



Predictive digitization, restoration and degradation assessment
of cultural heritage objects



D4.4 – Reassembly and Object Repair Methodology 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)
Reassembly and Object Repair Methodology Report	
Period Covered	from 10.02.2013 to 30.09.2015
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 (Public)

Address
O. S. Bragstads plass 2 E
Gløshaugen, NTNU, NO-7491 Trondheim,
Norway

<http://presious.eu>

Contact person
Karelle Gilbert-Soni
karelle@ime.ntnu.no
+ 47 735 91481

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	M 30
Deliverable Number	D4.4
Deliverable Name	Reassembly and Object Repair Methodol-ogy Report
Type	Document (\LaTeX)
Deliverable Version	1.0
Status	Final
Associated WP / Task	WP4
Author(s)	G. Papaioannou (AUEB-RC) R. Gregor, I. Sipiran, T. Schreck (UKON) M. Savelonas, F. Arnaoutoglou, K. Zagoris, P. Theologou (ATHENA-RC)
Project Officer	Philippe Gelin
Abstract	This document contains the Reassembly and Object Repair Methodology Report, summarizing the methodological results achieved within WP4.
Circulated to Participants	25. September, 2015
Read by Participants	29. September, 2015
Approved by General Assembly	30. September, 2015

Document Revision History

Date	Version	Author/Editor/Contributor	Summary of Changes
08.07.2015	0.1	T. Schreck	Initial draft, document structure
16.07.2015	0.2	G. Papaioannou	Object Assembly section
23.07.2015	0.3	I. Sipiran	Object Repair section
07.08.2015	0.4	R. Gregor	Object Repair section
08.08.2015	0.5	T. Schreck	Object Repair section and Conclusions
30.08.2015	0.6	I. Sipiran	Revision of Technical Details for Symmetry-based Completion and Scale Alignment
21.09.2015	0.7	R. Gregor	Revision of Technical Details for Template-based Retrieval
29.09.2015	1.0	R.Gregor	Final Version

Contents

1	Introduction	8
1.1	Scope of the Methodology	8
1.2	Defects of CH Artifacts	9
1.3	Taxonomy of 3D Shape Repair Methods	11
1.3.1	Modeling / Assembly	11
1.3.2	Reassembly	12
1.3.3	Inpainting	12
1.3.4	Symmetry-based completion	13
1.4	Comparison of Similarity-Based Repair Methods	14
1.5	Proposed Workflow for Automated Reconstruction	15
2	Object Reassembly	16
2.1	Introduction	16
2.1.1	An overview of Reassembly Cases	17
2.1.2	Methodological Goals	19
2.2	Related Work	21
2.2.1	State of the Art	21
2.2.2	Background: Feature Extraction	23
2.3	Method overview	23
2.4	Fragment Processing	25
2.4.1	Segmentation and Classification	25
2.4.2	External Feature Extraction	31

2.5	First Tier - Contact Surface Registration	34
2.5.1	Problem formulation	36
2.5.2	Optimization Strategy	36
2.5.3	Results Refinement	39
2.6	Second Tier - Feature Curve Object Registration	39
2.6.1	Problem formulation	40
2.6.2	Initial Alignment, Global and Local Search	41
2.6.3	Penetration-free Registration	42
2.7	Multi-part Assembly	43
2.8	Third Tier - Symmetry-based Fragment Registration	47
2.9	Evaluation Methodology, Results and Discussion	49
2.9.1	Cultural Heritage Datasets	49
2.9.2	Measurements and quantitative evaluation	52
3	Object Repair	57
3.1	Pre-processing	58
3.1.1	Removal of Fracture Surface	58
3.1.2	Poisson Reconstruction	60
3.2	Symmetry-based Completion	60
3.2.1	Related work	61
3.2.2	Method overview	63
3.2.3	Detection of Symmetry Planes in Partial 3D Meshes	63
3.2.3.1	Multi-scale Local Features	63
3.2.3.2	Vote-based Symmetry Detection	67

3.2.4	Reconstruction	70
3.2.5	Evaluation Methodology, Results and Discussion	71
3.3	Template-based Completion	72
3.3.1	Template Shapes	74
3.3.2	Local Features	76
3.3.2.1	Extensions	77
3.3.3	Global Feature Encoding Scheme	81
3.3.4	Quantization	81
3.4	Scaling of retrieved part & alignment to input	83
3.5	Merging & Annotation	84
3.6	Automated Inpainting	85
3.7	Missing Parts Computation	85
4	Conclusions	86

Abstract

In this document, we summarize the methodological results researched and achieved within PRESIOUS on the reassembly and object repair methods. We first provide a survey of typical imperfections in CH artifact scan data and in turn motivate the need for respective repair methods. We survey and classify a selection of existing reconstruction methods with respect to their applicability for CH objects, and then discuss how these approaches can be extended and combined to address various types of physical defects that are encountered in CH artifacts by proposing a flexible repair methodology in form of a workflow. This workflow accommodates an automatic reassembly step which can deal with fragmented input data (fragment reassembly). It also includes the similarity-based retrieval of appropriate complementary object data which is used to repair local and global object defects. Finally, we discuss options for evaluation of the effectiveness of such a CH repair workflow, taking into account results from the first evaluation report as well as recent results in benchmarking methodology.

This report is partially based on [\[20\]](#).

1 Introduction

We start with an introduction to the scope of the methodology, and define commonly encountered object defects. We then present a taxonomy of general methods for shape repair. Lastly, we give a comparison of similarity-based methods.

1.1 Scope of the Methodology

3D digitization of Cultural Heritage (CH) objects has recently gained significant ground, with applications ranging from preservation and presentation to analysis and re-usage e.g., in the cultural industry sector. A range of high- to medium-precision acquisition techniques exist, recently complemented by widely available commodity-type methods and hardware. However, the acquired 3D data are frequently imperfect or incomplete with respect to the represented original objects. This may be due to many factors, including acquisition artifacts and inaccessible views, but also due to a problem specific to the CH domain, the deterioration and fragmentation of the original objects. Due to deterioration effects, local detail may have been eroded away over time and worse, objects are often fragmented or missing important parts of the shape. The repair of all of these effects typically requires manual effort of CH domain experts and is time-consuming and thus expensive.

Recent advances in 3D shape processing allow for intelligent, automated processing of defects that stem from physical defects of the previously digitized objects. Successful automatic repair and/or reconstruction approaches have been discussed for specific applications, exploiting domain knowledge and constraints which control the process. Examples include pottery data, exploiting rotational symmetry [35], or city models applying search for block schemes and supervised learning [48]. However, such methods are specific to the addressed object class. More generic object repair may be possible if one assumes parametric templates of the object classes to be reconstructed [84]. Such approaches, however, inherently require definition of parametric repair templates and are increasingly computationally expensive with rising numbers of templates and template parameters.

Relying on recent advances in 3D Shape retrieval, we propose an approach that applies, extends and combines several techniques to address various defect categories while maintaining a high level of automation. Among others, we propose two new data-driven techniques to synthesize missing parts. The first approximates symmetry planes in defective objects to infer missing parts by intra-object similarity. The second relies on a repository of template shapes and a heuristic for automated surface classification to deduce missing object parts by inter-object similarity. Whereas the first technique does not imply any need for external data, the second one can by design handle objects that don't have prevalent symmetry planes. Furthermore its applicability scales with the ever-growing availability of digitized 3D CH models. However, the second technique can also benefit from parametric templates without

suffering in terms of scalability. The realization of our approach involves solving a range of 3D processing problems from retrieval of similar shapes based on partial shape information, registration and merging of shapes, repair of local mesh defects and transfer of shape detail.

Our contribution in this paper is to (a) characterize the problem of CH object repair based on a survey of typical shape defects occurring and a survey of related shape repair and reconstruction methods, (b) introduce and exemplarily apply a comprehensive similarity-based CH object repair workflow, addressing the identified object imperfections while presenting first results. Then, (c) we propose an evaluation methodology for the effectiveness of the repair workflow.

1.2 Defects of CH Artifacts

The notion of a defect or flaw indicates a certain frailty or shortcoming of an object that prevents it from having certain qualities that are precondition for utilizing the object for a certain purpose. In the context of CH object digitization, we distinguish between defects that are inherent to the physical object, and defects that are introduced during acquisition, conversion or processing of its digital representation. The latter (also denoted as mesh defects, in case the representation is of type mesh) can be considered an ubiquitous problem in computer graphics and has been subject to extended research [5]. However, there are few publications that specifically address identification or repair of the original physical defects of objects. Mesh defects such as singularities, holes or self-intersections can often be detected by rather simple and well-known algorithms. On the other hand, identifying physical defects of CH objects can in general be regarded as an ill-posed problem and is hard to do fully automatically. Typically, to repair such defects, background knowledge as e.g. the class of the object, its material, or environmental conditions that it was exposed to, needs to be brought in.

We can classify important types of defects that we encounter in CH objects:

Small-scale decay results in a high number of small and irregular changes to the objects geometry and texture. Examples of such processes are chemical weathering and mechanical abrasion. As these defects are normally densely collocated, they often affect larger, connected areas of the object surface. Although the spectrum of small-scale features of an object can be severely affected, the effect on the overall shape of an object is usually small.

Missing of small fragments encompasses chipped off slivers, fissures and larger scratches which reduce the volume of the object. These effects can be caused by physical weathering or other mechanical stress. In comparison to the previous defect category, these defects are usually less in number but cause stronger local effects on the surface geometry. Thus they can be more easily distinguished and the geometrical information for their repair is more likely to be derived from its surroundings. However, their exact geometric appearance is still dependent on the objects material and environmental



Figure 1: Defective objects from Nidaros Cathedral, Trondheim, Norway: Fragments of a doorway arch (top left), reassembled embrasure with missing parts and cracks (top right), reassembled column base with large cracks (bottom left), reassembled tombstone with hole and cracks (bottom right).

conditions.

Sediments on the surface increase the volume of the object and can have a varying intensity which especially affects the texture as well as small-scale geometric features of the surface. The affected area may be large, depending on the root cause. Some parts of sediments may be manually removed in preparation of 3D acquisition.

Missing of Large Parts leads to significant changes of the overall shape and can arise due to mechanical stress or as a long-term effect of physical and chemical weathering. Missing large parts also correspond to breaking edges that introduce small-scale local features which are not related to the original shape. If broken-off large parts were recovered and digitized, they can be used for reassembly.

For parts that can not be recovered, all explicit information on the geometry and texture is lost. Due to their varying size, it is in many cases difficult to infer their geometry directly from the remaining recovered parts. Under such circumstances, missing larger parts could look alike to missing small scale fragments or residues. For example, a mug missing its handle could resemble a vase. A fragment of an embrasure could be misclassified as a decoration of a pillar. Often, even CH domain experts have difficulties in predicting the class or completion of a CH artifact which is missing larger parts. Then, domain specific knowledge and assumptions may be used to create hypotheses on the original appearance and meaning of the object.

To summarize, when dealing with CH artifacts in practice a range of imperfections will be encountered. Figure 1 illustrates some of the defects occurring in real CH object data. To obtain a digitization that is closer to the original geometry of the object, these need to be dealt with.

1.3 Taxonomy of 3D Shape Repair Methods

In this section, we present a brief survey of 3D shape repair methods that can be applied to digitization of defective CH objects. Based on an extensive literature study, we can distinguish a) general methods which can address mesh structure defects [5], and b) methods which rely on shape similarity to repair objects. While general mesh repair methods are useful for automatically removing smaller defects or global mesh inconsistencies, similarity-based methods are needed to repair or replace larger shape parts. We sub-divide the similarity-based approaches into those that address Modeling (or Assembling), Reassembly, Inpainting, and Symmetry-based methods. While the former two are mainly based on exploiting similarity to externally provided shapes, the latter two focus on exploiting similarity within the shape itself. Fig. 2 shows our taxonomy. We next survey some recent work in each of these categories, before giving a comparative summarization.

