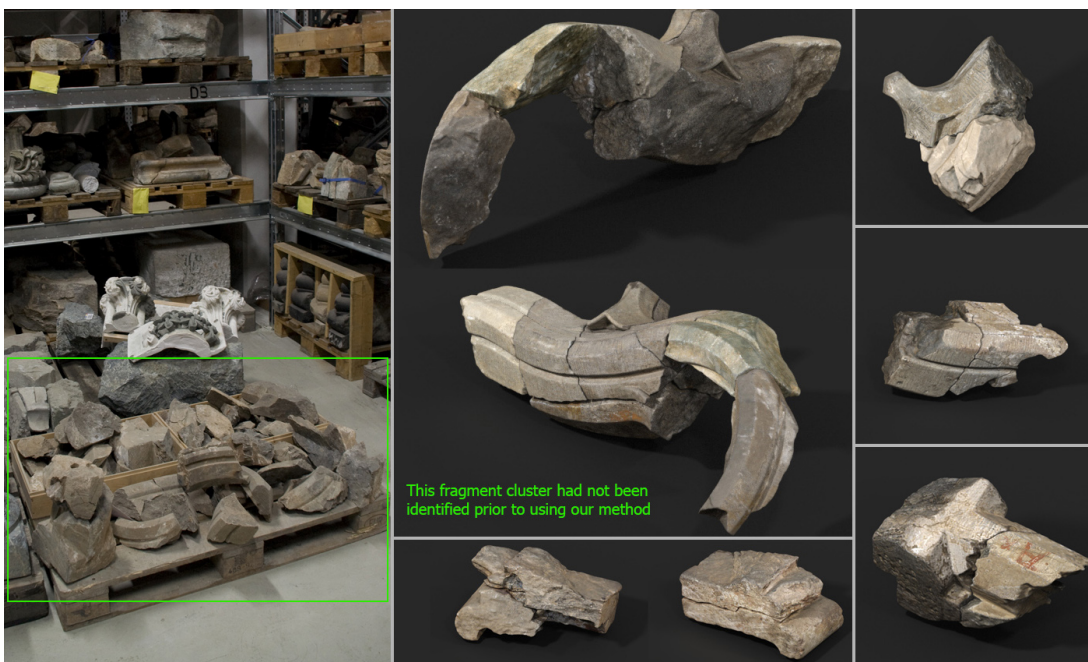


Figure 56: The new part configurations discovered by our algorithm.

A noteworthy contribution of the use of the *f*-curves approach in the context of the archaeological data we have, is that we managed to discover some new combinations of fragments in the DoraArch set of fragments from the Nidaros Cathedral. The reassembler proposed new combinations that were nearly impossible to detect by visual inspection of the physical fragments or their virtual counterparts. The joined parts in the proposed reassembly, that uses *f*-curves, contact surfaces and white-/black-listing is presented in Figure 56.

Finally in Figure 57 we present complete object reassemblies that were obtained using pairwise alignments of both contact surface and *f*-curves. Due to the fact that these results were obtained in a semi-automatic way, as user evaluation of alignments was required, we do not present timings.

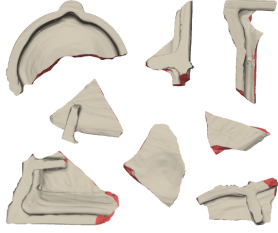
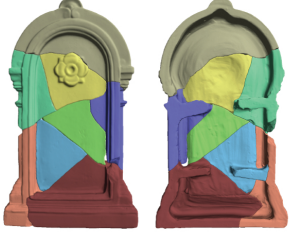
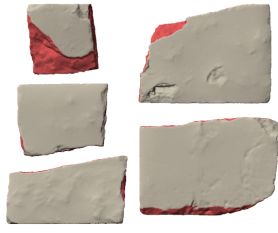
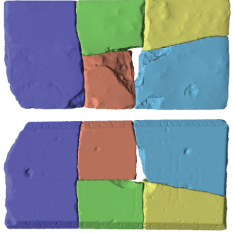
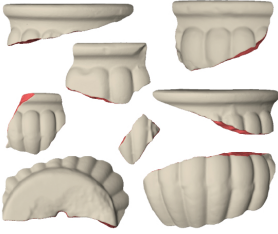
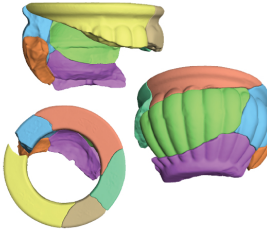
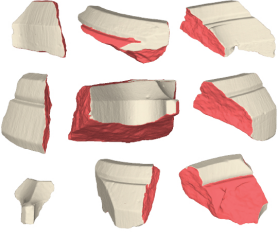
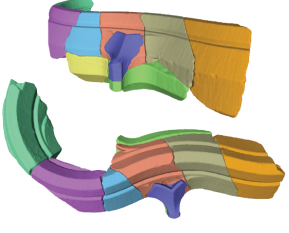
| | Fragments | Reassembled Object | Evaluation |
|-----------|---|---|---|
| Fountain |  |  | Penetration : 0.01 mm Pair-Tests : 273 Hausdorff RMS : 1.16 |
| Tombstone |  |  | Penetration : 0.81 mm Pair-Tests : 157 Hausdorff RMS : - |
| OrnatePot |  |  | Penetration : 0.46 mm Pair-Tests : 422 Hausdorff RMS : - |
| DoraArch |  |  | Penetration : 0.68 mm Pair-Tests : 772 Hausdorff RMS : - |

Figure 57: Object reassemblies that were obtained using both the contact surfaces and the *f*-curves. We report the number of pair-wise tests, the penetration of the reassembly and the Hausdorff distance RMS from the ground truth where possible.

3.2 Evaluation of Object Repair

The goal in T4.2 is the completion of damaged objects and the prediction of missing parts. This section describes the experiments and results obtained by our Symmetry-based Completion algorithm which was already described in D4.2 (First Object Re-assembly and Repair Engine). The focus in this evaluation report is on three main aspects: effectiveness, efficiency and robustness. For each aspect, we conducted a set of experiments that show the suitability of our algorithm for the required goals. In addition, we briefly describe the data used in our experiments as well as the evaluation criteria.

For the sake of completeness, here we include the setup we used in all our experiments. For a detailed description of each parameter, we refer to D4.2 Section 3. The setup follows:

- **Poisson reconstruction:** depth of octree $\delta = 8$, depth of Gauss-Seidel solver $\rho = 8$.
- **Eigendecomposition:** num. eigenvalues = 300.
- **Symmetry detection:** time $t = 0.1$, similarity threshold $\beta = 10$, threshold for distance coherence $\theta = 0.1$, threshold for orientation coherence $\alpha = 0.14$, threshold for feature coherence $\rho = 0.1$, curvature threshold $\delta = 0.01$.

3.2.1 Dataset

We conduct several experiments over a number of input shapes. The data can be divided in two groups: real data from the PRESIOUS project and simulated data of fragment sets. In this section, we briefly describe the data we used in our experiments.

3.2.1.1 PRESIOUS data

We selected four shapes from the project collection (see Fig. 58). The shapes and their properties are listed in Table 19. More details about acquisition and properties can be found in deliverable D5.5 Pilot Acquisition Data.

| Shape | # Frag. | # Vert. | # Tri. |
|----------------|---------|---------|-----------|
| DoraEmbrasure | 10 | 524,537 | 1,048,999 |
| DoraColumnBase | 5 | 260,018 | 519,999 |
| Tombstone | 5 | 605,010 | 1,210,000 |
| Fountain | 8 | 908,814 | 1,817,608 |

Table 19: Properties of PRESIOUS data.

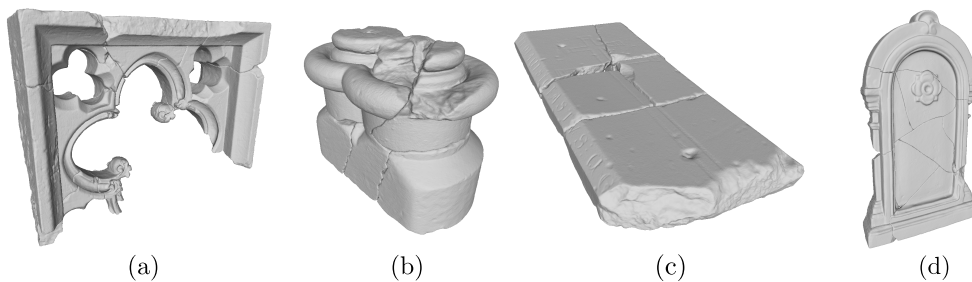


Figure 58: PRESIOUS Data. (a) Dora Embrasure, (b) Dora Column Base, (c) Tombstone, and (d) Fountain.

3.2.1.2 Benchmark of Fragmented Objects

As the number of digitized fragment sets is limited, we incorporate additional fragment sets to provide encompassing results. It allows us to use a considerable number of test cases, making the evaluation more robust and meaningful.

Relying on OpenVDB's [Mus13] Level Set Fracture algorithm, we developed a tool that can generate random, pre-aligned fragment sets from arbitrary input shapes. Although the Level Set Fracture Algorithm is not able to simulate physically correct fragmentation results (as it was originally intended for computer animation purposes), the features of the breaking edges can be influenced by the shape that is used as the *cutter object*.

This object is repeatedly rotated and translated along random paths that intersect the input object. New breaking edges are synthesized from the intersection of the cutter surface and the remaining volume of the input shape. In contrast, an opposite breaking edge is derived by computing the difference of the input volume and the original breaking edge. In a post-processing step, complementary fragments are degraded to avoid intersections and provide more plausible fissures on the exterior of the resulting aligned fragment set. The most plausible results were achieved when using a model of a stone fragment as the cutter object. An example of fractured objects can be observed in Fig. 59.

Although we could fragment any 3D object and use it for our evaluation, the idea is to limit the scope of the benchmark to cultural heritage data. The base dataset for our benchmark is the Hampson collection¹, which provides a varied set of archaeological objects. We took a sub-set of 76 objects (listed in Appendix B) and applied the aforementioned fragmentation tool to produce the data used in our evaluation. An example of 3D object and its resulting fragments is shown in Fig. 60.

3.2.2 Experiments

The evaluation is divided in three aspects: general effectiveness, efficiency and robustness. In the following sections we describe the performed experiments and discuss

¹<http://hampson.cast.uark.edu/browse.htm>

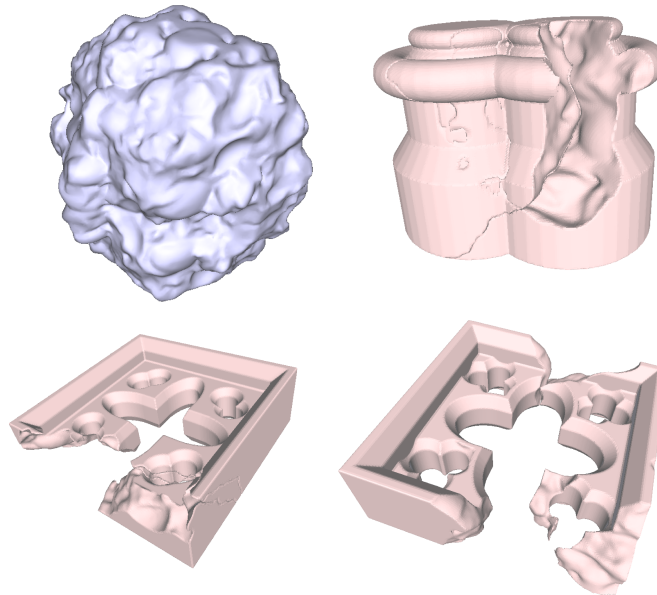


Figure 59: Artificial Object Fragmentation. Top left: The enlarged stone model used as cutter object for OpenVDB's Level Set Fracture Algorithm. Top right and bottom: Results of the Level Set Fracture on different generated templates after omitting one of the resulting fragments

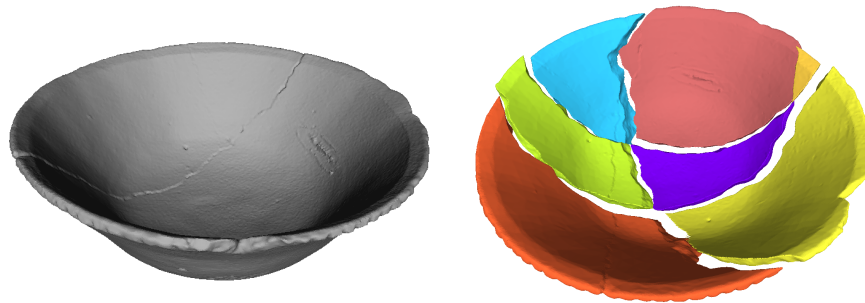


Figure 60: Hampson object and its corresponding fractured set.

the obtained results.

3.2.2.1 General Effectiveness

Our first goal is to test the effectiveness of our algorithm to complete damaged objects. In this context, we use the PRESIOUS data since they present a high variability of shapes, characteristics and complexity. Nevertheless, the lack of a ground-truth for real data leads us to make an evaluation based on the visual inspection of the results. Also, although the success in the resulting completed object can be subject to interpretation by archaeologists, some geometric characteristics can be still evaluated. We plan to include the user judgment for evaluation in D5.8. Among the criteria that are useful for evaluation, we can consider the visual feasibility of the completion, the continuity of geometric features and the level of detail of the completion, just to name a few. We evaluate each PRESIOUS object separately in order to show the

relation between shape characteristic and the aforementioned criteria.

Dora Embrasure

The embrasure is a difficult example for completion due to the large amount of geometric features and the severe damage. The object contains curves and feature lines that need to be respected in the final completion.

Our algorithm exploits the information of symmetry to produce a visibly good approximation of completion. More in detail, the completion algorithm is able to produce smooth transitions between the input object and the new geometry. For example in Fig. 61(a), despite the damaged reference line in the upper part of the object, our method approximately predicts a suitable completion. Likewise, the continuity of feature lines and curves can be evidenced in Fig. 61(b) and 61(c). The completion algorithm guarantees this behavior through two of its steps. First, the use of symmetry guarantees that a large amount of features match almost perfectly, including the curves and feature lines. Second, the final registration refines the corresponding geometry, making both shapes (input and result) to fit as much as possible.

On the other hand, an advantage of our method is the ability of produce a resulting completion at the same level of detail as the input shape. Therefore, techniques for inpainting or geometry transfer could not be necessary.

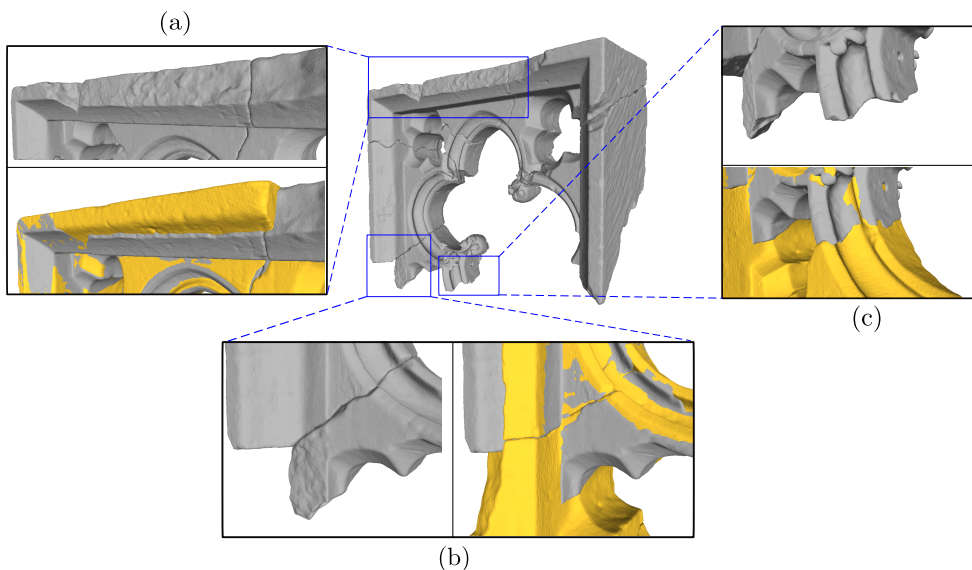


Figure 61: Detailed result for Dora Embrasure.

Tombstone

Unlike embrasure, the tombstone object contains less features and more uniform regions. This example is particularly interesting because it demonstrates that our algorithm (which was meant to be feature-driven) is also useful for feature-less objects.

A challenge in tombstone is that a large missing part is located in a planar section of the object. The result of our algorithm can be seen in Fig. 62. In Fig. 62(a), the missing part was partially completed to the extent by which geometry was available along the symmetry plane. The main problem with this large missing part is its extension, which covers both sides of the symmetry plane used in the reconstruction. Obviously, the algorithm copies the geometry through the symmetry plane, and therefore there is no way to complete the gap present in the middle. Still, overall the method recovers a large portion of missing geometry which could be useful in a subsequent processing step.

On the other hand, the completed object respects the feature lines and preserves the continuity of geometry, as can be shown for instance in Fig. 62(b).

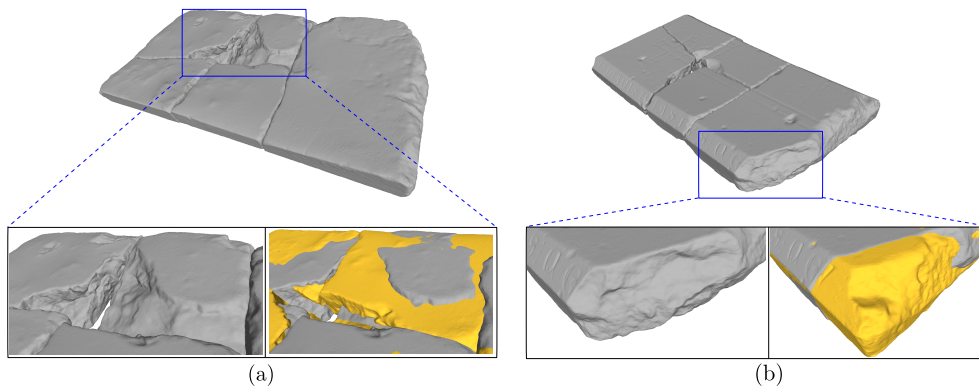


Figure 62: Detailed result for Tombstone.

Column Base

This object contains big curved surfaces and very damaged zones. Interestingly, our algorithm is able to complete the deep fracture in the top of the object, as shown in Fig. 63(b). In addition, the geometry of the completed part fits well with the geometry of the surrounding, allowing a smooth transition. Likewise, the completion algorithm was able to reconstruct the large curved missing part in front of the object while respecting the feature curve, as depicted in Fig. 63(a).

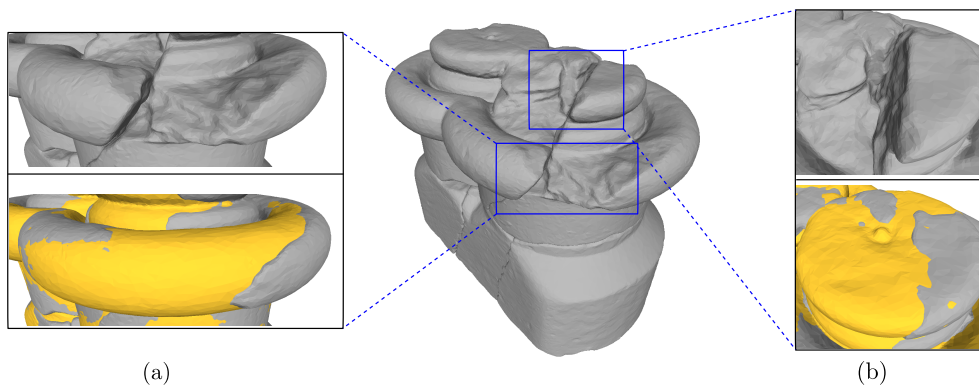


Figure 63: Detailed result for Column Base.

Fountain

This object contains a mixture of feature lines and planar sections. For this evaluation, we intentionally removed a fragment and applied the completion algorithm (see Fig. 64).

The completion algorithm was able to complete almost perfectly the missing fragment by copying the corresponding symmetric geometry. It is also worth to mention that our algorithm performed very good in providing smooth transitions in the feature lines, as shown in Fig. 64(a). Also the planar section was well predicted and fitted. Likewise, the small missing part at the bottom of the object was successfully filled by our method. Note the continuity in the feature lines and the almost exact repair of the missing part and the contact surface between two fragments.

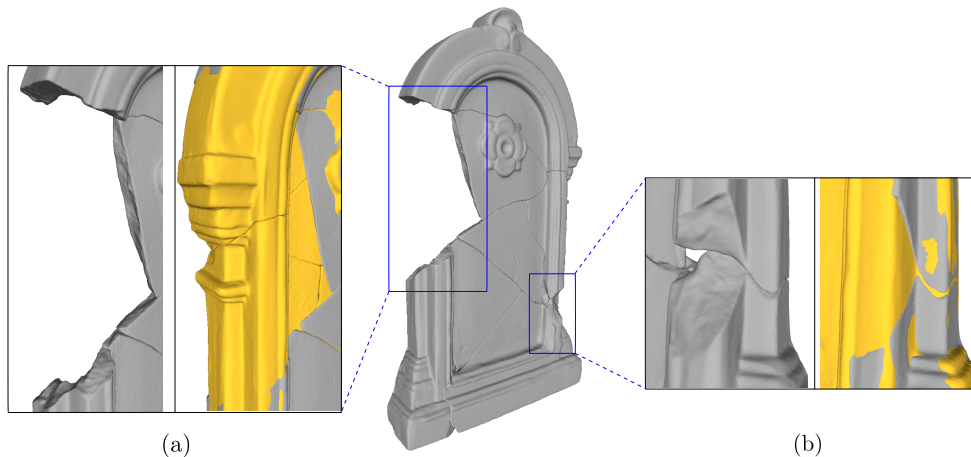


Figure 64: Detailed result for Fountain.

From this first subjective evaluation of the results on real data, we can conclude that our Symmetry-based Completion algorithm is able to deal with varied types of geometries. In particular, this makes our method suitable for the cultural heritage domain. Also, it is important to remark that our method is also a good alternative in different levels of damage. As long as there is symmetric information to take advantage, our method can guarantee a well-approximated solution.

3.2.2.2 Efficiency and Scalability

To measure the efficiency, we executed our completion tool on each PRESIOUS object in 50 different runs. The time presented in this section shows the average time of the 50 runs. It is worth noting that we take special care on the time measurement procedure. For this purpose, we measured the wall-clock time which is the elapsed time to complete a run from the point of view of the human perception. This measure is useful because it is the time the user will perceive when the algorithm runs. In the implementation side, we took advantage of the *chrono* library, which is now incorporated in the C++11 standard library. All the experiments were performed on a Intel Core i7-4770 CPU @ 3.40GHz with 32GB of RAM, running Linux 64bit.