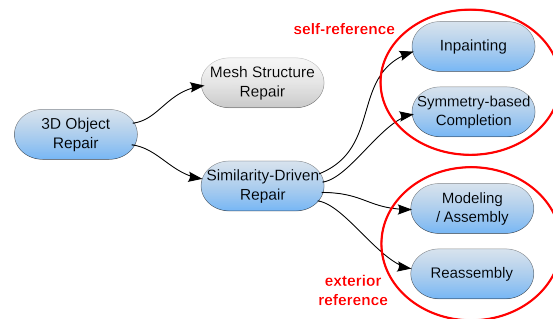


Figure 2: A basic taxonomy of 3D Object repair methods. We distinguish similarity-driven repair methods that exploit intra or inter-shape similarity to synthesize a plausible overall shape.

1.3.1 Modeling / Assembly

In approaches such as the classic “Modeling by Example” [19] and its extensions, the user interactively models the reconstruction by reusing parts retrieved from an external shape repository. Repository queries are performed by user-provided metadata keywords or by content-based similarity to the object within user-denoted query regions. Subsequently, the user selects a shape from the retrieved candidate list for geometry transfer. In [45] the original query approach is extended to use 2D sketches that can be drawn on top of the object. Furthermore, the shape repository is extended to contain the segmentation of objects as well as semantic annotations of objects and their parts. These are used to increase retrieval accuracy while reducing online user input. E.g. based on annotations of already assembled parts, the system can propose relevant parts even before the user initiates the next query. Both approaches can be regarded as highly versatile, yet they require external example data from which to obtain information for the repair process. To address this, in [34] an approach is described that can be used to automate the creation of such repository data by machine-learning-based segmentation and part-labeling. Although further extensions as [15, 33] aim to automate the generation of repository data even more, modeling inherently requires a high number of online user interventions.

1.3.2 Reassembly

While also primarily relying on inter-object similarity, reassembly approaches differ from modeling in that they operate on a set of fragment objects that are expected to be parts of the reconstruction. Reassembly can be driven by two entirely different processes: Contact surface (multi-part) registration and part-to-whole matching, which, in many cases, can also be combined.

In multi-part registration, the fragments are first automatically segmented to detect potentially fractured surfaces. Subsequently, all combinations of marked segments are subjected to a rigid registration procedure, where both the estimated relative pose and the matching error (residual) are computed. This (potentially offline) processing results in a large number of possible fragment pair configurations, which are finally resolved by a combinatorial optimization stage to form clusters of aligned fragments. Two prominent automatic approaches are reported in this area; Papaioannou et al. [68] perform a rigid registration by optimizing the local residual gradient and contact area between the fracture surfaces. Residuals are measured via the projection of the fractured segment pair on a common plane using hardware rasterization and stochastic pose optimization is used for the fast alignment of the surfaces. Results are further refined by performing contour matching on fracture boundary segments [67]. Multi-part reassembly is achieved by a genetic algorithm that globally optimizes the pairwise links. Huang et al. [27] extract multi-scale features on the fracture surfaces and perform a forward search for an optimal alignment of local descriptor tuples. Combinatorial multi-part reassembly is handled via graph optimization, by incrementally clustering and merging best-matching fragments and testing for interpenetration. Both methods are generic, i.e. not restricted to cultural heritage objects or particular types of shapes and require no online user input.

In the part-to-whole matching, fragments are compared against an objective goal shape and their pose is primarily guided by their best-fit registration on the latter. A prominent example is the method by Wei et al. [86], where the reassembly is specifically geared towards the reassembly of human skulls models. In contrast to the multi-part registrations above, the approach also operates on fragments that do not share breaking edges. Local shape descriptors are extracted from the fragments and a pose is estimated by feature registration and distance minimization. Subsequently, the alignment of the fragments is globally optimized by shifting them to eliminate fragment intersection. Finally, missing patches between the fragments are synthesized using non-rigidly deformed patches of the skull template. This approach is reported to work without any need for online user input.

1.3.3 Inpainting

Very similar to 2D image inpainting, 3D Shape Inpainting synthesizes local object details in smaller defective areas based on the features found in the surrounding or remote but similar intact areas. Kawai et al. [37, 38] proposed a method that does

not require any user input after candidate target areas to reuse are identified from the shape, and thereby improves over previous approaches such as [7, 8]. In the method, first the boundary of a whole where an object needs to be repaired, is detected. All vertices on the boundary are connected to a new vertex that is placed in their center of gravity. For determining features that are to be matched and reproduced by the hole-filling, the geodesic vicinity around the hole is chosen as a target area. In the next step, points are iteratively added and deleted from the solution to match the point density and distribution of the target region. In a final step, a cost-function that compares the principal curvature of vertices on target and source area is minimized by shifting the inserted vertices.

Similarly, Harary et al. [22] proposed to use self-similarity to inpaint holes in surfaces. When a hole is identified in the surface, a rough initial triangulation is produced. Subsequently, the algorithm computes Heat Kernel Signatures [80] for every vertex in the mesh and a new descriptor for small regions is proposed by averaging the HKS's of vertex in the region. Then, for each vertex in the initial region, the nearest neighbor patch in the HKS space is used as hypothesis to inpaint. Finally, a blending step is performed to combine all the patches for the points in the initialized region. In contrast to the Inpainting method from Kawai et al., this approach is not limited to source areas that are located in direct vicinity of the target hole.

1.3.4 Symmetry-based completion

Many man-made objects are roughly mirror, point or radially symmetric (*global symmetry*). Even more man made objects can be decomposed into segments that are in some way symmetrical on their own (*partial symmetry*). Almost all man made objects can be regarded as partially symmetric if instead of euclidean (*extrinsic*) distances, geodesic (*intrinsic*) distances of features on the segment's surface are considered. There are numerous publications [60] on the detection of global, partial as well as extrinsic and intrinsic symmetry and their application to areas such as shape segmentation, recognition, compression, missing parts detection and reconstruction. However, rather few methods are actually robust to cope with larger missing parts and the often imperfect (i.e. inexact) symmetries found on digitized CH objects.

Related to the skull reassembly method of [86] (see also Section 1.3.2), in [47] missing skull fragments are synthesized according to their counterparts across the global, extrinsic symmetry plane of the skull. First, a uniform sampling of the previously aligned fragments is obtained. In a second step, the sampled vertices are paired according to their local principal curvature and shape diameter function. Reportedly, this combination is robust against missing large parts. Then the transformation between normals and direction of minimum and maximum curvature of matched pairs are computed. Each of these transformations corresponds to a pair-specific ideal symmetry plane. To determine the global plane, these transformations are clustered using a mean-shift algorithm. As the reassembly proposed in [86], this approach does not require any online user input. The approach assumes the existence of exactly one global extrinsic

symmetry plane and has thus a limited versatility. Besides the hard-coded assumption about symmetry characteristics, there is actually no need for external data. In Jiang et al. [31] an approach is published that uses intrinsic symmetry for completion of defective models. Symmetry detection is based on an a priori skeleton extraction. Once the skeleton is computed, the algorithm finds symmetric correspondences between the bones. Regions around the skeleton can be completed by copying them from the corresponding symmetric bones. While this method is suitable for repairing medium sized defects, it would not be suitable if large parts are missing since the skeleton would change dramatically.

1.4 Comparison of Similarity-Based Repair Methods

repair type / method	large parts	small parts	input actions	input complexity	external data	versatility limits
Modeling Funkhouser et al. [19]	+	-	several per defect	3d query boxes, ...)	fully detailed models	amount of external data & input
Modeling Lee et al. [45]	+	-	several per defect	2d sketches, ...	fully detailed models	amount of external data & input
Reassembly Papaioannou et al. [67]	+	-	none	n/a	none	shared breaking edges
Reassembly Wei et al. [86]	+	+	none	n/a	none ¹	skull fragments only
Inpainting Kawai et al. [38]	-	+	one per defect	defect selection	none	source area adjacent to target, fading quality for larger targets
Inpainting Harary et al. [22]	-	+	one per defect	defect selection	none	fading quality for larger targets
Symmetry ² Li et al. [47]	+	+	none	n/a	none	dominant global symmetry plane
Symmetry Jiang et al. [31]	o	+	none	n/a	none	not robust to larger missing parts

Table 1: Comparison of related repair approaches according to their applicability for reconstruction of CH Objects

As the discussed similarity-oriented repair methods have been developed for different use cases and applications, it is hard to directly compare them. Yet, we can compare them according to qualitative criteria regarding their expected applicability for CH object repair. Specifically, we consider the following comparison criteria along which we can discuss the identified methods:

large parts repair estimates the effectiveness of a method for reconstruction of missing large parts.

small parts repair estimates the effectiveness of a method for reconstruction of areas that are affected by small scale defects such as e.g. chipped or fragments, fissures, scratches or small-scale decay.

input actions classifies the number of user input actions that are required during the repair of an object. e.g. one or more input actions are required per input object, fragment or defect

input complexity classifies the methods according to their most complex input. Low complexity resembles e.g. the choice of a solution out of a small list of candidates, whereas manual alignment or CAD-like geometry input would be classified as high complexity.

external data summarizes a methods overall need for external data, such as e.g. Shape Repositories or other object specific parameters. External data, when provided offline, can often reduce the amount or complexity of user inputs during the repair process, however this external data usually cannot be computed automatically.

versatility limits enumerates the most limiting aspects of a method according to their application to generic defective CH objects.

Table 1 summarizes the discussed methods and comparative criteria. Note that these are our own assessments based on a literature analysis but more substantiation and also, empirical comparison of the methods to each other is a work yet to be done.

1.5 Proposed Workflow for Automated Reconstruction

TODO: revise text to fit workflow... refer to SIGGRAPH extended abstract

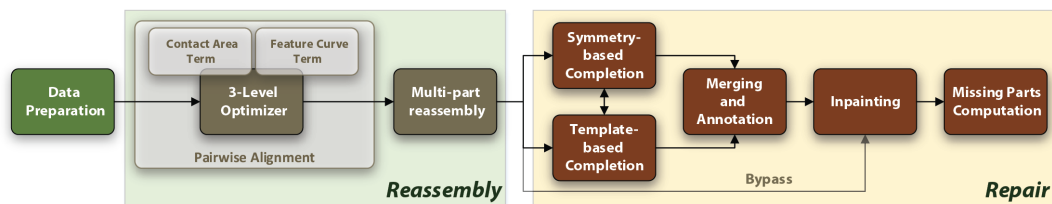


Figure 3: Overview of our workflow for automated repair of defective CH objects

When examining the comparison of the repair methods above, one can easily observe, that none of the methods is simultaneously highly versatile and also well suited for repairing both small-scale defects and missing large parts. This leads to the conclusion that, to obtain a more versatile approach for automated reconstruction, several

methods have to be combined. However, special care has to be taken that their sequence does not degrade the quality of results of the individual methods and that expensive intermediate results are reused where possible. Figure 34 summarizes our proposed combination and sequence of mesh processing and repair methods. While every step could be executed without the need for any online user input, several steps require external data that has to be provided a priori. For the workflow to produce results, not all completion steps have to contribute repairs in every case. In fact, for many objects, at least one of Reassembly, Symmetry- or Template-based Completion cannot always be applied successfully. If necessary, robustness and solution quality could optionally be improved by relying on additional user input. E.g. every step could potentially be executed multiple times with slightly different parameter sets to produce a list of solution candidates from which the user could choose interactively. Alternatively, the user could modify parameters such as thresholds directly on the fly³, while the corresponding processing results were visualized for feedback. Our contribution is the conceptual combination and sequence of the methods within the overall workflow. In principle, for each processing step it is in many places possible to identify alternate methods that implement similar functionality. However, to show the feasibility of the workflow, we sketch approaches for each step and also show first, preliminary results in the following sections.

2 Object Reassembly

2.1 Introduction

The physical reconstruction of cultural heritage finds from fragments found at excavation sites is a time consuming and difficult task, especially for large objects or large collections of fragments, given also the fact that the pieces may be remotely located in different physical collections. The digital counterpart, *virtual object re-assembly*, has received significant research interest in the past years, mainly with regard to specialized object types, such as frescos and pottery. Moving to the virtual, computer-assisted, domain, provides numerous advantages, including the access to remotely located physical finds and the ability to easily manipulate 3D shapes, whose physical counterparts may be hard to handle. Above all, it benefits from the exploitation of robust and fast (semi-) automatic algorithms for computing and exhaustively testing hypotheses at a large problem scale, even extreme ones. Interestingly, algorithms and methods developed in the scope of computational archaeology can be also applied to the domains of forensics and computer-assisted surgery, which have also expressed interest in such methods and perform related research.

In computational archaeology, typically the problem is described as the automatic process that involves the identification of potentially fractured parts/regions of an object, the search for corresponding pieces within a fragment collection and finally the

³e.g. by dragging simple sliders

clustering and pose estimation of multiple parts that result in a virtual representation of (partially) reassembled objects. In the general case, the problem has $2 \dots N$ input part representations (surfaces, volumes, point-clouds etc.), each expressed in its own local coordinate system. The resulting solution consists of $1 \dots M$ clusters. If no cyclic associations are allowed, this translates to at most $N - M$ rigid transformations that describe the pose of the fragments in the output clusters.

Typically, the process starts with the digitization (e.g. 3D scanning) of the physical fragments and continues with the pre-processing of the fragment geometry in order to extract the fractured and intact surfaces (segmentation and classification). Subsequently, all pair-wise combinations of the fragments are tested for alignment and a matching error is computed. This step usually begins with a global registration process that examines the solution search space for a good but rough alignment, that in turn initiates a local registration process in order to refine the solution. The complete set of the pair-wise results drives subsequently the multi-part alignment (reassembly), where complete objects are formed by finding the global position for each fragment.

2.1.1 An overview of Reassembly Cases

It is often informative to categorize the input domain of a specific problem, here the shape of the parts involved, as well as the target outcome, which in our case is the shape and attributes of the reassembled objects. This taxonomy helps specialize the algorithms to involve, take advantage of specific conditions or exploit rich content, where available. In Figure 4, we present a high-level distinction of the classes of target shapes according to 3 broad criteria: thickness, symmetry and presence of features on the intact surfaces. As will be explained below, in the state of the art section, it is very effective to rely on specialized algorithms that operate in a reduced parameter domain, if prior knowledge about the shapes is given. For example, relatively thin surfaces can be treated as 2D embeddings in 3D, substituting the notion of a break surface with that of a break line. Symmetrical objects, such as vessels can effectively exploit the axial symmetry to reduce the optimization parameters for the pose estimation of the fragments (pottery sherds) and for flat objects, such as frescos, we can operate on the plane instead of the three-dimensional space.

In practical scenarios we seldom encounter mixed cases of object categories, since for all practical reasons, the CH expert has already performed a pre-classification of objects in appropriate collections. Therefore, it is typical to have large collections of pottery sherds, potentially pre-classified by material and location. For larger fragments, such as broken architectural elements, ornaments, statues etc., although their number is typically an order of magnitude smaller than those of sherds, the reassembly problem, from the solving strategy point of view, is far more complex and is highly affected by the following factors:

- **Deterioration.** Sherds are typically found in clusters of broken pottery and are either fairly well-preserved in terms of shape and thus usable, or in a very poor

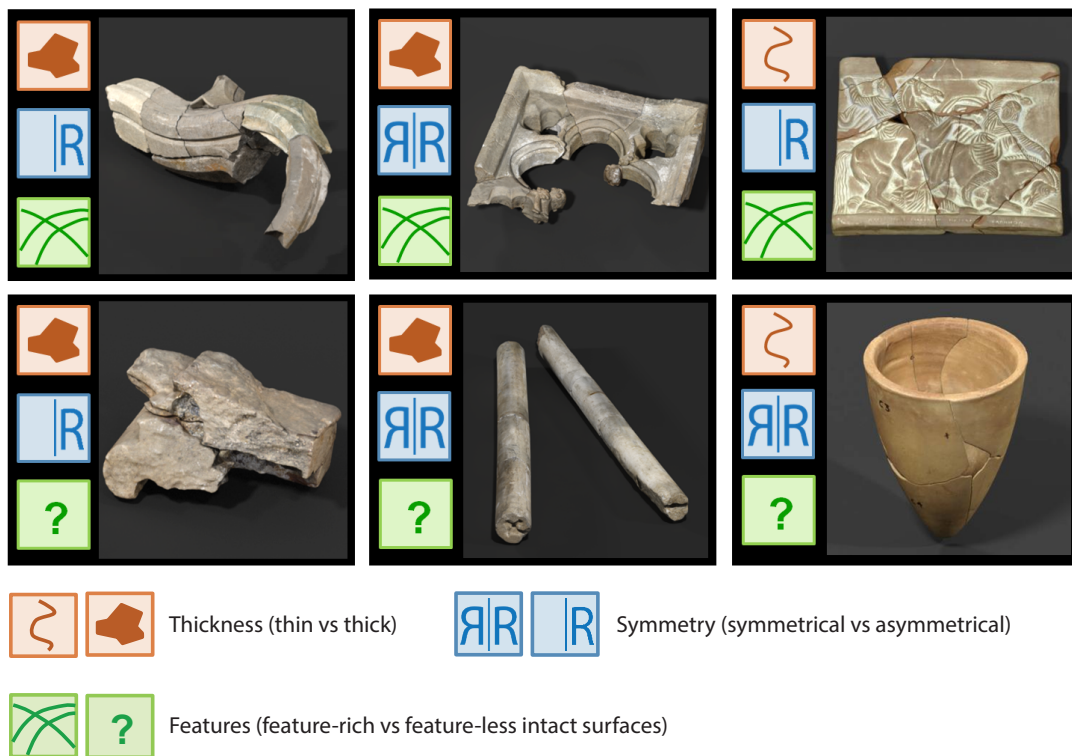


Figure 4: A high-level taxonomy of the target shapes for reassembly.

condition that makes them impossible to use without making strong assumptions and subjective interpretations. On the other hand, bulkier fragments are typically exposed to weathering, erosion and physical stress, partially damaging their surfaces but some times retaining certain usable features on their surfaces. Furthermore, missing parts in the case of generic objects tend to result in room for more interpretations, greater ambiguity and sometimes make impossible to determine the exact shape and number of the original object, given a set of fragments.

- **Fragment accessibility.** Due to illicit trade of antiquities, exposure of monuments in different museums, jurisdiction or simply due to monument elements re-purposing or different excavation periods, very often parts belonging to the same object are physically separated and therefore impossible to access at the same time. This situation is far more common in the case of larger finds than in collections of sherds or frescos, which are typically discovered, collected and archived in a localized manner.
- **Physical constraints.** Actual fragments are sometimes impractical to handle for measurements and compatibility, due to their size, and this is one reason why digitally attempting their assembly has a significant impact. In fact, even the digitization of large objects in a complete manner is nearly impossible, demanding that the algorithms involved in the assembly are robust and tolerant

to missing information and outliers (e.g. erroneously including "background" geometry like the ground a heavy object rests on).

- **Geometric priors.** Typically, architectural parts and other bulky elements do not possess the canonical and well-structured form of common types of vessels, making it harder to determine global constraints about the expected shape of the resulting forms. However, when this information can be deduced, from symmetries or similar objects found, we explore here potential ideas on how to exploit it to strengthen our search strategy.

In general, the combinatorial part of the multi-part assembly shares many common ideas across all types of objects and *puzzling* problems. Still, the generic three-dimensional case prohibits any convenient simplifications that allows the mitigation of the pairwise testing as part of the multi-part approach, since the cost of operations is at least two orders of magnitude higher. Furthermore, a two-dimensional scenario, e.g. flat, thin sherds, can be converted to a least squares fragment orientation problem, given the low dimensionality of the parameters involved. The same does not hold for the three-dimensional case.

2.1.2 Methodological Goals

For the assembly stage of the object repair pipeline, we tried to capture the most generic approach to solving the pure three-dimensional puzzling problem, involving irregularly shaped fragments with or without generic geometric priors, not specific to one object type. So, in the methodology followed, we make no assumptions about specific classes of objects, we do not consider a reduced pose parameter space, nor do we implement any case-specific geometric constrain (e.g. axial symmetry etc.).

From the start of the project we soon realized that an important and *unexplored* aspect of the computer assisted assembly methodology is the ability to exploit information beyond the contact surface of the fragmented parts. In most cultural heritage scenarios, many fragments are either heavily eroded or pieces of the puzzle are missing, the latter attributed to both the physical destruction of smaller fragments or the absence of certain interconnecting pieces. Therefore, we developed a three-tier geometric registration and puzzle solving strategy that would enable us to provide plausible solutions for objects with high erosion or even large missing parts, which almost all existing approaches failed to address. More importantly, for very hard cases, we facilitated the introduction of decision points to leave room for interpretation, led by the CH experts.

In terms of technical goals, we singled out the following drivers, based on the broad methodological scope described above:

- **Pure geometric solution.** No dependence on subjective data (annotations, classification etc). No dependence on texture, as it proved to be misleading in

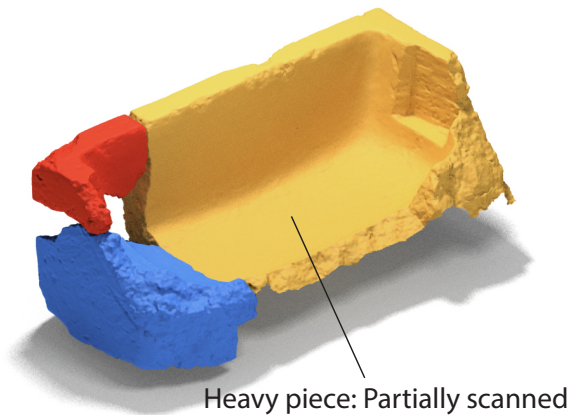


Figure 5: Example assembly using both complete and partial scans.

the test cases we encountered. However, other specialized cases (e.g. frescos) could benefit from textural landmarks, though their exploitation is orthogonal to our approach and could be seamlessly integrated (see integration of external feature curves with the contact-surface metrics).

- **Input Generality.** Algorithms should be applicable to both point clouds and meshes and be able to handle both fully and partially scanned fragments. The latter is important since in many scenarios, it may be technically challenging to scan an artefact from all sides (see Figure 5). It can also enable the development of on-line, *process-as-you-scan* systems.
- **Resilience to outliers and errors.** Due to large gaps between fragments, erosion and the potential presence of holes and errors in the input data, the geometric registration approach involved must be robust to outliers. The same holds for all stages of the geometric processing pipeline, including the segmentation and feature extraction. Especially for segmentation and surface region classification, since erosion renders the boundaries of break surfaces fuzzy and unreliable, the registration algorithm that follows this step, must not closely depend on the exact output of a segmentation stage.
- **Shape-guided assembly.** Going beyond the previous work in the field, we pursue in this project the discovery and exploitation of global structure in the input parts. Unlike efforts that use a particular shape class as a geometric prior (e.g. pottery with cylindrical symmetry), we assume no such prior knowledge exists and try to infer the structural constraints from the shape of the intact surfaces.

2.2 Related Work

2.2.1 State of the Art

The most general and hardest case of object assembly is that of 3D fragments and it has been addressed with both automatic and semi-automatic approaches. Automatic methods generally assume that matching fragments share a significant contact surface and after testing and measuring rigid pairwise part combinations for alignment, solve a generalized geometric puzzling problem (multi-part alignment).