In this point it is important to recall the main steps of our algorithm in order to present more detailed information about timing. Our completion algorithm can be summarized in the following steps:

- **Pre-processing (Pre-pro):** includes the Poisson reconstruction [KH13] of the set of input fragments and the removal of small components.
- **Eigendecomposition of Laplace-Beltrami operator (Eigen):** includes the computation of the Laplace-Beltrami matrix of the pre-processed mesh and its eigendecomposition.
- **Completion:** includes the detection of symmetry planes, the registration for refinement and the computation of the completed object and missing parts.

| Shape | Pre-pro. (sec.) | # Vert. Pre-pro | Eigen. (sec.) | Completion (sec.) |
|----------------|-----------------|-----------------|---------------|-------------------|
| DoraEmbrasure | 0.982068 | 8,042 | 19.5218 | 34.5934 |
| DoraColumnBase | 0.788206 | 17,152 | 78.5114 | 165.981 |
| Tombstone | 1.102610 | 4,736 | 11.6716 | 217.886 |
| Fountain | 1.390400 | 6,073 | 17.8395 | 158.738 |

Table 20: Time processing for each step of our completion algorithm. All times are given in seconds. We also include the number of vertices after the pre-processing since it is the resolution in the subsequent analysis.

Table 20 shows the average times for each processing step of our algorithm. A first finding is that the pre-processing step is fast, and at the same time it allows us to decrease the resolution of the input geometry. The pre-processing facilitates the computation of the subsequent step, where the computation of a Laplace-Beltrami operator for a large-scale mesh would be unmanageable (note the considerable reduction of the number of vertices after the pre-processing (cf. Table 19)). We noted that the entire resolution was not necessary for the analysis of symmetries. Nevertheless, the execution time is dominated by the completion step, mainly due to the registration. The final refinement is performed on the original input object, and hence it is time-consuming due to the large resolution of the input objects (in this step, the resolution of the objects is the same as shown in Table 19).

The final refinement needs to be done in the original data because the maximum precision in fitting is required. It means we need to sacrifice time for the best effectiveness we can obtain. This should not be a problem given that in the cultural heritage domain, the precision is more important than providing real-time solutions.

3.2.2.3 Robustness

This section is devoted to test several aspects related to the robustness of our method.

Completion and Missing Parts

In this experiment we evaluate the ability of our algorithm to produce completed objects and the missing parts from a set of input fragments. For this purpose we used the benchmark of fragmented objects described in Section 3.2.1.2.

Each object in the benchmark is associated to a set of fragments which were obtained by our fragmentation tool. Subsequently, we randomly discarded one fragment and executed our algorithm on the remaining fragments to obtain a completed object and the missing part. In the evaluation, we can use the original object as the ground-truth for the completed result and the discarded fragment as ground-truth for the computed missing part.

More formally, let O_i be the i -th object in the benchmark. Let $F_i = \{f_i^j\}$ be a set of fragments associated to O_i . Let $f_i^* \in F_i$ be the discarded fragment. After applying our algorithm to the set of fragments $F_i \setminus \{f_i^*\}$, we obtain a completed object C_i and a missing part M_i . Our criterion to evaluate the robustness of our algorithm with respect to the completion is the ratio of volumes

$$E_{\text{completion}}(O_i) = \frac{\text{vol}(O_i \cap C_i)}{\text{vol}(O_i \cup C_i)} \quad (4)$$

It is worth to note that $E_{\text{completion}}$ is a conservative measure for the robustness of our algorithm. This is because we penalize the congruent geometry ($\text{vol}(O_i \cap C_i)$) with the divergent geometry ($\text{vol}(O_i \cup C_i)$). A clear example where this penalization is important is when the original object is completely covered by the completed object, but in addition the completed object introduces more geometry than needed. In that case, the ratio will decrease depending on the amount of divergent geometry. Also note that a perfect completion will give a $E_{\text{completion}} = 1$.

We also compute the degree of missing geometry for each object which will be useful as a reference for comparison

$$D(O_i) = \frac{\text{vol}(F_i \setminus \{f_i^*\} \cap O_i)}{\text{vol}(F_i \setminus \{f_i^*\} \cup O_i)} = \frac{\text{vol}(F_i \setminus \{f_i^*\})}{\text{vol}(O_i)} \quad (5)$$

Note that $D(O_i)$ is the volume ratio of the input fragments compared to the original object. This quantity gives us an idea about how much information was removed by discarding one fragment.

Regarding the evaluation of missing parts, our criterion is also defined by the ratio of volumes

$$E_{missing}(O_i) = \frac{\text{vol}(f_i^* \cap M_i)}{\text{vol}(f_i^*)} \quad (6)$$

where the measure computes the fraction of missing geometry that has been predicted by the algorithm. In this case, we are not interested in a conservative measure given that we only want to know if the real missing geometry is part of the solution set delivered by our algorithm.

To compute the volume of a 3D shape, we implemented the algorithm proposed in [ZC01]. Briefly, the algorithm accumulates the signed volume of the tetrahedron formed by each triangle in the shape and the origin.

Completion results In our experiments, we executed our algorithm in 76 sets of fragments (one per object in the benchmark of fractured objects). On average, our method obtained $E_{completion} = 0.8267$. Nevertheless, this average alone is not a good indicator for the effectiveness of our method. A more meaningful comparison is the gain of effectiveness; that is the difference $E_{completion}(O_i) - D(O_i)$ for each object. This difference can reveal whether our method really complete some geometry or not.

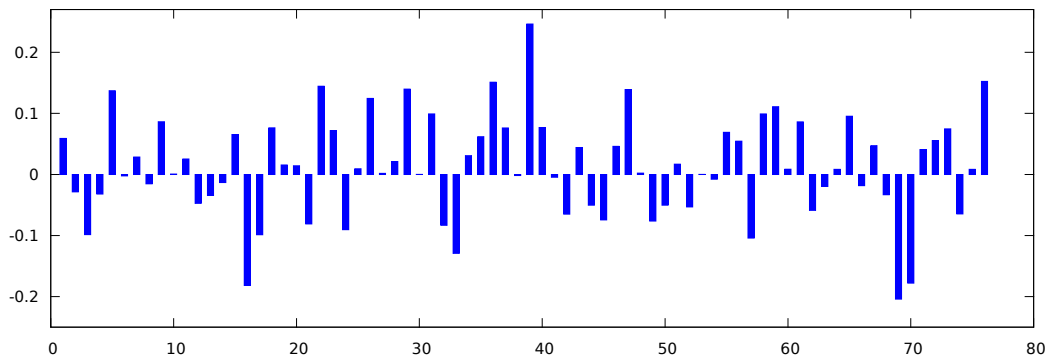


Figure 65: Difference between effectiveness and partiality of the fragmented objects. Each bar symbolizes the magnitude of how much geometry was completed. Positive bars illustrate successful completions, whereas negative bars represent bad completions.

Fig. 65 shows the gain of effectiveness for each query object. Our method obtained a positive gain in 45 out of 76 objects. This fact reveals that on average our method is able to infer and complete missing geometry. After inspecting some random objects with negative gain, we discovered a reason for the negative difference: when the missing fragment is present across the symmetry planes, our method is not able to reproduce some parts of the completion. Recall that this behavior was already evidenced in the evaluation of the Tombstone object.

Additionally, given the availability of the ground-truth of the Fountain object, we tested the completion algorithm assuming one piece at a time is missing. Figure 66

shows the results obtained when one piece at a time of the fragment set was removed. We also add the $E_{completion}$ measure for each result for the sake of completeness. Note that in all cases our algorithm was able to predict missing information. Of course, the amount of predicted geometry depends on how much symmetric information is available a priori (for instance in Fig. 66a, b, c, g and h, the completion algorithm cannot predict the central part, just because that information is not available in the symmetric counterpart). Nevertheless, the results shows that it is still possible to get part of the missing object, which can be useful for subsequent completion tasks.

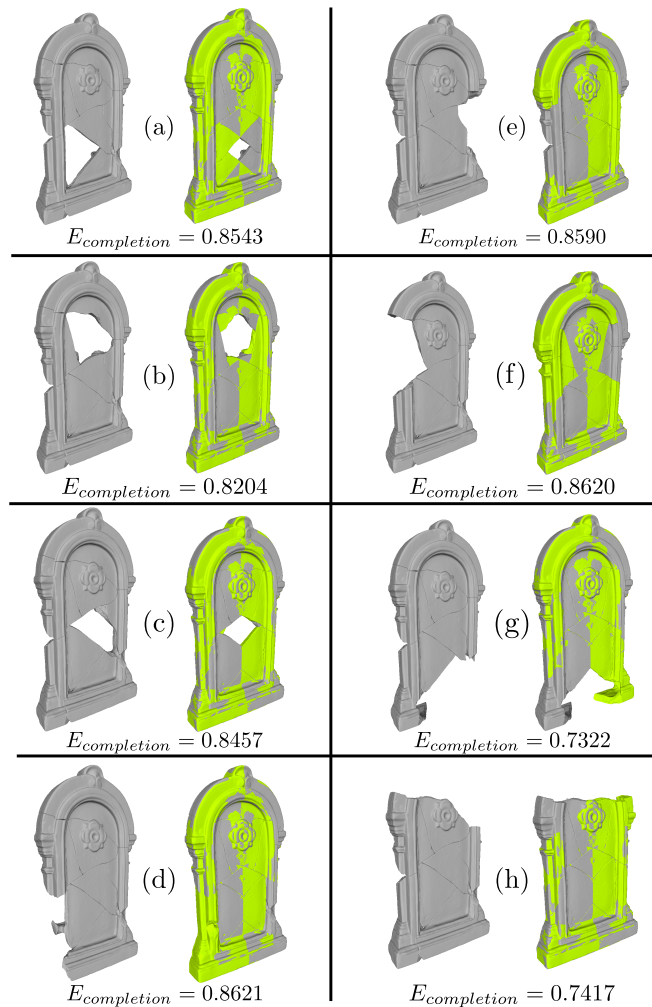


Figure 66: Several experiments with Fountain object. We removed one piece at a time and applied the completion algorithm. The results show the overlapping between the original shape and the symmetric counterpart. In addition, each result is accompanied by the $E_{completion}$ measure.

Missing part results Regarding the computation of missing parts, we computed the $E_{missing}$ measure for each object in the benchmark. The average $E_{missing}$ is 0.5928. This result is encouraging because it proves that, on average, our algorithm is able to predict missing geometry in the input data. In fact, the amount of missing data predicted is above 50%, which reveals that our algorithm is guaranteed to create

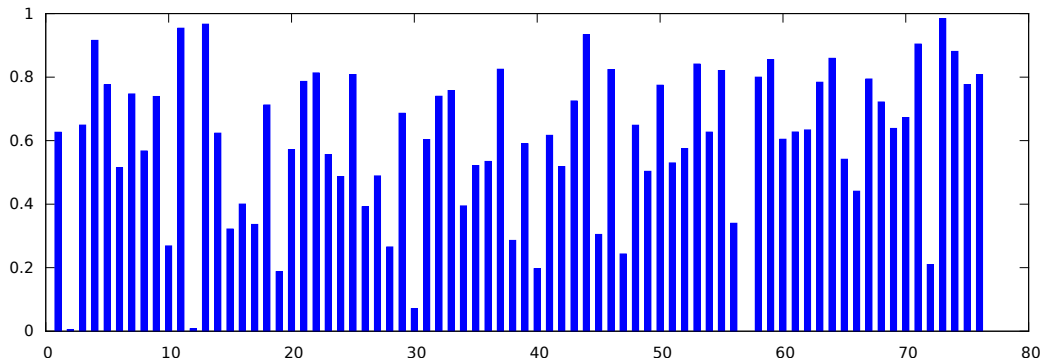


Figure 67: Percentage of missing geometry predicted by our algorithm in each query set.

more than one half of the missing part. This result opens a new interesting question whether a consecutive application of the completion algorithm could predict even more geometry or not. We will leave the evaluation of consecutive runs for D5.8.

Likewise the evaluation in the completion part, the average itself could be hiding relevant information about the results. For this reason, we present the $E_{missing}$ for each object in the benchmark in a bar plot shown in Fig. 67. Note that our algorithm is able to produce good results in most of the query inputs. This result shows that our algorithm produces missing geometry with good chances of containing the real missing part.

Scale and Resolution Invariance

In this experiment, we evaluate the robustness to scale and resolution invariance. With respect to the scale, our algorithm is fully invariant to changes in object scale. This is guaranteed because the symmetry analysis is performed on a mesh with surface area of 1. This means that whatever is the scale of the input object, the output is the same.

With respect to the resolution, we experimented with the down-sampling of the input fragments. That is, for each set of fragments used in Sec. 3.2.2.3, we applied a simplification algorithm [GH97] to reduce the number of points in approximately 50% and 25%. Subsequently, we executed our algorithm over the set of down-sampled fragments, while preserving the ground-truth in the original resolution. The obtained results in this experiment are shown in Table 21.

| % Down-sampling | Avg. $E_{completion}$ |
|-----------------|-----------------------|
| 100% | 0.8267 |
| 50% | 0.8202 |
| 25% | 0.8059 |

Table 21: Effect of down-sampling the input fragments in the average effectiveness measure for completion.

The change in the resolution in the input fragments does not affect considerably the effectiveness of our method. This result confirms that the pre-processing step (initial Poisson reconstruction) plays an important role in controlling the resolution of the analyzed mesh. On the other hand, the decreasing value of effectiveness in 25% of down-sampling can be caused by the small modification of the geometry in the simplification algorithm. Also, the method used to measure the volume of a 3D shape heavily relies on the resolution of the evaluated objects, somehow affecting the computation of our evaluation measures. The same conclusion can be drawn from the evaluation of the prediction of missing parts when the resolution changes. Table 22 shows the average $E_{missing}$ in different resolutions.

| % Down-sampling | Avg. $E_{missing}$ |
|------------------------|--------------------------------------|
| 100% | 0.5928 |
| 50% | 0.5642 |
| 25% | 0.5511 |

Table 22: Effect of down-sampling the input fragments in the average effectiveness measure for missing parts prediction.

Introduction of Mesh Defects

In this experiment, we evaluate if our algorithm produces clean meshes as output. For this experiment, we implemented a tool that evaluates the geometry of the output. More specifically, our tool evaluates the manifoldness of an object and the presence of small holes. As a result, no defect was observed among all the experiments we performed.

3.3 Deviations and corrections with respect to WP4 evaluation plan

Generally, there were no major deviations or corrective actions that were necessary in WP4 during year 2.

For task 4.2 as mentioned above, within the evaluation plan there is a shift to focus first on symmetry-based repair (for D5.4), and on template repair later (for D5.8). This re-prioritization as compared to the plan (D5.1) was done to take advantage of first results obtained by symmetry-based repair, which showed to work well right from the initial implementation. In the mean time, preparation of the template-based repair continues by further populating the template repository and customizing the HKS descriptor extraction and coding, in order to work best for partial retrieval using fragments as queries. Hence, template-based repair will be evaluated in-depth in year 3.

3.4 WP4 items remaining to be evaluated in D5.8

Further advances and improvements in the reassembly process beyond the current state will be evaluated in D5.8. More specifically, in Y3 we will investigate practical approaches to exploit overall-shape-based reassembly (external fragment shape has been cover so far, see f-curves in D1.2). This includes (but is potentially not limited to) symmetry-based reassembly, which has been successfully applied in the literature to trivially symmetric objects (vessels), extended to more general objects, potentially including non-rigid deformations. Furthermore, we will re-evaluate current experiments to include improvements in certain steps, such as the multi-part final adjustment (see multi-part reassembly experiments).

Regarding the object repair task, next is a list with the items we would like to cover in D5.8:

- We plan to evaluate the template-based retrieval and repair, and the subsequent alignment process.
- We plan to evaluate the impact of the use of additional information (surface classification as provided by T4.1) in our approach.
- We plan to evaluate the consecutive execution of symmetry-based completion to increase the prediction of missing geometry.
- We plan to include a user study to evaluate our algorithm from the perspective of the final user.

4 Evaluation of WP5

This section covers the evaluation of the systems delivered on M24 (D5.3). The evaluation of the integrated systems has a different focus compared to that of the work packages targeting the scientific objectives of the project. It must:

- Ensure the operational state of the delivered systems.
- Validate that the envisioned scenaria using the two platforms can be implemented.
- Demonstrate that the final tools and underlying workflows involved make a significant contribution to the objectives of the project, indicating a clear benefit from using them.

The following sections present the evaluation of the systems according to the scenarios and respective the criteria as described in the evaluation plan (D5.1). Please note that since this is the first evaluation report, whose primary purpose is to evaluate the scientific correctness of the approaches followed, the evaluation of the integrated systems only covers aspects that comply with the state of the integration on M24, according to the Description of Work. A full evaluation of the procedures implemented and the impact of the systems on the CH expert’s workflows is programmed for the final evaluation report.

4.1 Virtual Repair and Measurement Workspace Experiments

According to the evaluation plan, the following table summarizes the mapping of criteria to the related scenarios:

| | Plausibility | Robustness | Generality | Completeness | Automation | Responsiveness | Usability | Effectiveness |
|-----------------------------|--------------|------------|------------|--------------|------------|----------------|-----------|---------------|
| Collection Management Tools | | • | | • | | • | • | |
| Object Erosion | • | • | • | • | • | | • | • |
| Shape-guided reassembly | • | • | • | • | • | • | • | • |
| Object repair | • | • | • | • | • | • | • | • |
| Missing parts prediction | | • | • | • | • | • | • | • |

Table 23: Criteria for the evaluation of the VRMW workflows

According to the DoW, the first integrated version of the VRMW is an early version of the 3D workspace, where the automatic reassembly, repair, measurements and degradation simulation takes place. This version focuses on building a GUI, a working set of import/export tools, connectivity with the data sources, including collection management, logging, reporting and basic object manipulation.

The VRMW in its current state (M24), implements all the above and additionally, it has a working interactive reassembly pipeline ready, albeit with limited interaction with the pre-processing and facet classification sub-modules (fragment pre-processing), stages that are going to be fully accessible for user intervention, beyond the automated execution, which is already functional.

The object degradation simulator is also integrated with the VRMW executable at an API level (i.e. it is callable through the VRMW), although no visual output is fed back to the main workspace. Nevertheless, objects corresponding to the simulated epochs are generated and stored, which can be later loaded separately into the VRMW.

The object repair module is also integrated at an API level with the main interface,

but in its current state, it is not yet fully automated as a pipeline and cannot be tested in par with the host system.

4.1.1 System-wide Features

Apart from the dedicated usage scenarios, described in D5.1, three general criteria also apply to the system as a whole: robustness, responsiveness and automation.

4.1.1.1 Robustness. VRMW should be self-contained and fail-safe regarding accidental misuse of the interface and its satellite components, such as the reassembler and preprocessor servers.

The VRMW environment installation comes with all its dependencies either statically linked into the system or dynamically loaded from DLLs residing in the executable folder and has been tested in various independent systems that meet the minimum system requirements listed in D5.3. Special care has been taken not to require extreme system specifications to run VRMW. Furthermore, certain computationally heavy components, such as the reassembler, are completely separated from the main application and run as independent processes (servers). This also facilitates the maintenance and functional upgrade of these components during M25-M36, without interfering with the development of the VRMW itself.

The VRMW application requires that the reassembler and preprocessor servers are up and running for the reassembly pipeline to be operational but also for pre-processing fragments intended for object repair. The two servers are automatically restarted in case they are unexpectedly terminated.

4.1.1.2 Responsiveness. For interactive tasks, and all GUI-related operations, "this is a sub-criterion of the usability, requiring the timely or progressive feedback of an operation to avoid stalling the system. Lengthy processes should not withhold the entire system and have to progress in the background, so that other tasks can be completed simultaneously" (from D5.1).

All time-consuming operations in the VRMW system, from the loading of large data sets to lengthy calculations are threaded and asynchronous, either at the engine side (e.g. reassembly and repair engines), or at the integrated environment side (erosion, collection management), according to the system specifications (D5.2). This is especially useful when loading multiple fragments in the interface, each of which consists of many thousands of polygons. This is also the case for the reassembly, erosion and repair procedures, which can all take considerable time for high precision parameters and large datasets (e.g. many epochs for the erosion, many new, non-cached fragment combinations etc.), in the order of many minutes.

After evaluating the interface with lengthy operations, despite the fact that individual modules report the progress independently in the system log console (Figure 68) and the main application is never stalled, for lengthy operations, it was deemed that a progress estimator feedback is a useful addition and will be added in a future version of the interface.

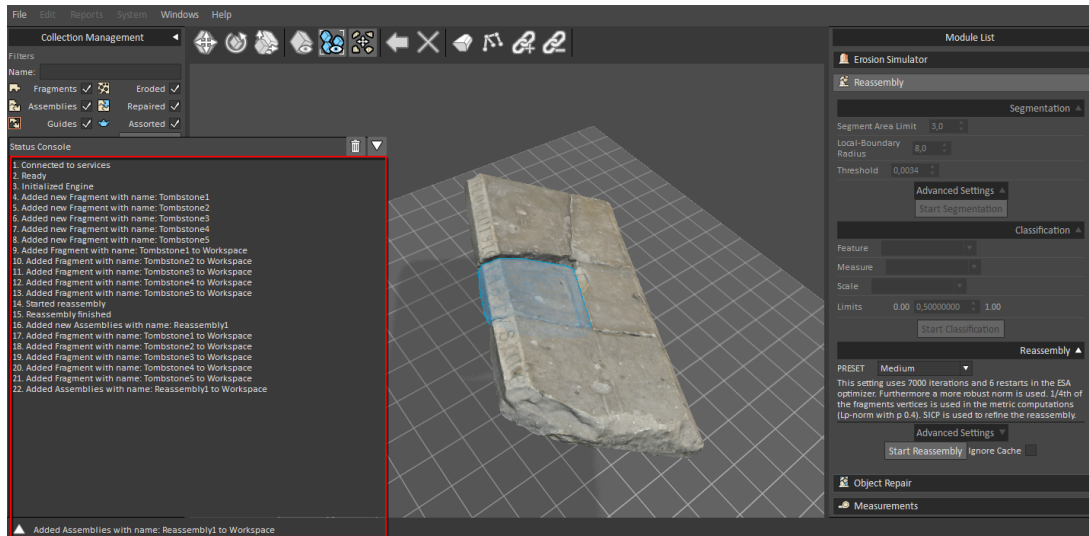


Figure 68: All asynchronous events are logged at accessible via the system console of the application.

4.1.1.3 automation.

Level of detail selection. Objects loaded for processing in the VRMW workspace may consist of many thousands of polygons. The accumulated payload of polygons to be rendered in each frame can quickly reach several millions of polygons, prohibiting interactive GUI response rates, at least on low-spec systems. Therefore, when multiple resolutions of a piece are available, they are detected and lowest is used for visualization, whereas the medium (when available) or high Level of Detail (LoD) is used during processing.

Visual aids. Providing a reference system during object manipulation was early on deemed important, since "floating" geometry is less intuitive to place in space and relate to other objects. Therefore a dynamically expanding grid was added as well as shadows to provide the visual cue of distance from it. Despite the fact that the grid extents are adjusted to fit the displayed geometry, the grid lines are bound to the coordinate system origin (they don't "slide" along with the grid's extents) so that they provide a fixed reference. The reference plane is also automatically adjusted for height, in order to always be beneath the objects.

Rendering. In order to provide as faithful and physically correct appearance of the displayed meshes, we employ a Cook-Torrance shading model (see Figure 69) with suitable modifications in the import format (OBJ) to also support metallic surfaces. We also support normal and material maps (specular attributes), for greater fidelity.