Papaioannou et al. [68] were the first to address the 3D object reassembly problem, under the assumption that fractured faces are nearly planar with sufficient overlap. Fragments are initially segmented using region growing, classified using a normal deviation metric and compared using hardware-accelerated distance-to-plane queries. Huang et al. [28] proposed another 3D object reassembly method that utilizes multi-scale features of the fractured surfaces in order to perform the matching operations. Segmentation is performed through a contour extraction method and classification is performed using a normal variance measure. The proposed system performs remarkably on fractured objects, where fractured areas contain rich intrinsic geometric features, but depending on the material and/or erosion of the surface this information can be easily lost. Li et al. [46] proposed another multi-scale feature-based pairwise reassembly approach, while Altantsetseg et al. [2] evolve the same idea to matching of feature curves on the fractured facets, approximated by Fourier series. Similarly to the method by Huang et al. [28], both methods require fractured surfaces with rich intrinsic geometric features. Winkelbach et al. [88] introduced a different pairwise alignment approach that tries to maximize the contact area using a branch-and-bound search heuristic. While the method does not rely on feature extraction, the contact maximization criterion of the method is not robust to noise and outliers and subsequently cannot handle chipped and eroded surfaces. Finally Mavridis et al. [54] proposed a minimization scheme that is also based on the contact area of the fragments, and show that the use of an ℓ_p -norm with low values of p can partially address noise and outliers. Still, similar to previous approaches, the contact surface formulation of the problem does not lead to plausible solutions in the case of significant erosion or missing parts.

With semi-automatic methods, the user can drive the reassembly by either specifying constraints or enforcing particular matches. Parikh et al. [69] use an approach, where the user selects possible pairs from a set of compatible parts. Compatibility is computed based on local features of the fractured surfaces. Mellado et al. [57] propose an interactive loop, where the user specifies approximate initial positions for pairwise matches and use an ICP variant to refine the user input. Palmas et al. [66], in order to address eroded fragments or fractured surfaces without features, exploit user-specified constraints on fragment pairs in an energy minimization scheme, which still relies on the distance of the fractured surfaces. Both [57] and [66] in essence, exploit user interaction to avoid the most difficult step of the reassembly, the global registration. The proposed method, in contrast to other semi-automatic approaches,

tries to solve both global and local registration steps, by exploiting salient feature curves on the intact surfaces of the objects.

Other methods, try to solve specialized reassembly cases, by exploiting either rotational symmetry or reduction of dimensions. Willis et al. [87], Kampel and Sablatnig [36] and Son [79] solve the specialized rotationally symmetrical cases of vessels by using estimates of symmetry axis and break curves. Brown et al. [13] address another specialized case, that of thin-walled flat fragments (frescos), by uniformly sampling the ribbon of each fragment and using a 3D contour matching approach. Similarly, Belenguer et al. [76] proposed another solution for fresco reassembly, using a shape descriptor on the discretized fracture ribbon and a hierarchical approach.

Lines and other features of the intact surfaces of the objects, have been used in order to help the reassembly process mostly in the 2D case. Koller and Levoy [43] try to address heavily eroded fresco fragments that sometimes do not share a matching surface, using manually placed markings on digital photos of the fragments, while Toler-Franklin et al. [83] identify and design descriptors based on the orientation of regular surface patterns, such as brush strokes. The first approach to incorporate features of the intact surfaces of the fragments in the 3D case is the work of Thuswaldner et al. [82]. Their approach is limited on rectangular shapes and uses context-specific information like surface roughness and bolt holes. Special cases of planar surfaces and straight lines are used as features, which provide additional constraints to the feature-based registration approach of Huang et al. [28] to align unbroken architectural monument parts.

Huang et al. [24] were the first to exploit data extrapolation in the three-dimensional domain, in order to achieve pairwise surface registration of non-overlapping parts, with an application in shape composition. Two parts are registered, by first computing a feature-conforming volumetric vector field extrapolated from the first one, and subsequently aligning the second with respect to the extrapolated field. The work of Huang et al. is generic enough, as it does not rely on salient features. On the other hand, this is the shortcoming of the approach in the case of highly fractured and eroded parts, where the boundaries are not well defined and the field extrapolation could be severely distorted, rendering the method incapable of properly aligning the pieces. In our approach, the overall shape of the unbroken regions strongly affect the shape of the salient feature curves generated, making them resilient to outliers.

In addition to the the 3D and 2D cultural heritage cases, data extrapolation has also been recently exploited in the computer vision field for non-overlapping image alignment. Two of the most characteristic examples are the works of Poleg and Peleg [72] and Huang et al. [25]. Poleg and Peleg extrapolate images and subsequently search for alignment of other pieces with the extrapolated part. Huang et al. initially extract salient feature curves using an edge-preserving filter and gradient magnitude maps and subsequently filter them using several heuristics. The feature curves of the individual pieces are paired and an ambient vector field is constructed. This ambient field is used for the alignment of the pieces.

2.2.2 Background: Feature Extraction

Since a large part of our methodology relies on features that can be detected on the intact surfaces of the fragments, we briefly present background work in this area. Feature extraction on surfaces and point clouds is a research area that has been studied thoroughly in the recent past. It is a computationally intensive task and usually relies on the estimation of the surface curvature. All methods in this area are semi-automatic as parameters have to be adjusted according to the target feature scale and the underlying noise.

Ohtake et al. [63] use a global fitting method in order to compute the curvature and detect line features. However, while global fitting methods are expected to achieve more accurate detection of features, local estimation schemes are much faster and often demonstrate good results as shown by Yoshizawa et al. [93]. Local estimation is achieved either by local polynomial fitting (Yoshizawa et al. [91] and Kim and Kim [42]) or by discrete differential operators (Hildebrandt et al. [23] and [92]). While fitting methods can frequently dispense with noise elimination, they incorporate smoothing in the curvature estimation process that is very difficult to control and are relatively slow in comparison to the discrete differential operators. Pauly et al. [70] proposed another local estimation method for the extraction of linear features. The method measures the local surface variation through a statistical operator on local neighborhoods, which utilizes multiple scales to increase the robustness in the presence of noise. Finally, Daniels II et al. [17] present a local surface fitting method that uses a Robust Moving Least Squares algorithm in order to construct feature curves from a set of extracted feature points. The authors showcase very good coverage of the object's feature curves, but their approach requires up to 10 minutes for objects of 500k points. This renders the method impractical for scenarios where iterations are required (ex. parameter fine-tuning).

2.3 Method overview

Related Publications
A. Andreadis, G. Papaioannou, and P. Mavridis. Generalized digital reassembly using geometric registration . In Proc. IEEE/EG Digital Heritage, 2015.
P. Mavridis, Andreadis, A., and G. Papaioannou. Fractured object reassembly via robust surface registration . Eurographics Conference (short paper). 2015.
A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., Fractured 3D Object Restoration and Completion . ACM SIGGRAPH Poster Session, 2015.

The assembly pipeline implemented in this project is a three-tier approach, where gradually more relaxed constraints are used in order to provide an increased set of

solutions to the puzzling problem. This of course implies that the more unreliable the constraints become, the more the results are up to interpretation and the more important the validation from a CH expert is. Still, such a virtual assembly is an important tool, as it can generate clusters of solutions with adequate precision to be reliable for inspection and final validation, while suggesting fragment configurations that might have been otherwise difficult to test and explore, even in a virtual (manual) restoration environment.

As with most reassembly approaches, our data require an initial processing in order to extract the potentially fractured and intact facets of the fragments. Subsequently, the first tier of geometric registration is applied in order to form pairwise matches with good contact area. The resulting pairs can be evaluated by the expert user who is responsible to either accept or discard them. At this point, the user can initiate the second tier of geometric registration for fragment pairs with similar characteristics, but with significant erosion or missing information in their matching fractured facets. Using an intuitive and trivial approach, the user can quickly extract salient feature curves on the intact surfaces of the fragments. These feature curves are next used in an automatic alignment process of the pairs. Having generated the set of matching fragment pairs, the multi-part reassembly procedure is then initiated. Here, the set of matching pairs is explored and objects consisting of multiple fragments are generated.

It is possible that during this process some fragments remain isolated, either because they form bonds with high error, due to incompatible contact surfaces or external features, or because they are not directly coupled with the rest of the fragments (disjoint). In order to address this issue, when appropriate, we utilize partial object symmetries in order to extract the complementary geometry of a cluster already formed by the first two tiers. While the symmetrically expanded shape by itself is of little scientific value to the archaeologists, in the third tier of our geometric registration approach, we use it as a guide in order to generate possible registration poses of the disjointed fragments, which can be afterwards evaluated by the experts.

The outline of the above three-tier reassembly pipeline is presented in Figure 6. In the rest of the section we present the processing of our data and each of the proposed stages of geometric registration.

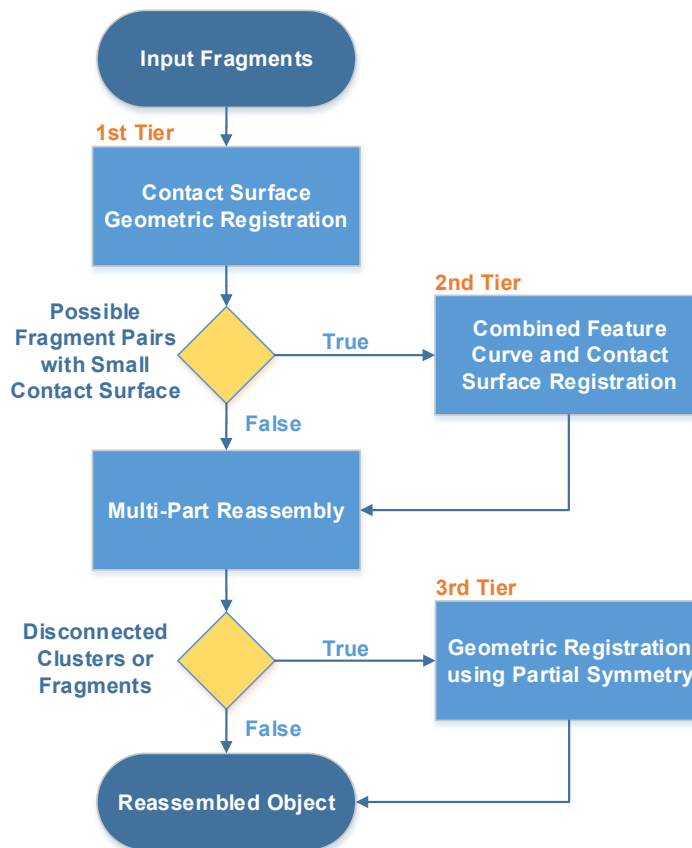


Figure 6: Flow chart of the proposed three-tier geometric reassembly pipeline. Decision points reflect intervention by an expert user.

2.4 Fragment Processing

2.4.1 Segmentation and Classification

As mentioned earlier, in this step the main goal is to extract distinct contiguous surface regions (*facets*) from the scanned fragments and classify them as fractured or intact (see Figure 7). This pre-processing of the input data is performed primarily for robustness, since pairwise matching operations between flat intact surfaces will always yield trivial, yet undesirable matching results. It also increases efficiency, as the exclusion of the intact facets from the geometric comparisons reduces both the search space and the distance measurement operations performed.

Segmentation. Two aspects of a segmentation algorithm have a direct impact on the quality of the results: the ordering of operations and the distance metrics used in order to test the compatibility of two regions.

For the ordering of clustering operations, we have experimented with both *region growing* (RG) and *hierarchical agglomerative clustering* (HAC). RG proceeds by developing a single region based on a seed element. The region grows by iteratively

Related Publications

A. Andreadis, Papaioannou, G., and P. Mavridis. [A Parametric Space Approach to the Computation of Multi-Scale Geometric Features](#), International Conference on Computer Graphics Theory and Applications. 2015.

A. Andreadis, G. Papaioannou, and P. Mavridis. [Generalized digital reassembly using geometric registration](#). In Proc. IEEE/EG Digital Heritage, 2015.