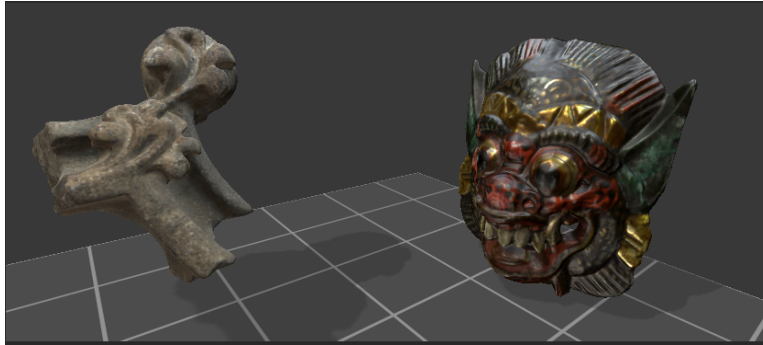


Figure 69: Shading in the VRMW supports a wide range of materials, from granular stone to polished wood and metals, via the Cook-Torrance model and support for material and normal maps.

Lighting. For the easy inspection of the data presented in the VRMW workspace, we have experimented with various photographic light configurations and ended up with a solution that would allow a) adequate lighting from all viewing angles, b) capture of edge details (i.e. avoid "flattering" the shading), c) adequate directional discrimination. 2 and 3 light setups, although typical for studio lighting, are not designed for multi-angle viewing, especially for inspection from underneath the objects. On the other hand, simply adding a fourth light flattened all shading, de-emphasizing details. What we came up with involved a combination of "artificial" under-lighting with sky illumination with subdued highlights: a world-oriented key light with shadows, 2 low-altitude peripheral blue-tinted overhead lights and a yellow-hued tertiary source below the objects, that could simultaneously provide adequate illumination for bottom surfaces and discrimination due to different hue. Using a first-person lighting setup (bound to the camera) was the initial choice, but turned out that due to the many surface discontinuities, the constantly changing illumination during inspection would be more distracting than helpful. Ambient occlusion is also enabled by default to emphasize contact shadows and crevices, but with a subdued presence so that features are not misinterpreted.

The distance and power of all (photometric) light sources are automatically adjusted to provide neutral and constant lighting, regardless of the geometric extents of the loaded objects in the workspace. Shadows were deemed necessary for experiments to provide distance cues, at least from the reference plane (grid).

4.1.2 Collection Management Tools

The fragments of objects to be reassembled, objects used in erosion experiments, clustered objects (assemblies), the results of an erosion simulation, the missing parts from a repair process, the completed objects or generic parts loaded into the VRMW are all accessible via the collection management sidebar. Selected meshes can be transferred to the main workspace for direct manipulation or participate in one or more of the workflows of the VRMW (erosion, reassembly, repair).

4.1.2.1 Robustness and Completeness. Meshes imported into the VRMW come in many different units, depending on the scanning software and tools used to process them. To be able to take into account the original measurements correctly when comparing surfaces and measuring distances, when importing a mesh, a dialog box intercepts the procedure and asks the user to define the units for the coordinates. The loaded coordinates are normally scaled according to the reference unit system so that all measurements are compatible. This function is temporarily disabled in order to fix certain issues with the grid sizing, but will be fully functional in the next intermediate version of the software.

Since the workspace must also support arbitrary objects that can be used either in the erosion simulation or as guides, there are no structural restrictions (connectedness, manifold etc.). New, assorted objects can be flagged as such during import. To demonstrate the generality of the collection management system, a set of highly atypical meshes is loaded and manipulated in the example of Figure 70.



Figure 70: Generality of the mesh manipulation tools: Non CH objects loaded into the woekspace.

In the current version of the software, due to some rendering API issues (selection buffer abstraction layer), selections performed in the collection management object list and the workspace are left unrelated. This can lead to some difficulties in the use of the software and a potentially unexpected behaviour, in terms of human workflow. This is a known issue, which will be rectified in the next version.

4.1.2.2 Responsiveness. To quickly begin working on a dataset, we decided that all geometry be lazily loaded, when and if required. For instance, the actual mesh of an object is only loaded when the user requests its transfer to the workspace from the collection list, and again, only the appropriate level of detail. The same goes for the various data buffers that need to be communicated to the attached engines to process the geometry; the necessary files are read whenever necessary and provided to the engine APIs. As an example, if a fragment's geometry (medium/high LoD) is already available to the reassembler server, it is never loaded and submitted upon a reassembly request.

4.1.2.3 Usability

Filters. The entries can be queried by name or type, using a set of filters in the top part of the collection sidebar (Figure 71). The type of an object is determined by the available data for it. For example, a mesh, when loaded to the workspace, is classified as "fragment", if the necessary binary data for its segmentation and facet classification can be located on any one of the searchable paths of the application (typically in the same folder, but not necessarily). Erosion results are accompanied by simulator-generated data that can identify them as such. Results from the repair module are queried by type directly from the repair engine and are directly flagged accordingly. The same holds for the reassembly results, only in this case, an entire group of *existing* fragments creates a compound object, an *assembly*, i.e. a virtual object in the workspace, which can be released and its participating fragments disassociated.

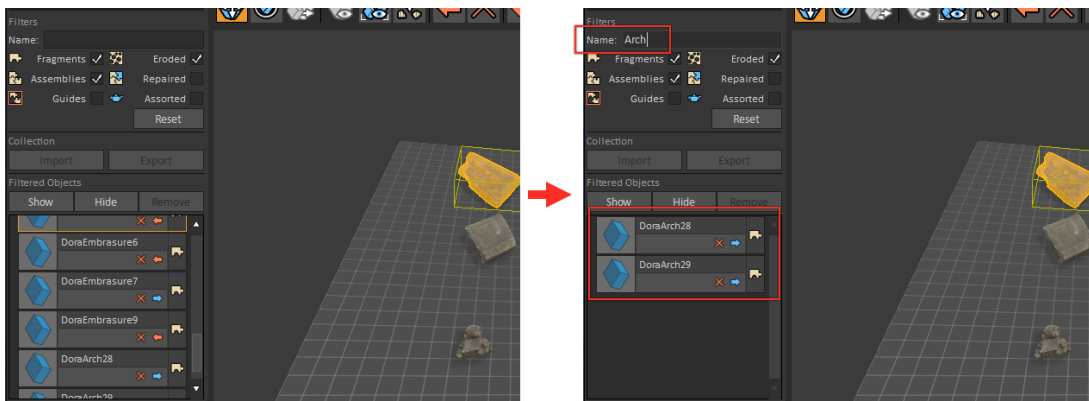


Figure 71: Filtering of a collection to locate a particular series of fragments.

Selection and highlighting mechanisms. Especially in the case of cluttered environments, highlighting certain objects and maintaining visual contact with the active selection can be challenging. Initially, highlighting of parts only involved a different shading for the selected mesh, with an additional silhouette delineation. However, very often selected objects became obscured by others, closer to the viewer. To solve this issue, we borrowed a common trick done in computer games, where very often objects of interest (such as enemies) must be highlighted through walls and other obstacles: highlighted geometry is rendered in a separate pass "over" the rest of the geometry and blended according to relative depth, as shown in Figure 72.

In the case of multiple selection, we evaluated the visual separation of the parts under different viewing angles, affecting the depth complexity of the view (Figure 73). In most cases the combined shaded and lit wireframe surfaces provided adequate enhancement of the shapes. The post-multiplied transparency mode over the regular shading pass provided good visibility of the outlines, too. Only in situations with many overlapping objects (> 4) it was difficult to discern the selected shapes, but this is not a typical case, since a change of viewing angle typically eliminates this problem for most realistic examples.

Transformations. The current implementation requires that the geometric trans-

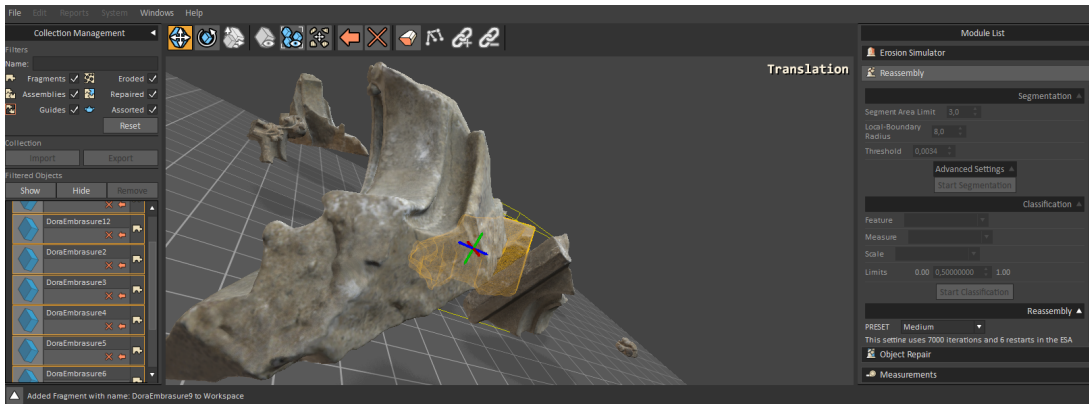


Figure 72: An overlaid see-through highlighting mechanism is implemented to allow selection visibility of hidden parts.

formation procedure is performed in two steps, or in fact three, if the part selection is also considered: First the mesh is selected, then the axis of the transformation is selected by picking the right widget on the transformation gizmo and finally the transformation itself is executed by dragging either the piece or the axis. Initially, this was considered a good idea so that selection and actual movement were decoupled for more precision. To facilitate precise transformations, all changes are clearly displayed on the workspace during the object(s) manipulation (Figure 74). After working with the interface for some time though, this movement breakdown was deemed ineffective and in the next version we are going to adopt the transformation paradigm of 3DStudio Max, which is also common in other 3D geometry manipulation software.

Automatic object arrangement. When objects are loaded into the workspace, they may initially overlap each other, since their original (object-space) coordinate

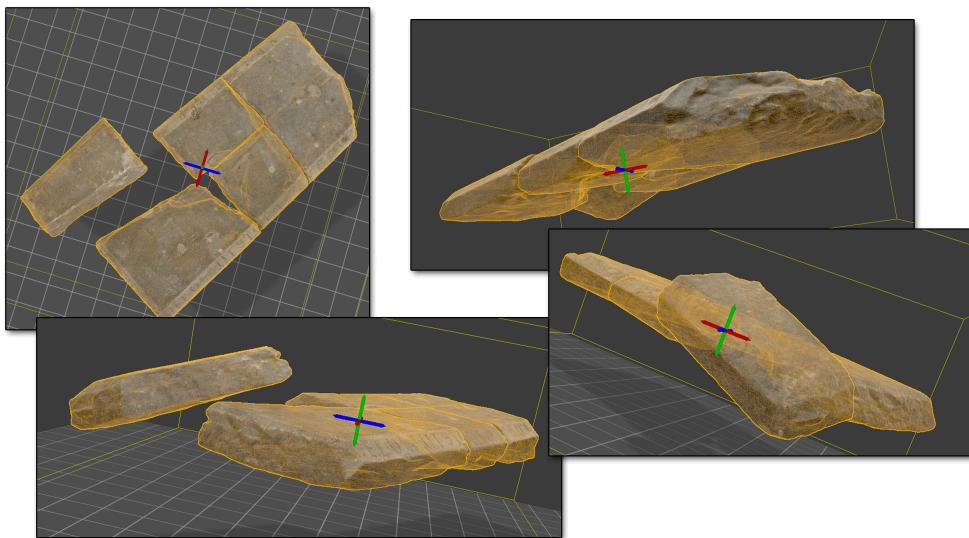


Figure 73: Visibility and part separation inspection for multiple, overlapping selected objects.

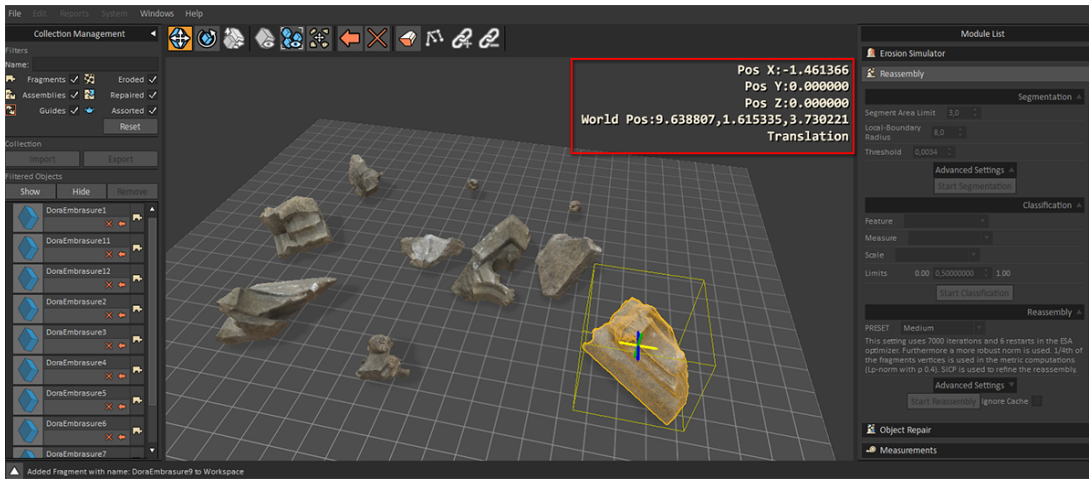


Figure 74: Indication of the exact object transformation offset is provided to accurately position object in the workspace.

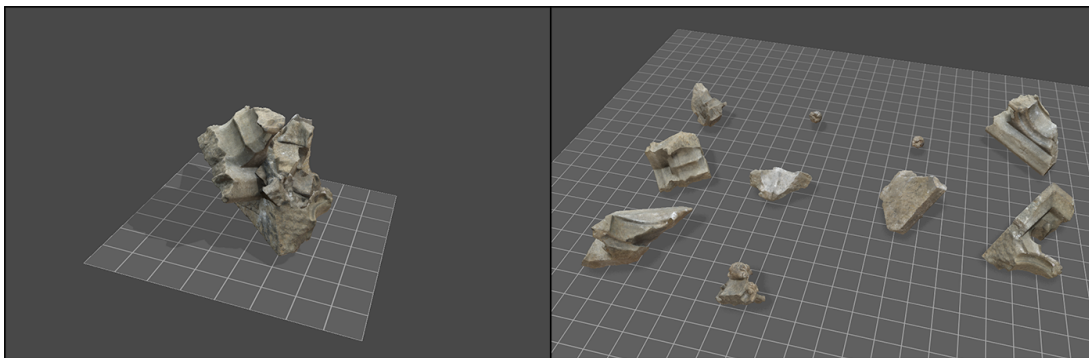


Figure 75: Automatic object separation. Left: objects loaded into the workspace initially may overlap each other. Right: The automatic separation tool spreads out the pieces to facilitate inspection and picking.

system is defined by the scanner coordinate system, which is more or less the same for most parts (Figure 75 - left). Manually having to arrange the various meshes so that they are distinguishable and possible to select individually, quickly became very time consuming and frustrating. So, we implemented an automatic separation function that evenly spaces the loaded pieces on a grid according to the diameter of the largest part (Figure 75 - right). Although this is not the most sophisticated algorithm for automatic arrangement, it is sufficiently effective for this task.

Finally, in our design, we have also decided to represent the loaded geometry by an icon, generated from rendering the geometry once, the first time ever the object is loaded into the workspace. This is necessary as in many cases the name or code of an object is not very descriptive (especially true for fragments). This will be implemented in a future version of the VRMW.

4.1.3 Object Erosion

We have only evaluated the geometric aspect of the erosion simulator (see **Evaluation of WP3** above and Section 4.3). The present version of the erosion simulator has three functionality modes:

- *mode 1*: a user-defined geometric offset is applied to every surface vertex. This is useful if the amount of erosion is externally provided.
- *mode 2*: a geometric offset based on the Gauri model of marble erosion is applied to every surface vertex. The user provides the input parameters to the Gauri model: material parameters (assuming homogeneous stone material) and environmental parameters (see D3.4 - First degradation simulator version).
- *mode 3*: a different geometric offset is applied to every surface vertex, based on the mineral properties of the vertex, which are available on the associated mineral map. Although this functionality is available, a physicochemical model has not yet been developed to support it.

4.1.3.1 Effectiveness. The erosion simulator is unique in that it presents the possibility of applying simulated erosion to cultural heritage monuments, which cannot be placed in erosion chambers. The only requirement is accurate 3D scanning of the monuments (modes 1 and 2 above) and a registered mineral map for mode 3. In our experience, 3D scanning takes of the order of a few hours of field work (for relatively small scanning areas of the order of a square meter) plus a few more hours of post-processing. The mineral map can be provided by appropriate chemical scanning technologies (e.g. QEM) or by a simulated statistical model (e.g. the Stone Builder component of our software). Once a mesh is available, the erosion simulator can apply an amount of erosion dictated by the user in a matter of seconds to hours, depending on the complexity of the mesh.

4.1.3.2 Usability. The erosion simulator requires minimal interaction with the user. After specifying the input mesh, the optional mineral map and the erosion parameters, the simulator begins the erosion process and the user is notified when the process is complete. Currently, The eroded object can then be visualized and measurements can be taken on it using the tools of the VRMW.

4.1.3.3 Automation level. Once the object to be eroded has been selected and the erosion parameters have been input, the erosion simulator works totally automatically. This is particularly important, since the erosion simulation may require considerable time. When the results are ready, the VRMW is notified accordingly.

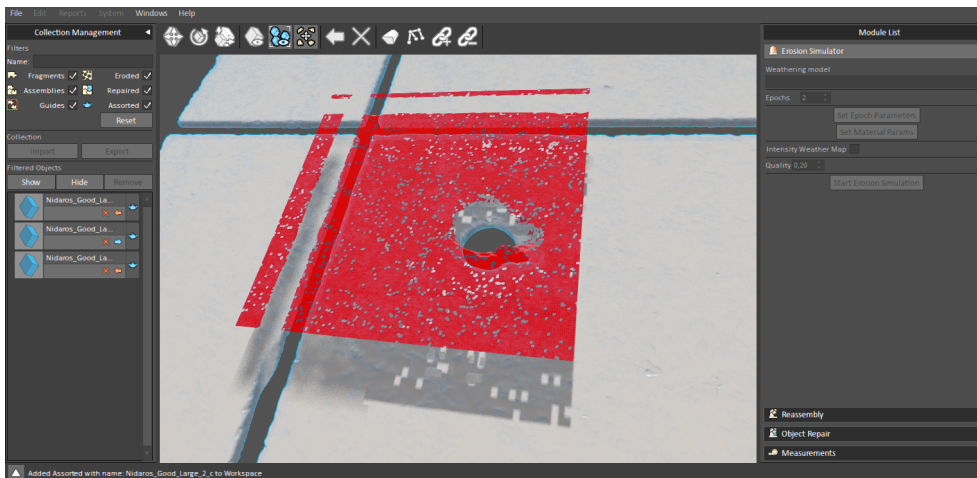


Figure 76: Results from the Erosion Simulator imported for inspection in the VRMW.

4.1.3.4 Completeness. The data input functionality for object selection is complete. The functionality for erosion parameter specification is complete. The data output functionality to allow visualization, measurement and storage of the eroded object is present. Currently, for demonstration purposes, the erosion simulation runs using a predefined set of parameters stored in external configuration files, although all input settings for the simulation epochs etc. have been implemented in the interface. The next versions will also include direct visualization of the results in the workspace. In the current version simulation results are imported as assorted files (Figure 76).

4.1.3.5 Generality. The erosion simulator is applicable to any properly 3D scanned object but the erosion model currently handles marble and geometrical erosion only.

4.1.3.6 Robustness. At the moment the erosion simulator software is deterministic. Given the same input mesh and the same erosion parameters, the output will be the same.

4.1.3.7 Plausibility. The eroded surfaces are visually convincing and the produced meshes are of a similar quality to the original meshes.

4.1.4 Shape-guided Reassembly

4.1.4.1 Plausibility. The plausibility of results is highly dependent on the input dataset and is also affected by manual constraints that the user may impose on the fragments. For a complete set of experiments on the reassembly of objects from fragments, see Section 3.1. The manual constraint procedure as a GUI-driven workflow is not yet available in this version of the VRMW and will be evaluated in D5.8.

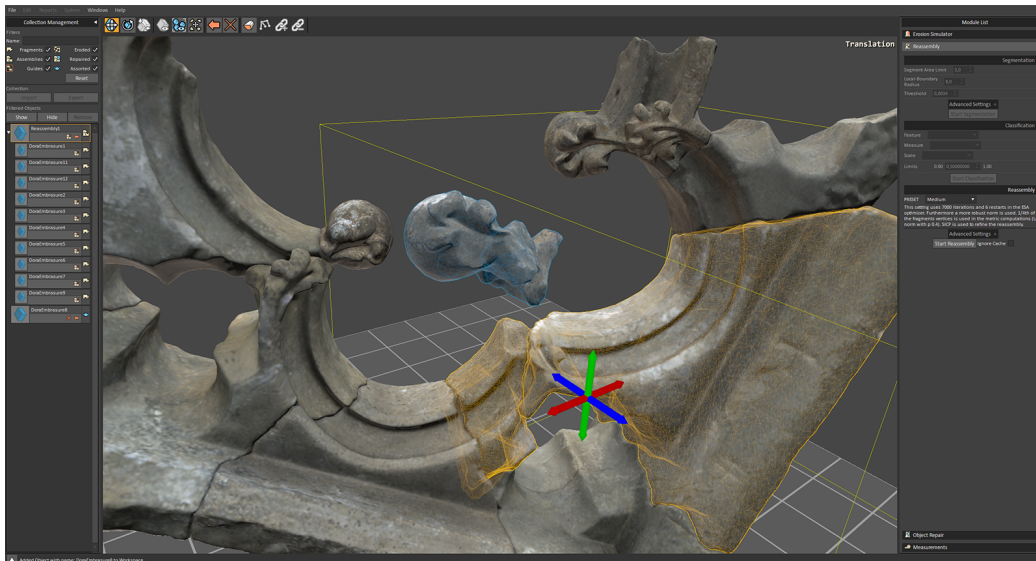


Figure 77: Problematic part (highlighted in blue): The part does not share a descriptive enough contact surface with its connected neighbours (highlighted in yellow) and the dominant features on intact surface are unusable in an alignment attempt.

In general, we observed that given fragments with adequate fractured facet overlap ($\sim > 20\%$ of the largest piece's fractured surface) and low deterioration of the fractured facets, pairwise and multi-part assemblies (including mixed puzzles) produced plausible results. For smoothed out surfaces, the pre-processing of the fragments sometimes produced false negatives, which required manual labelling of the problematic facets as "fractured". Most of these problematic fractures did not cause any problems, after manually labelling them, during the reassembly. Still, in some cases, problematic facets caused slippage or a completely ambiguous matches due to a small area of overlap (see Figure 77).

Typically, there are two ways to correct the problem: a) manual black-listing of a (wrong) solution, or enforcement of a user-selected alternative (white-listing) and b) prioritization of external features over contact surface registration. This last approach however can only work when there are usable features on the problematic part. Figure 77 shows a case where biasing the solution towards using f-curves cannot help, since the detected features on the piece cannot contribute to the solution. This was a clear indicator that for certain, usually small parts, or isolated pieces with small area of contact, we would have to resort to a different strategy, such as a symmetry-based approach (investigated in Y3).