P. Mavridis, Andreadis, A., and G. Papaioannou. [Fractured object reassembly via robust surface registration](#). Eurographics Conference (short paper). 2015.

A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., [Fractured 3D Object Restoration and Completion](#). ACM SIGGRAPH Poster Session, 2015.

A. Andreadis, Mavridis, P., and G. Papaioannou. [Facet Extraction and Classification for the Reassembly of Fractured 3D Objects](#). Eurographics poster session. Strasbourg, 2014.

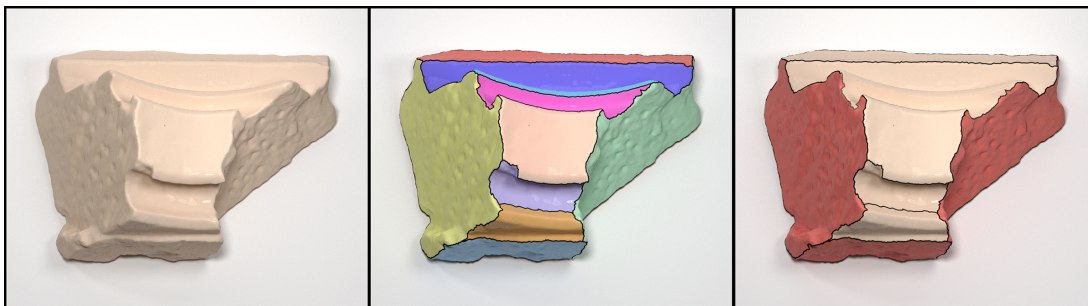


Figure 7: From left to right, input fragment, segmentation result with coloured segments for clarity and the resulting classification of segments, where red denotes the fractured ones.

merging neighbouring elements that meet some user-defined criteria. When no other neighbours can be found, the process starts with a new seed and the algorithm terminates when all input elements belong to a region. We have experimented with two variations of the algorithm, one that selects a random neighbour (RGN) and a second that selects the one with the closest metric distance (RBF).

On the other hand, HAC starts by considering each element as a region (cluster). Regions are then incrementally clustered, by merging the two regions with the closest distance, according to a user-specified metric. The merging process stops when no neighbour clusters meet the merging criteria. A custom 2-level caching scheme is used by our implementation in order to avoid redundant distance and sorting calculations.

The ordering of operations does not seem to affect the resulting segmentation in this

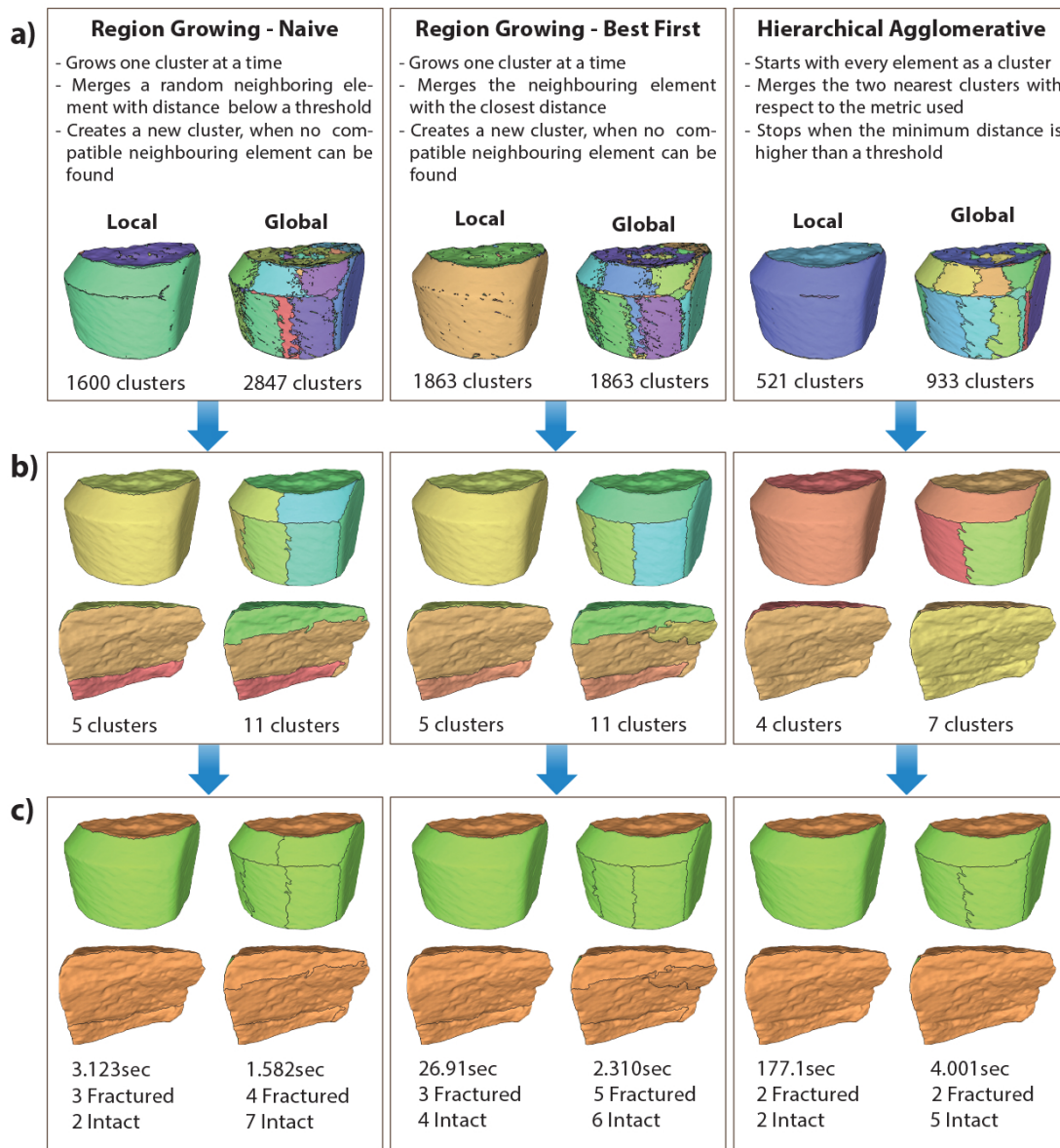


Figure 8: A step-by-step visualization of the segmentation and classification results for an example fragment. a) is the resulting segmentation using all the combinations of the described ordering of operations and distance metrics. Results after the Post-Processing step are shown in b) and finally, in c) we see the classified results. Green and orange colours are for the intact and fractured facets in respect.

case, favoring thus the naive approach due to the performance impact inflicted by a more sophisticated approach (see Figure 8a).

Segment Post-Processing. The greedy nature of the region growing algorithm can lead to severe over-segmentation. This is fixed by a custom post-processing algorithm that first decomposes small regions into single elements, which are subsequently merged to larger neighbouring segments based on their average normal and normal variance in order to achieve good discrimination between adjacent distinct facets (see

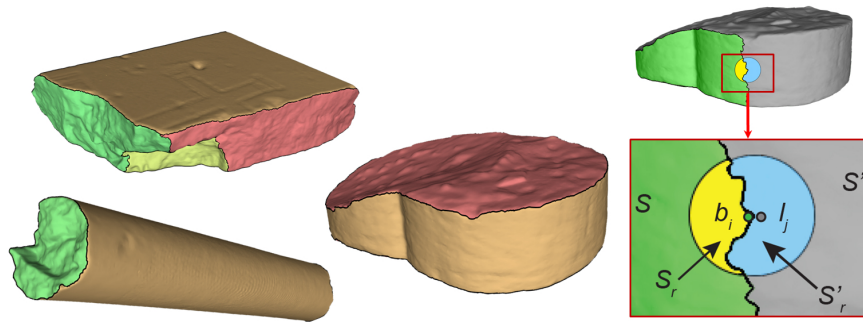


Figure 9: Effective segmentation of both planar and curved surfaces using a locally averaged dihedral angle metric, in the neighbourhood of the element under examination and the corresponding cluster.

Figure 8b). The same post-processing also improves the results of the hierarchical clustering method.

One of the simplest distance metrics for mesh segmentation is the angle between the average normal of two segments (dihedral angle). While this metric performs well on planar surfaces, it results in over-segmentation on curved ones, as shown in Figure 8. A simple way to properly handle both planar and curved surfaces is to use the average normal computed on the local neighbourhood on the boundary of the two segments. The size of this local neighbourhood is a parameter that determines the tendency of the algorithm to break curved regions to multiple segments. This local metric on the other hand, significantly increases the computational complexity, as each distance computation involves the determination of the local neighbourhood within a segment's borders, near the contact line with the adjacent segment (Figure 9).

While more sophisticated segmentation approaches exist, the robustness of the registration methodology presented in the following sections, does not require a perfect segmentation. Furthermore typical 3D scans of fractured objects are of very high resolution and the segmentation process should be efficient in order for it to be practical.

Parametric Space Feature Extraction. It is worth mentioning here that the estimation of features for the segmentation and classification tasks (see next) can be significantly accelerated by switching all computations to the parametric texture domain of the input mesh, if available. Although the mesh pre-processing is performed once and therefore such attributes need not be re-computed frequently (or in fact, at all), having a very fast process to estimate them, renders the whole procedure an interactive, user-guided process.

For this reason, we have devised and implemented a method for the direct computation of several features in the parametric domain, using MC sampling, as detailed in [4], with low distortion and real-time feedback, even for large areas of support for the feature calculation. Our GPU-based approach exports the requested attributes as texture maps for the input mesh and the corresponding tool (see Figure 10) is

available online at: <http://graphics.cs.aueb.gr/graphics/downloads.html>.

Segment Classification. Given the distinct segments of a fragment we need to discriminate the fractured from the intact ones. We achieve this by computing a *local roughness* term $R_r(\mathbf{p})$ for an Euclidean neighborhood $\mathbf{q}_i \in S(\mathbf{p}, r) : \|\mathbf{q}_i - \mathbf{p}\| \leq r$ with corresponding normal vectors \mathbf{n}_i , as:

$$R_r(\mathbf{p}) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{n} - \mathbf{n}_i\|^2, \quad (1)$$

where \mathbf{n} is the normal at the central point \mathbf{p} and N is the count of samples taken in the $S(\mathbf{p}, r)$ neighborhood. Local roughness is similar to *local bending energy*, $e_k(\mathbf{p})$, the metric defined and used for classification by Huang et al. [28], but in our experiments the use of $R_r(\mathbf{p})$, proved better at discriminating intact from fractured surfaces.

We should note here that we deliberately and conservatively bias our classification towards false positives (intact regions marked as fractured), in order to minimize the occurrence of false negatives (fractures not identified as such). Not being able to detect a fracture may lead to missed fragment combinations, while mistaking an intact surface for a fracture only decreases the efficiency of our method. Due to this fact and the nature of the descriptor used, segments with rich geometric details are classified falsely as fractured. However, such decorative or engraved facets will not be matched against other surfaces in the geometric alignment that follows and therefore cause no problems in the process. In the rare occasion, when an intact segment with a characteristic feature curve, that could be potentially used in the matching process, is falsely classified, a user evaluation of the results eliminates any inconsistencies.