4.1.4.2 Effectiveness. For the general algorithmic effectiveness of the reassembly pipeline, please refer to the experiments in Section 3.1. With regard to the effectiveness of the user workflow, since no manual constraints are allowed so far by the GUI (to be implemented in the next version), this is a straightforward procedure; parts are loaded in the user interface (see Collection Management) and a reassembly

is triggered on a selected subset. Upon the completion of the task, VRMW proposes a rearrangement of the selected pieces and the user may iterate through different accuracy settings in order to obtain a satisfactory result. The user may also experiment with manual placement of the fragments in the virtual workspace and inspect the reassembly from any viewpoint. For datasets for which no manual black-listing or white-listing is required, the process is self contained and rather simple. This is also attributed to the fact that for new or custom meshes imported into the workspace, the procedure to convert them to a *fragment* representation is very simple, by following the two steps of segmentation and classification either with the pre-configured settings or by choosing custom ones. We are also considering the option to silently and automatically converting the imported meshes to fragment data, when a particular mesh is requested to participate in a reassembly calculation.

4.1.4.3 Robustness. For the general algorithmic robustness of the reassembly pipeline, please refer to the experiments in Section 3.1. The reassembly procedure has been shown to be robust in terms of outliers and missing parts. Key to this is the three-fold validation of the generated assembly: first, the contact-based geometric alignment uses the robust registration algorithm developed for this task and presented in [MAP15] and ². If no usable contact area is found between the fragments, the latter are compared using detected external features. Third, the pairwise results participate in the multi-fragment optimization, where unmatched fragments can be properly aligned due to indirect linkage. Penetration testing rejects incompatible fragments and the final registration corrects any minor misalignments.

In terms of implementation and integration of the GUI with the pre-processing and re-assembly engine, The design and implementation of the reassembly and preprocessing modules as remote services, enables the VRMW (as a client) to run even on low-spec machines, by accessing the two procedures on a different networked and potentially centralized host. Furthermore, this design decision enables the future implementation of a web-service-style paradigm or the implementation of a client for reassembly on a diverse set of platforms, such as tablets running Android or iOS. The client - server architecture is designed in such a way that there are never stale data on either side; for any parameter change on the client side, the reassembler checks for the corresponding cached data and if there is a mismatch on a parameter that affects the validity of the results (e.g. re-segmentation), the server-side data are re-calculated and the result of the new reassembly posted back to the client.

4.1.4.4 Generality. For details about the generality of the underlying methods used in the object reassembly, please refer to Section 3.1. In this section we have shown that the method is general enough to operate on both well-preserved fragments and heavily eroded ones, a wide range of different shapes (ranging from pottery to

²P. Mavridis, A. Andreadis, G. Papaioannou, Efficient Sparse ICP, accepted for presentation at GMP 2015. To appear in Elsevier CAGD (minor revision pending).

bulky stone parts), as well as mixed puzzles.

All procedures can be applied from scratch to plain meshes and do not require a specific kind of input, since all stages that are required for the conversion of a simple mesh to a fragment in order to be part of a reassembly procedure are built in the VRMW. This particular feature is important in order to enable the independent evaluation of the software from 3rd parties that do not have access to our repository of fragment data, and may have diverse types of parts to examine. If the required auxiliary data for a particular stage are not yet available the only option open to the user is the missing task. For example, the reassembly cannot start, when one or more of the selected parts are not yet segmented or classified.

4.1.4.5 Automation. Using the default settings or one of the reassembly presets (see Usability, next), an entire reassembly experiment can be performed from the click of a button, for a set of selected parts in a collection. In the case of a bad match, parameters can be readjusted to allow for outliers or more iterations, increase the measurement accuracy etc, or again rely on a more accurate preset to correct the solution. If, for the reasons explained in the Plausibility criterion, there are still problematic combinations proposed, the user must resolve to manually linking or prohibiting a connection between two fragments. This operation is not yet supported by our system through the GUI, although it is supported by the reassembly engine.

One important thing to note is that the reassembler and segmenter servers are completely self-contained and there is no need for the data to pre-exist on both the client and the server. For every operation signalled by the client on a mesh, the server first checks whether the geometry is also available on the server (obviously, when they are both hosted on the same machine, this is always the case). If not, the geometry, complete with all referenced assets is transferred to the server.

4.1.4.6 Usability. We have observed from performing certain experiments with reassembly in the VRMW that the independent careful adjustment of the related parameters is neither user-friendly, nor necessary. More precisely, in an event of a failed reassembly, most parameters that can be changed make the algorithm error margins stricter, enable more iterations etc. and in general provide a time over quality trade-off with a simultaneous change of parameters affecting all aspects of the procedure. Instead of having the user independently control these parameters, we have grouped certain parameter configurations under convenient high-level settings:

- **Interactive.** This setting performs a rough initial alignment based on PCA and directly uses ICP to refine the solution (no global registration step). This setting does not perform any sophisticated alignment operations and is therefore bound to succeed only for "clean-cut" and well-structured fractures. Therefore, it is the less robust and generic setting.

- **Fast.** Uses global registration with a small number of iterations and sub-sampling of the surface vertices. ICP is used for the final reassembly refinement. More robust setting than the interactive one, since it uses global search. Still quite fast.
- **Medium.** Uses global registration with a higher maximum number of iterations and denser sampling of the surface vertices. Furthermore, outlier-suppressive metrics are enabled (slower convergence) and Sparse ICP is used for the final pairwise alignment. A good trade-off between speed and quality.
- **Fine.** Uses global registration with a higher maximum number of iterations and all surface vertices. It uses outlier-suppressive metrics closer to ℓ_0 ($p = 0.2$) and Sparse ICP. This is the highest quality pre-defined setting.
- **Custom.** "Expert mode" configuration, where all the method's parameters are open for experimentation.

Having the above presets available, facilitates the use of the reassembly procedure in the case of non-expert users and allows quick multiple parameter changes for expert user experimentation. The detailed parameters only appear in the custom preset, as shown in Figure 78.

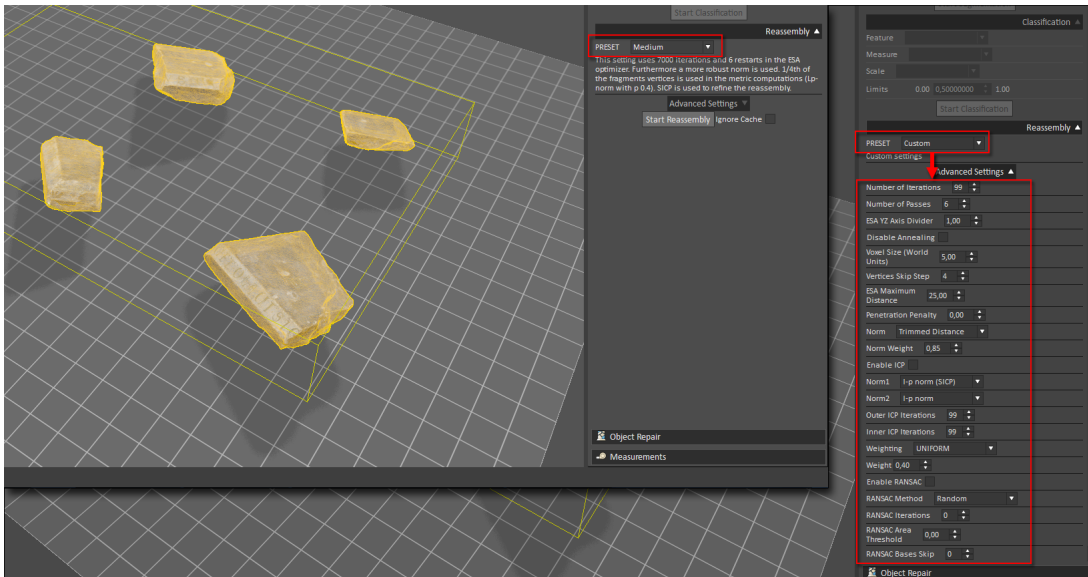


Figure 78: Detailed reassembly parameter settings are enabled only in the custom preset. 4 pre-defined presets are provided for easier reassembly control for non-experts.

4.1.5 Object Repair

According to the Evaluation Plan D5.1 the object repair modules are evaluated not only in isolation, but also as part of the VRMW integrated system. At the time of the first evaluation report, not all repair components have yet been integrated

into the system, hence not all evaluation criteria listed in D5.1 can be meaningfully evaluated at this time as regarding to the integrated system. We note that most of the per module evaluation detailed in Section 3.2 will carry over and stay valid for the integrated system. We next detail the evaluation criteria and summarize the findings where applicable.

The criteria for evaluation in this category comprise the following and are discussed regarding *symmetry-based* repair.

4.1.5.1 Effectiveness of the repair solution regarding the output of whole models which are meaningful to an Archaeologist in a respective workflow. According to the evaluation, our symmetry-based repair methodology can output whole models, up to a limitation with respect to missing geometry that spans across the symmetry plane. We conclude from the experiments in Section 3.2 that the method produces smooth transitions between the original and predicted parts, preserving small details like e.g., feature lines. By definition, the method provides a result showing the same level of detail for both the original and the predicted parts. The method is also able to predict larger missing parts. Hence, we consider the resulting objects per se plausible. Yet we also want to consult with Archaeologists on the plausibility in specific Archeology repair workflows, and report in D5.8.

4.1.5.2 Usability regarding user input requirements. As detailed in Section 3.2, the symmetry-based repair operates automatically. Needed user input is restricted to expert parameter setting; standard parameter settings have been found to provide reliable results across a larger number of objects. Hence required user input is minimal.

4.1.5.3 Responsiveness of the repair process. As detailed in Section 3.2, the symmetry-based repair requires time in the range of minutes to repair appropriately preprocessed objects. While this is not interactive time, the response of the system can be considered sufficient for users. Especially as compared with the time required for the digitization, the time needed for repair is only a small fraction of the overall time.

4.1.5.4 Completeness of the module user interface in the VRMW. So far, the module UI provides all necessary user input widgets, including setting of parameters and visual inspection of the repaired object.

4.1.5.5 Generality of the repair method. This criteria regards the types of objects which can be processed. Our method is applicable to repairing objects of symmetric property. As shown in Section 3.2, this applies to a significant part of the

realistic test data sets studied in PRESIOUS, including the Embrasure, Tombstone, Column Base, and Fountain objects. The method is applicable to objects which show local features, but it also gives satisfactory results for objects low on local features (e.g., Tombstone).

4.1.5.6 Robustness of the repair. As the results in Section 3.2 show, the method can be considered robust with respect to the mesh resolution of the objects, where the degree of completion only slightly degrades with lower resolution versions.

4.1.5.7 Plausibility of the results. The method is applicable to symmetric shapes and according to observations on the test data, provides plausible results. Specifically, we observed that the merging step produces smooth transitions between the repaired elements and the fragment shape, thereby creating a plausible whole shape with little non-continuous areas. This implies the continuity of feature lines, as obvious e.g., from the Dora Embrasure or Fountain objects (see Section 3.1.4).

4.1.6 Missing parts prediction

We observe that as the method provides plausible and smooth repaired shapes, also the predicted missing parts fit the objects and are expected to be a suitable basis for the physical restoration using 3D printing technology.

4.2 Predictive Digitization Platform

Effectiveness. For each partial query, PDP execution time ranges from 30 to 50 seconds, depending on the number of model vertices. Taking into account that scanning with Breuckmann Optoscan scanner requires approximately 10-20 seconds, a total of 40-70 seconds is required for overall predictive digitization. This process is connected approximately 20 times faster than the traditional digitization on a turntable, which requires approximately 20 minutes. The accuracy of the predictive digitization, as well as the performance of individual components, have been evaluated in Section 1. Both partial retrieval and model reshaping involve novel methods, which have either been published (D1.2) or are in the process to be submitted for publication in peer-reviewed international journals and conferences.

Usability. PDP GUI is intuitive and straightforward to use, as verified by a cultural heritage expert. It only displays the necessary cues when needed. For example, the 'Predict' option is invisible to the user, until the results of partial retrieval are available. This minimal design allows PDP to be more accessible to novice users, requiring almost no training at all. Still, options adjustment in this version is not

trivial, involving numerous parameters associated with model reshaping. This sets a point for improvement in the final version.

Responsiveness. Prediction digitization is a linear process, which starts from a partial 3D scan and ends with a complete model, without much user involvement. The partial retrieval stage usually takes about 15 seconds and the model reshaping stage roughly another 30 seconds. The partial retrieval stage usually takes about 15 seconds and the model reshaping stage roughly another 30 seconds. These time values have been obtained with the hybrid Matlab/C++ architecture, on an Intel Core i7 workstation, operating at 3.5 GHz with 16 GB of RAM

Automation level. User intervention is intentionally kept to a minimum, considering that PDP acts as an application assisting the digitization device. At the current stage of development, 3D input is automated but complete model prediction requires two clicks: 'query' and 'predict'. 'Query' initiates retrieval of similar 3D models from the database which results in a ranked list. The user is allowed to manually pick a 3D model from the ranked list and 'Predict' leads to a complete object, fusing the partial data from the digitization device and the complete data from the database. This user interaction is aimed to enhance retrieval accuracy. However, PDP is also capable of working in a completely automated fashion, proposing predicted results to the user.

Generality. Apart from the pottery dataset, the large-scale partial retrieval benchmark of SHREC 2013 has been used to evaluate the partial retrieval module. In addition, the model reshaping modules have been evaluated on objects derived from the publicly available Human 3.6M dataset (<http://vision.imar.ro/human3.6m/description.php>) [IPOS14], [CI11] (see Section 1).

Robustness. The performance of predictive digitization is approximately stable in cases of alterations in the exact position of the scanning device. In the case of two sets of partial queries, which were obtained with different positions of the Breuckmann Optoscan scanner, the variance in total prediction error e is less than 3%. Taking into account that the field of view of Breuckmann Optoscan scanner is rather wide, one partial scan is sufficient.

4.3 Deviations and Corrections with Respect to WP5 Evaluation Plan

With regard to the reassembly workflows and the collection management, there were no major deviations from the evaluation plan. We intentionally prioritized the development of a reassembly pipeline (even a reduced-functionality one) over the implementation of measurement tools, for review and demonstration purposes. The GUI part of these tools will be implemented in Y3. The underlying functionality for computing residuals and matching errors is already implemented in order to evaluate our methods in Section 3.1.

With respect to the Object Erosion items (see D5.1 Evaluation Plan sections 4.4.2.2, 4.4.2.3 and 4.4.2.4) some deviations were necessary, as per the deviations reported in the evaluation of WP3 in the present document. Specifically:

- Object Erosion: Surface Geometry and surface physicochemical data experiments (section 4.4.2.3 of D5.1), have not yet been fully analyzed as the second round of data from the erosion slabs came in early 2015. Although surface physicochemical functionality has been added to the erosion simulator, it has not yet been thoroughly evaluated and cannot thus yet be evaluated as part of the VRMW in WP5. This functionality is not expected to significantly alter the interface however.
- Object Erosion: Surface Geometry and volume physicochemical data experiments (section 4.4.2.4 of D5.1), cannot be performed as our measurement modalities did not produce meaningful volumetric physicochemical data on the eroding slabs. Volume physicochemical erosion functionality has thus not been included in the erosion simulator and cannot thus be evaluated as part of the VRMW in WP5.

4.4 WP5 Items Remaining to be Evaluated in D5.8

Regarding the reassembly experiments, the manual constraint procedure as a GUI-driven workflow is not yet available in this version of the VRMW and therefore a comprehensive testing of all reassembly possibilities will be evaluated in D5.8. The detailed facet selection mechanism required for the explicit white-/black-listing of combinations and the visualization of the classification outcome via the main interface are not yet implemented. However, this interaction functionality is scheduled for Q1-Q2 of Y3, as it is also necessary for a system-wide demonstration of VRMW for publication purposes.

Finally, for the VRMW, we will prepare an additional "end-to-end" experiment, starting from fragments and resulting to the missing parts of a repaired reassembly. In fact, we have performed such experiments and the results are in the WP4-related evaluation sections, but the algorithms have been executed individually, outside the integrated environment. We were not able to map these procedures to an experiment via the GUI, since certain steps are not yet integrated.

A Parameters for the reshaping module (WP2)

The reshaping operation involves a series of algorithms for rigid and non-rigid registration that requires some tuning. The parameters used for the first experiment in the WP2 pipeline evaluation are:

```
-passes 6 -maxtrans 10 -n 5000 -udf -verbose -norm 2 -rweight 0.9 -ransac wide 2  
20000 -fparams 0.5 0.5 25.0 -vsize 1.0 -rskip 2 -oneway -rlimit 30000 -step 4 -rstep  
4 -sicc 1 1 40 20 6 0.25 -angle 1 -nlopt 0 -nonrigid pauly cg 20 0.99 0.6 0.1 1 0 0.0  
40 7 100 0.3 -msampling 1.0 -cutoff 200
```

These parameters control various aspects of the reshaping framework, such as the volume resolution for the VDB (-vsize) or the maximum iterations in the simulated annealing (-n). The meaning of these parameters was briefly explained in D2.2. For the second experiment, we have changed the distance threshold (shown in bold) from 7 to 40. We expect that as we continue to develop our algorithms, in the final system there will be no need to tune any parameters.

B Benchmark of Fragmented Objects

The list of objects from the Hampson dataset considered in our benchmark and the number of fragments is presented in the following table:

| | | | |
|----------------------|---------------------|------------------------|--------------------|
| Ark_HM_10.Lo-0(7) | Ark_HM_218.Lo-0(6) | Ark_HM_447.Lo-0(8) | Ark_HM_6.Lo-0(7) |
| Ark_HM_110.Lo-0(5) | Ark_HM_223.Lo-0(10) | Ark_HM_45.Lo-0(7) | Ark_HM_704.Lo-0(8) |
| Ark_HM_112.Lo-0(6) | Ark_HM_234.Lo-0(8) | Ark_HM_46.Lo-0(7) | Ark_HM_720.Lo-0(7) |
| Ark_HM_122.Lo-0(8) | Ark_HM_240.Lo-0(8) | Ark_HM_480.Lo-0(8) | Ark_HM_721.Lo-0(9) |
| Ark_HM_1245B.Lo-0(8) | Ark_HM_263.Lo-0(7) | Ark_HM_489.Lo-0(8) | Ark_HM_728.Lo-0(9) |
| Ark_HM_1246.Lo-0(6) | Ark_HM_267.Lo-0(10) | Ark_HM_48.OBJ.Lo-0(6) | Ark_HM_73.Lo-0(9) |
| Ark_HM_1247.Lo-0(7) | Ark_HM_270.Lo-0(6) | Ark_HM_521.Lo-0(9) | Ark_HM_77.Lo-0(7) |
| Ark_HM_1260A.Lo-0(8) | Ark_HM_285.Lo-0(7) | Ark_HM_538.Lo-0(6) | Ark_HM_786.Lo-0(6) |
| Ark_HM_1260B.Lo-0(5) | Ark_HM_288.Lo-0(8) | Ark_HM_546.Lo-0(6) | Ark_HM_81.Lo-0(8) |
| Ark_HM_1260C.Lo-0(8) | Ark_HM_31.Lo-0(9) | Ark_HM_554.Lo-0(8) | Ark_HM_XXb.Lo-0(9) |
| Ark_HM_1300.Lo-0(6) | Ark_HM_338.Lo-0(8) | Ark_HM_586.Lo-0(8) | BEY_1415.Lo-0(8) |
| Ark_HM_1301.Lo-0(8) | Ark_HM_340.Lo-0(7) | Ark_HM_595.Lo-0(7) | BR_1503.Lo-0(8) |
| Ark_HM_1302.Lo-0(8) | Ark_HM_341.Lo-0(7) | Ark_HM_597.Lo-0(9) | MM_2485.Lo-0(9) |
| Ark_HM_158A.Lo-0(7) | Ark_HM_375.Lo-0(8) | Ark_HM_612.Lo-0(7) | RL_907.Lo-0(7) |
| Ark_HM_194.Lo-0(8) | Ark_HM_38.Lo-0(5) | Ark_HM_641.OBJ.Lo-0(7) | SV_4337.Lo-0(11) |
| Ark_HM_202.Lo-0(9) | Ark_HM_418.Lo-0(6) | Ark_HM_653.Lo-0(8) | TO_2757.Lo-0(8) |
| Ark_HM_208.Lo-0(6) | Ark_HM_419.Lo-0(7) | Ark_HM_661.Lo-0(7) | UN_5044.Lo-0(9) |
| Ark_HM_216.Lo-0(9) | Ark_HM_424.Lo-0(8) | Ark_HM_670.Lo-0(9) | UN_542.Lo-0(6) |
| Ark_HM_217.Lo-0(8) | Ark_HM_435.Lo-0(8) | Ark_HM_682.Lo-0(6) | UN_5526.Lo-0(6) |

Table 24: Objects from the Hampson dataset used in our benchmark. The number of fragments is specified in parenthesis.

References

- [AMCO08] AIGER D., MITRA N. J., COHEN-OR D.: 4points congruent sets for robust pairwise surface registration. In *ACM SIGGRAPH 2008 Papers* (New York, NY, USA, 2008), SIGGRAPH '08, ACM, pp. 85:1–85:10.
- [BTP13] BOUAZIZ S., TAGLIASACCHI A., PAULY M.: Sparse iterative closest point. *Computer Graphics Forum (Symposium on Geometry Processing)* 32, 5 (2013), 1–11.
- [CI11] CATALIN IONESCU FUXIN LI C. S.: Latent structured models for human pose estimation. In *International Conference on Computer Vision* (2011).
- [GH97] GARLAND M., HECKBERT P. S.: Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., pp. 209–216.
- [IPOS14] IONESCU C., PAPAVALA D., OLARU V., SMINCHISESCU C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7 (jul 2014), 1325–1339.
- [KH13] KAZHDAN M., HOPPE H.: Screened poisson surface reconstruction. *ACM Trans. Graph.* 32, 3 (July 2013), 29:1–29:13.
- [MAM14] MELLADO N., AIGER D., MITRA N. J.: Super 4pcs fast global pointcloud registration via smart indexing. *Computer Graphics Forum* 33, 5 (2014), 205–215.
- [MAP15] MAVRIDIS P., ANDREADIS A., PAPAIOANNOU G.: Fractured object reassembly via robust surface registration.
- [Mus13] MUSETH K.: VDB: High-resolution sparse volumes with dynamic topology. *ACM Transactions on Graphics (TOG)* 32, 3 (2013), 27.
- [SMB*13] SIIPIRAN I., MERUANE R., BUSTOS B., SCHRECK T., JOHAN H., LI B., LU Y.: SHREC'13 track: Large-scale partial shape retrieval using simulated range images. In *Proc. 3DOR* (2013).
- [SMB*14] SIIPIRAN I., MEROUANE R., BUSTOS B., SCHRECK T., LI B., LU Y., JOHAN H.: A benchmark of simulated range images for partial shape retrieval. *The Visual Computer* (2014).
- [WLVG07] WEISE T., LEIBE B., VAN GOOL L.: Fast 3d scanning with automatic motion compensation. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on* (June 2007), pp. 1–8.

- [ZC01] ZHANG C., CHEN T.: Efficient feature extraction for 2d/3d objects in mesh representation. In *Image Processing, 2001. Proceedings. 2001 International Conference on* (2001), vol. 3, pp. 935–938 vol.3.