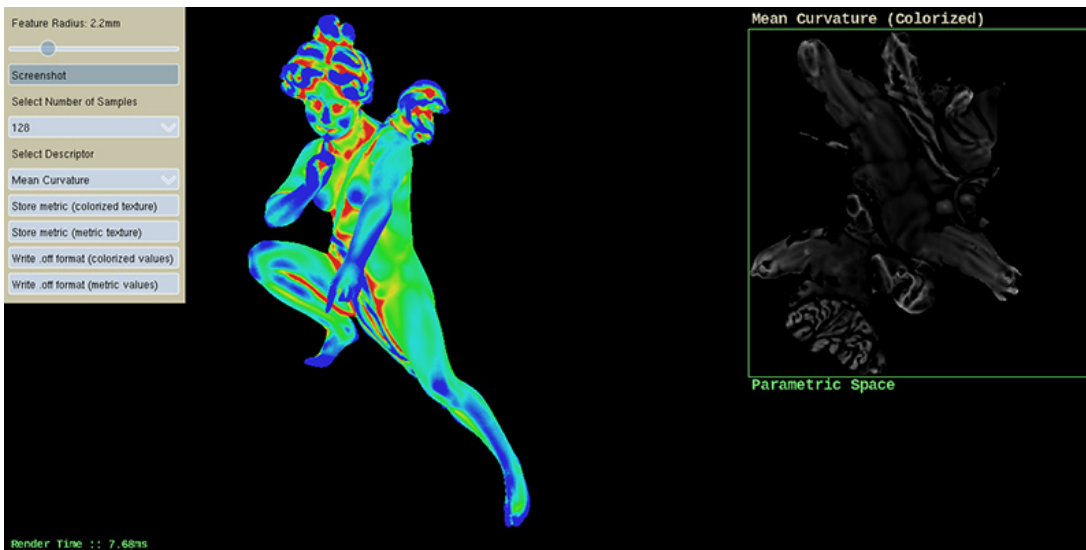


Figure 10: The parametric space, GPU-accelerated feature estimation tool interface.

In practical terms, the above classification is threshold-based, meaning that according to the data set, the user must choose an appropriate value for the local roughness threshold. Typically, this remains unchanged within a collection of compatible or similar fragments, though and it is not required to set it per object.

We have also experimented with unsupervised classification using machine learning over *Local Roughness* and *Sphere Volume Approximation*, using three different scales (r_1, r_2, r_3) , in order to better capture both subtle and large (shape) features of the surface. The specific values used in our experiments are 6mm, 11mm and 16mm. While the size of each fractured object may vary greatly, we use the same feature scales for all experiments, as these values are related to the underlying material and its distinct fracture morphology and not the surface size. For each feature we calculate the following statistical properties: a) average, b) median, c) min, d) max, e) variance. Furthermore for each vertex \mathbf{v} of the segment, we compute the local variance $\sigma_{local}^2(g, (q)_i)$ of the features g on points \mathbf{q}_i in an Euclidean neighborhood $\mathbf{q}_i \in \mathcal{S}(\mathbf{p}, r) : \|\mathbf{q}_i - \mathbf{p}\| \leq r$. The same statistical properties are computed for $\sigma_{local}^2(g, (q)_i)$ in radii r'_1, r'_2 and r'_3 with $r'_i = \frac{3 \times r_i}{2}$.

Our training dataset consisted of 55 fragments with 586 segments belonging to five fractured objects. To evaluate the classification of the generated model, we used the cross-validation methodology with 10 folds, due to its low bias and reduced variance [75], [41]⁴. Subsequently, we evaluated the true predictive performance of the generated classification models by using them in the classification of a set of 47 fragments (237 segments) belonging to three objects (brick, cake and gargoyle) from a different dataset (Vienna University of Technology⁵).

Several classification algorithms were examined, in order to evaluate their performance on predictive accuracy. To this end, we utilized the open source data mining platform *Weka* [21]. ADTree is the classification model that gave the most accurate predictions considering the constraints we discussed (100.0% TP recognition of fractured surfaces and 64.4% TN recognition of the intact ones), while both AdaBoost and LWL achieve similar TP but lower TN performance. In general, we noticed that the classification models performed better in recognizing fractured areas in the test set than in the training set. This is attributed to the fact that all fractured areas from the test set present highly rough surfaces, in contrast to the training set, where, while the majority of the fractured surfaces were rough, smooth fractured surfaces were present, as well. Finally, due to the fact that these surfaces were treated as outliers during the training phase of the classification models, we notice a lack of descriptive ability from the used features in such cases.

Although the results from the machine-learning approach were very encouraging, we decided *against adopting it* as the method of choice for the tools we developed for a practical reason: The manual threshold determination requires no training and therefore, no additional datasets to operate on and no extra procedures and external tools

⁴We would like to thank Michalis K. Titsias for his valuable insights on machine learning

⁵<http://www.dmg.tuwien.ac.at/fg4/3dpuzzles.html>

to use. Automatic classification potentially requires the adaptation of the classifier to a different range of fragments (i.e., new training). Furthermore, the threshold-based approach tends to be more intuitive and easy to control. Still, the fact that a high classification score is achievable with an automatic approach even when the classifier is trained on a completely different set, is an insightful information for future research.

2.4.2 External Feature Extraction

The main goal of our feature extraction procedure is to identify points lying on prominent linear or curved structures of significant extent on the intact surfaces of a fragment (see example in Figure 11). These features are later going to be exploited in the fragment registration process to lock non-contacting pieces in alignment.

We require that these features extend towards the fractured surfaces and therefore towards other candidate matching fragments. The features are subsequently expressed in terms of parametric curves, which we call *feature curves*. Finally, the curves are densely sampled to generate sets of surface and extrapolated curve points, respectively. Each of these extrapolated and corresponding surface feature curves is associated with one or more of the fractured faces. During the pairwise matching stage, where parts are compared across their fractured facets, only feature curves associated with the facets under examination are considered.

We describe two methods for the feature extraction. The first is semi-automatic and the second is an intuitive user-guided approach that could be used by archaeologists without technical training.

Prior to feature extraction, all intact facets are merged. It is important to note here that due to the robustness of both the salient feature curve extraction and the registration process, the accuracy of the segmentation is not critical. This is a valuable property, especially in the case of eroded fragments, where the boundaries between fractured and intact regions are frequently unclear. Other methods, such



Figure 11: Full pre-processing pipeline results. Left: Segmentation. Middle: Classified facets. Right: Extracted feature curves on intact surfaces.

as [24], need a more careful selection of break curves to avoid overestimating the intact regions, and therefore skew the extrapolated data.

Semi-automatic feature extraction. Initially, a local estimation method is used in order to compute the *Mean Curvature* H at each vertex point \mathbf{p} at multiple scales. Hult et al. [30] associate the *sphere volume integral invariant*, $V_r(\mathbf{p})$ with H as:

$$V_r(\mathbf{p}) = \frac{2\pi}{3}r^3 - \frac{\pi H}{4}r^4 + O(r^5) \quad (2)$$

where $V_r(\mathbf{p})$ represents the part of the sphere volume of radius r “inside” the surface at \mathbf{p} [73] from which we can estimate $H(\mathbf{p})$.

We compute the *sphere volume integral invariant* using its complement. McGuire et al. [55] presented a stochastic solid angle computation for the approximation of ambient occlusion in the hemisphere above a point \mathbf{p} . Inspired by this idea, we extend it to a full sphere and compute a fast approximation of the unoccupied volume of a sphere of radius r centered at \mathbf{p} via stochastic sampling. Assuming a smoothly varying tangential elevation around \mathbf{p} , the vector $\mathbf{q}_i - \mathbf{p}$ from the central point to any sample \mathbf{q}_i within the Euclidean neighborhood $S(\mathbf{p}, r)$ approximates the horizon in this direction with respect to the normal vector \mathbf{n} at \mathbf{p} at a distance scale equal to $\|\mathbf{q}_i - \mathbf{p}\|$. Taking a uniform rotational and radial distribution of samples (direction and scale) \mathbf{q}_i in $S(\mathbf{p}, r)$, we can approximate the open volume $V_o(\mathbf{p})$ above \mathbf{p} by:

$$V_o(\mathbf{p}) = \frac{4\pi r^3}{3A} \sum_{i=1}^N \frac{a_i [(\mathbf{q}_i - \mathbf{p}) \cdot \mathbf{n}]}{\|\mathbf{q}_i - \mathbf{p}\|}, \quad (3)$$

where a_i is the *voronoi region area* of q_i as defined by Meyer et al. [58] and A is the total area of the neighborhood is the total area of $S(\mathbf{p}, r)$. We compute the sphere volume integral invariant as its complement.

In our experiments we used seven scales for r at 1%–6% of the object’s diagonal and keep the median value for each point. Subsequently, we tag as feature the points with H in the higher or lower t_1 percentage of the entire set of points (see Figure 12-a), where t_1 is a user-specified threshold. Alternatively, we could restrict the extrema detection in local neighborhoods, as other methods suggest, but this would introduce additional parameterization for the neighborhood size selection.

The above set of feature points is subsequently split into feature point groups, in a greedy manner, by picking a random point and assigning neighbouring points with similar descriptor values that are within a distance r_n , to the same cluster. Due to the presence of noise (or erosion in our cases) continuous features may be interrupted. In order to address this issue, similarly to [70] we use a larger threshold t_2 and store the extra points between the two thresholds in a separate list that is used only for the

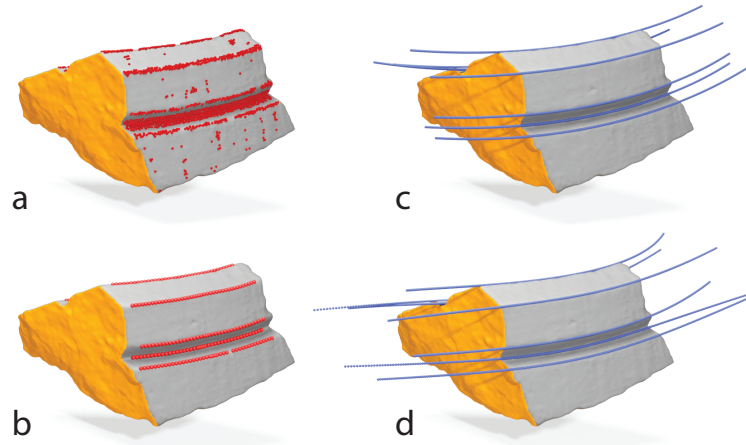


Figure 12: a) Feature points extracted using our semi-automatic approach, b) Skeletal points using [26], c) Quadratic B-Spline approximation of feature evaluated in $-0.5 \geq t_i \leq 1.5$, d) Cubic B-Spline approximation of feature evaluated $-0.5 \geq t_i \leq 1.5$.

bridging of gaps. Since we are only interested in features that span across multiple fragments, we keep features with at least one end near a fractured facet and discard the rest. The value of t_1 is set according to the model topology and the features we want to extract. t_2 in most of our experiments was set to $t_1 = 0.1$ but its value may also vary.

Having extracted the feature point groups, we trace their trajectory using a parametric curve. Due to the underlying fluctuations of the surface, the generated point set can be noisy or sometimes incomplete and thus a filtering step is essential, prior to any curve approximation. Several techniques exist in the literature for curve reconstruction and can be applied to our problem. Wang et al. [85] use an iterative quadratic minimization method, while Yang et al. [51] use a least squares minimization in order to compute an approximate B-spline of the input point cloud. Here, we utilize the skeleton extraction method of Huang et al. [26] that uses the L_1 -median metric and does not impose specific requirements about the geometry or the topology of the initial point set. The method is robust to noise and outliers and in our experiments it did not require many parameter modifications (see Figure 12-b). The resulting skeletal points can be easily approximated by uniform curves using least-squares fitting.

A parametric curve is of the form $\mathbf{P}(t) = (x(t), y(t), z(t))$ and is continuous in the range $0 \leq t \leq 1$. In order to extrapolate the feature curves we evaluate $\mathbf{P}(t)$ at $t_i < 0$ and $t_i > 1$ requiring that the parametric curve maintains C^1 continuity at the endpoints of the curve. We generate a *surface feature point set* and one or two *extrapolated feature point sets* in the parametric intervals $-0.5 \geq t_i < 0$ and $1 < t_i \leq 0.5$ for the extrapolating sets and $0 \geq t_i \leq 1$ for the surface set (see Figure 12-c). The extrapolated feature points are essentially a hypothesis we make for the shape of the extents of the features. Therefore, the farther a point on the

extrapolated curves is (i.e. outside the range $0 \leq t \leq 1$), the larger the uncertainty about its true location becomes. In order to properly model this, when we use these points in the matching process, we apply a weighting term that suppresses the contribution of the points the farther they are from the actual surface feature interval.

B-spline parametric curves fit our purpose as they provide more flexibility than Bézier curves due to the fact that the degree of a B-spline curve is separated from the number of control points. In our experiments, quadratic B-splines (2nd degree) with 3 – 5 control points gave the best extrapolation results. Initially the skeletal point set is approximated with 3 control points and based on a predefined threshold for the fitting error f_e , control points are increased up to 5. In cases where 5 control points are not sufficient for the fitting process, we split the skeletal point set in half and re-iterate. The use of cubic or higher degree B-splines results in high oscillations in the extrapolated part of the curve and cannot be used for our purposes (see Figure 12-d).

User-guided feature selection. No matter how robust a feature detection method is, it is still required by the user to modify specific parameters in order to achieve the desired result. Alternatively, we propose a user-guided process that actually speeds up the overall process and extracts all the desired features in a robust and efficient manner.

The user selects a small portion of the fragment with a brush tool, which includes the desired feature. Initially, using the median H of multiple scales for each point, we detect the extrema over the selected area and subsequently extract the feature points. Using again the skeletonization method of [26] we filter the feature point group and obtain the actual B-spline curve that locally describes the intended structure. Using extrapolation in the parametric intervals $-0.5 \geq t_i < 0$ and $1 < t_i \leq 0.5$, we search for points with similar descriptor values that are close to the extension of the curve. The newly discovered points are added to the feature point group and the feature curve is re-evaluated. The process is repeated until no more points can join the feature set. The resulting feature curve is presented to the user, along with its extrapolation that will be used in the matching step (see Figure 13).

2.5 First Tier - Contact Surface Registration

The first tier of our geometric registration approach is an automatic process, whose goal is to find all matching fragment pairs sharing a significant contact surface, given the complete list of fractured facets per fragment. Initially, the optimal relative rigid transformation and the corresponding matching score between every possible combination of fracture facets that belong to different fragments is computed. This score is based on the residual distance between the two fragments after their rigid registration. In fact, fragment registration and matching error measurement is equivalent to generic rigid registration for other application, such as partial scan alignment. This is why we employ the three-level coarse-to-fine search strategy (Mavridis et al. [54], [53]) that we developed to jointly address both the contact-surface fragment registration

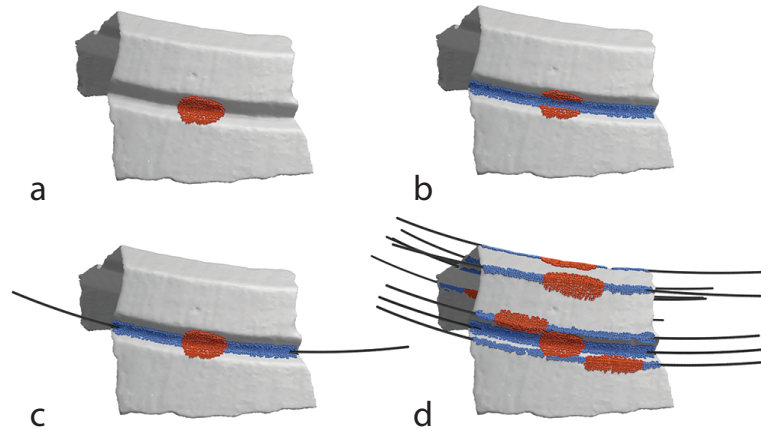


Figure 13: a) User selects a set of points, b) Extraction of feature points, c) Extrapolated points using the Quadratic B-Spline approximation of the feature, d) Extracted feature point sets and extrapolations.

Related Publications

P. Mavridis, Andreadis, A., and G. Papaioannou. [Efficient Sparse ICP](#), Computer Aided Geometric Design (Proc. Geometric Modeling and Processing), 2015.

P. Mavridis, Andreadis, A., and G. Papaioannou. [Fractured object reassembly via robust surface registration](#). Eurographics Conference (short paper). 2015.

A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., [Fractured 3D Object Restoration and Completion](#). ACM SIGGRAPH Poster Session, 2015.

and the scan alignment requirements of predictive scanning (see also D2.4 - Predictive digitization methodology report). The specialization of this generic registration strategy for the case of fragment alignment is elaborated below.

The results of this first-tier pairwise matching step are fed to a combinatorial optimization stage that aggregates fragments according to global error minimization and geometrically assembles the fragments into (potentially partial) clusters that represent the reconstructed objects.

While this problem has received substantial scientific interest, human evaluation of the results is still crucial, since frequently, non-matching fragments can be successfully registered, given that the registration algorithm has to be robust to noise and outliers in order to handle surface erosion and scanning defects.

2.5.1 Problem formulation

Given a source and a target point cloud representation of two fractured facets \mathcal{X} and \mathcal{X}' belonging to two distinct fragments, we formulate their registration as a distance \mathcal{F}_{surf} minimization problem:

$$\arg \min_{\mathbf{M}} \mathcal{F}_{surf}. \quad (4)$$

\mathbf{M} is the rigid alignment transformation matrix. \mathcal{F}_{surf} is defined as:

$$\mathcal{F}_{surf} = \sum_{i=1}^n \phi(\mathbf{M}\mathbf{x}_i, \mathcal{X}') + \sum_{i=1}^{n'} \phi(\mathbf{M}^{-1}\mathbf{x}'_i, \mathcal{X}), \quad (5)$$

where n, n' are the number of points of the fractured surfaces \mathcal{X} and \mathcal{X}' respectively. Equation 5 explicitly enforces a two-way distance minimization between the fractured surfaces of the fragments. The function $\phi(\mathbf{a}, \mathcal{B})$ measures the distance of an arbitrary point $\mathbf{a} \in \mathbb{R}^3$ to a point set \mathcal{B} and is defined as

$$\phi(\mathbf{a}, \mathcal{B}) = \min_{\mathbf{b} \in \mathcal{B}} \phi(\mathbf{a}, \mathbf{b}), \quad (6)$$

Choice of metric. For the metric $\phi(\mathbf{a}, \mathbf{b})$, which measures the distance between two points in space, we employ here the formulation discussed in D2.4 - Predictive digitization methodology report and [10], which is provided below for completeness. The distance metric in Equation 6 uses the ℓ_p -norm formulation with $0 < p < 1$:

$$\phi(\mathbf{a}, \mathbf{b}) = \mu_p(\|\mathbf{a} - \mathbf{b}\|_2), \quad \mu_p(x) = |x|^p$$

As shown in Figure 14, using Euclidian distance or an ℓ_1 norm, will skew the solution in order to minimize the residual gaps "on average", drastically penalizing distant points. In the case of fractured objects, however, due to erosion and shape differences on the break surfaces, there is a strong presence of outliers. The smaller p is set, the lower the bias towards the outliers becomes, which allows the optimization method to converge to the desired solution (ℓ_0 - contact area maximization).

2.5.2 Optimization Strategy

We summarize here the generic optimization strategy, which is used in both the scan registration and fragment registration problems and establish the modifications

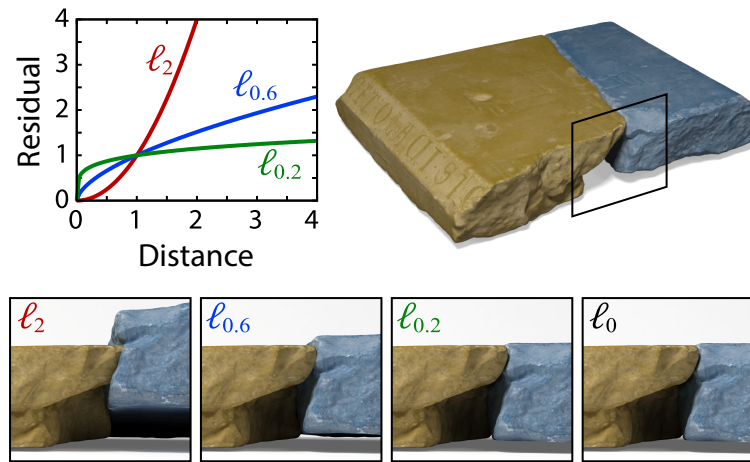


Figure 14: Alignment of two fragments using ℓ_p -norm minimization on the fractured surfaces. Missing parts on the left fragment create outliers, resulting in sub-optimal registration when using the ℓ_2 -norm. The contribution of large distances is reduced with ℓ_p -norms, making the method robust to outliers for sufficiently low values of p .

specific to the task of reassembly. The generic three-stage pose optimization strategy is presented in detail in [54], [53] and is also explained in D2.4 - Predictive digitization methodology report.

Coarse Initial Alignment. When the surface area of two facets is roughly equal, we perform a coarse initial alignment of the centroids and the average normals of the facets. Otherwise, we compute the alignment using a standard RANSAC-based alignment procedure that is based on 3-point congruent sets [16]. The 4-point congruent sets method [1] provided similar results and might be preferable in more noisy datasets. Because of this initial alignment, the original pose of the fragments is inconsequential.

Simulated Annealing. In the next stage, the alignment transformation is parameterized using three free variables for the translation and three (euler angles) for the rotation. These variables are initialized with the alignment of the previous stage and a *Simulated Annealing* method is used to further minimize the residual distance, as measured in Equation 5. The range of the search is restricted around the alignment of the previous stage. Our implementation uses the *Enhanced Simulated Annealing* [77] method, configured to perturb one variable at a time. To increase the efficiency of the method we use a uniformly sampled subset of points from the source surface in our computations.

Local Refinement. In the last geometric alignment stage, the computed registration is locally refined using the sparse ICP algorithm [10]. The choice of Sparse ICP is an important one, because unlike previous approaches, it can handle the outliers created from missing parts or eroded surfaces without introducing any application-specific heuristics or weighting functions. Furthermore, this stage of the pipeline uses accurate distance queries, instead of the discretized distance field, to avoid compromising the

quality of the final alignment and to allow the use of low resolution distance field volumes in the previous stages.

Non-penetrating Registration. What is unique in the case of fragment registration as opposed to a generic rigid alignment problem, is the fact that we require that the two objects under examination do not penetrate each other, up to an error margin (slight penetration is tolerable, as it is usually the result of digitization errors). Penetration testing is important both during the pairwise alignment, but also in the combinatorial optimization stage of the multi-part registration, since we seek the validation of particular pairs within larger clusters of fragments; non-penetrating fragment pairs can still cause overlaps with other fragments of a cluster, a fact that must be both detected and penalized accordingly (see multi-part registration below).

To enforce non-penetrating registration of two closed shapes, when a point of the source shape is in the interior of the target shape, then the corresponding candidate alignment should be rejected, by assigning a sufficiently high value to the cost function. This can be easily achieved by significantly penalizing the residual distances of the penetrating points. Penetrations are detected very efficiently using the *Signed Distance Field* (SDF) representation of the target shape, where the distances of points inside the corresponding shape are negative (see Figure 15-left). In this case, we use the distance value obtained from the SDF $\phi(\mathbf{a}, \mathcal{B})$ in a modified cost function $\phi'(\mathbf{a}, \mathcal{B})$:

$$\phi'(\mathbf{a}, \mathcal{B}) = \begin{cases} \phi(\mathbf{a}, \mathcal{B}) & \phi(\mathbf{a}, \mathcal{B}) \geq 0 \\ -\alpha\phi(\mathbf{a}, \mathcal{B}) & \phi(\mathbf{a}, \mathcal{B}) < 0 \end{cases} \quad (7)$$

where the parameter α controls the penalty of negative distances. In our experiments, this value was set to 100. Figure 15-right demonstrates the non-penetrating registration using this approach.

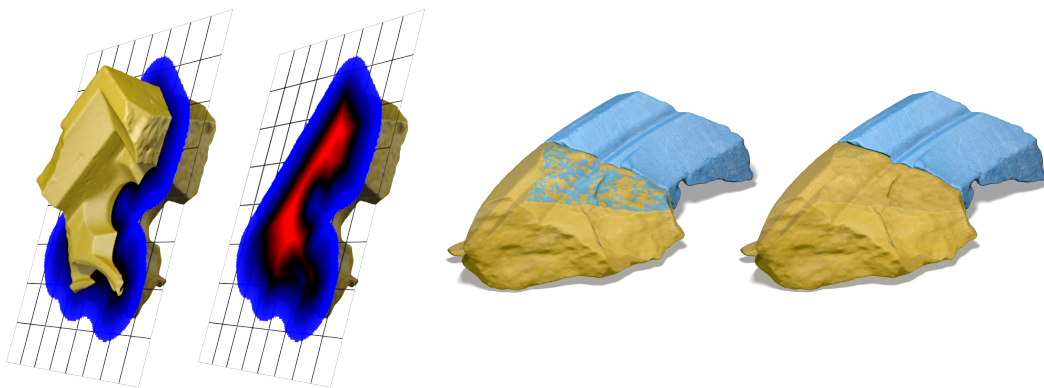


Figure 15: Left: Signed distance field for a fragment mesh. Right: Registration results without and with penetration testing and penalization. To better illustrate the mutual penetrations, the fragment in front is rendered with transparency.

The construction of the SDF assumes that input shapes are manifold triangle meshes. For non-manifold meshes, penetrations can be detected using the normals of the target surface. In particular, a point \mathbf{a} does not penetrate the target surface if $\mathbf{n}_b \cdot (\mathbf{a} - \mathbf{b}) \geq 0$, where $\mathbf{b} \in \mathcal{B}$ is the closest point to \mathbf{a} and \mathbf{n}_b is the normal at \mathbf{b} . While this formulation does not require manifold shapes or triangle connectivity, it relies on expensive closest-point queries, therefore in our method we use the SDF, where possible.

Since the inclusion of arbitrary alignment constraints, such as penetration avoidance, is not trivial for ICP methods, when such constraints are required, we perform the last stage of the pipeline (local refinement) using the ESA method with a very small search parameter range, instead of an ICP method.

2.5.3 Results Refinement

Some results of registration of compatible fragments using the First-tier fragment registration are presented in Figure 16). Pairs with significant contact area overlap and low error after the alignment are retained as valid combinations and the others are trimmed. This "valid" set of pairings can be further inspected by the expert user before proceeding to the next step (multi-part registration). During this inspection stage or even after the final clustering proposed by the multi-part registration, certain combinations of facets may be *white-listed* or *black-listed*. For example, in Figure 17, the fragment facet combinations shown exhibit a good matching score. However, the resulting shape cannot be confirmed by the CH expert. Additionally, some combinations will eventually cause penetration problems during the multi-part assembly stage and will be automatically rejected. In any case, it is possible to manually dictate the exclusion or inclusion of a particular pair in the set of valid candidate combinations, at any stage.

To be able to better discriminate valid solutions amongst the high-ranking pairs or, more importantly, discover valid combinations in the low-ranking fragment combinations, the user can initiate the second-tier registration for a set of selected fragments.

2.6 Second Tier - Feature Curve Object Registration

Related Publications

A. Andreadis, G. Papaioannou, and P. Mavridis. [Generalized digital reassembly using geometric registration](#). In Proc. IEEE/EG Digital Heritage, 2015.

A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., [Fractured 3D Object Restoration and Completion](#). ACM SIGGRAPH Poster Session, 2015.

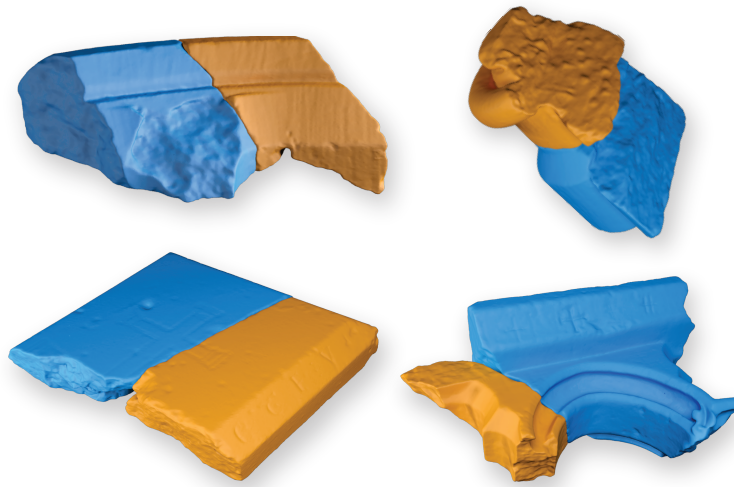


Figure 16: Pairwise matches obtained using the first tier: contact surface registration. Fragments are colored for visual clarity.

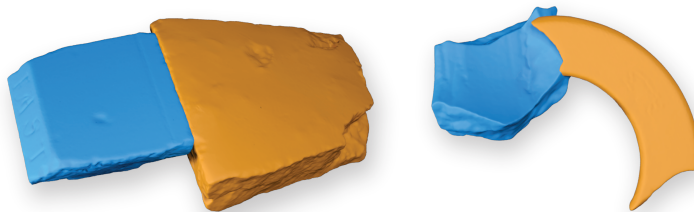


Figure 17: Failure cases of the surface-based registration, that are rejected by the expert user (left) or cause part penetration when driving the multi-part registration stage (right).

The second tier of the proposed geometric registration approach is a semi-automatic user-guided process that targets the pair-wise registration between fragments with significant erosion or large missing parts, i.e. pairs sharing minimal contact surface, where contact surface-based registration approaches fail (see Figure 17). Due to the nature of this problem, we require the input of an expert, who initially identifies fragments with similar characteristics that could be potentially matching and subsequently, using the mechanisms described in Section 2.4.2, extracts the the salient feature curves on the intact surfaces of the fragments.

2.6.1 Problem formulation

Similar to the contact-surface-based registration, given a source and a target point cloud representation of two fractured facets \mathcal{X} and \mathcal{X}' belonging to two distinct fragments, we now minimize *both* distance between them (\mathcal{F}_{surf} term) and the distance between the extrapolated-surface feature curve points (\mathcal{F}_{curve} term) associated with \mathcal{X} and \mathcal{X}' . We denote as \mathcal{E} and \mathcal{E}' the extrapolated points sets and as \mathcal{S} and \mathcal{S}' the surface feature-curve point sets associated with fractured facets \mathcal{X} and \mathcal{X}' respectively (see Figure 18). We formulate this as a minimization problem that combines

\mathcal{F}_{curve} and \mathcal{F}_{surf} in a complementary form:

$$\arg \min_{\mathbf{M}} (c \cdot \mathcal{F}_{curve} + (1 - c) \cdot \mathcal{F}_{surf}). \quad (8)$$

c is the relative contribution of curve features and \mathbf{M} is the rigid alignment transformation matrix. \mathcal{F}_{surf} and \mathcal{F}_{curve} are defined as:

$$\mathcal{F}_{curve} = \sum_{j=1}^k w_j \phi(\mathbf{M} \mathbf{e}_j, \mathcal{S}') + \sum_{j=1}^{k'} w'_j \phi(\mathbf{M}^{-1} \mathbf{e}'_j, \mathcal{S}), \quad (9)$$

$$\mathcal{F}_{surf} = \sum_{i=1}^n \phi(\mathbf{M} \mathbf{x}_i, \mathcal{X}') + \sum_{i=1}^{n'} \phi(\mathbf{M}^{-1} \mathbf{x}'_i, \mathcal{X}), \quad (10)$$

where k , k' and n , n' are the number of points of \mathcal{E} , \mathcal{E}' and \mathcal{X} , \mathcal{X}' respectively. w_j and w'_j (Eq. 9) is an exponential falloff weighting term that decreases the contribution of a measurement, the farther its location is from the parametric curve's valid range, as described in 2.4.2.

For the distance metric, see Section 2.5.1. Here too, the distance function for the contact surface alignment constraint is discretely sampled on a 3D grid that extends over the narrow band of \mathcal{X}' . On the other hand, for the extrapolated and surface feature point sets \mathcal{E} and \mathcal{S} , respectively, we use accurate distance queries using a kd-tree structure, as they contain only a small number of points.

2.6.2 Initial Alignment, Global and Local Search

Similar to the first-tier registration, we perform a coarse-to-fine alignment procedure. In the original method, two alternative schemes for the initial rough alignment of the

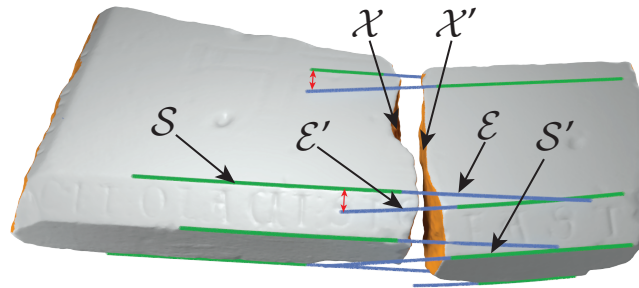


Figure 18: Visualization and annotations of the fractured facets \mathcal{X} , \mathcal{X}' , the extrapolated feature-curve points \mathcal{E} and \mathcal{E}' and the surface feature-curve point sets \mathcal{S} and \mathcal{S}' used in Equations 9, 10.

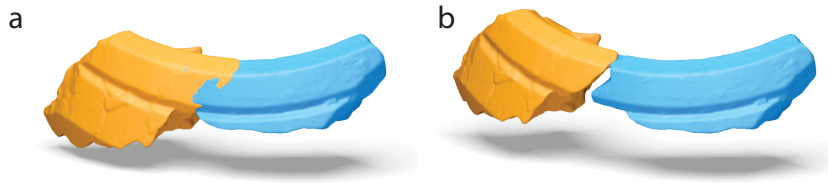


Figure 19: Pairwise alignment using our method. On the left, the result after the global optimization step and the local refinement and on the right after the penetration minimization, step using the rigid movement defined by the feature curves.

point clouds are discussed, one using the average fracture surface planes extracted via PCA and a RANSAC-based alignment procedure. However, in our case, the addition of the extrapolated features always provides a good guidance for the reassembly and the RANSAC pose estimation is not necessary. The initial alignment drives the global search strategy, which is performed using the Enhanced Simulated Annealing stage in order to find the optimal transformation that minimizes the residual distance of Equation 8.

After the annealing process terminates, the resulting transformation is refined using the Sparse ICP [10] (SICP) using only the \mathcal{E} , \mathcal{E}' , \mathcal{S} and \mathcal{S}' feature point sets. After the SICP refinement, we compute the fractured surface overlap in order to detect if a usable contact area between the two fragments exists. If either \mathcal{X} or \mathcal{X}' is covered by 10% we perform another SICP pass, this time using the surface points. In our experiments the robust behavior of the ℓ_p -ICP with $p \in [0, 1)$ allowed us to further refine our pairwise matches, while other ICP variants could not handle the large amount of missing data. This final refinement step is not essential to our methodology, but we need to remind here that, while use of the extrapolated point set allows us to reach efficiently into a good matching solution, they are still predicted data based on a hypothesis and minor misalignments cannot be avoided if feature curves are exclusively used.

2.6.3 Penetration-free Registration

While tier 1 registration tries to address penetration using the signed distance field to penalize negative distances, in our cases due to the small contribution of the contact surface term, penetration might exist after the global search step. However, we can achieve penetration-free registration using the rigid motion defined by the feature curves. We select the feature curve \mathcal{F} with the smallest alignment error (extrapolated and feature points) of the source fragment and using an iterative process, we move the fragment that \mathcal{X} resides on with a small fixed step along \mathcal{F} until there is no penetration (see Fig. 19). The same approach can be used to present results of pairings that do not share a contact surface, by allowing the user to "slide" a fragment along the dominant feature curves, thus generating a family of infinite solutions.

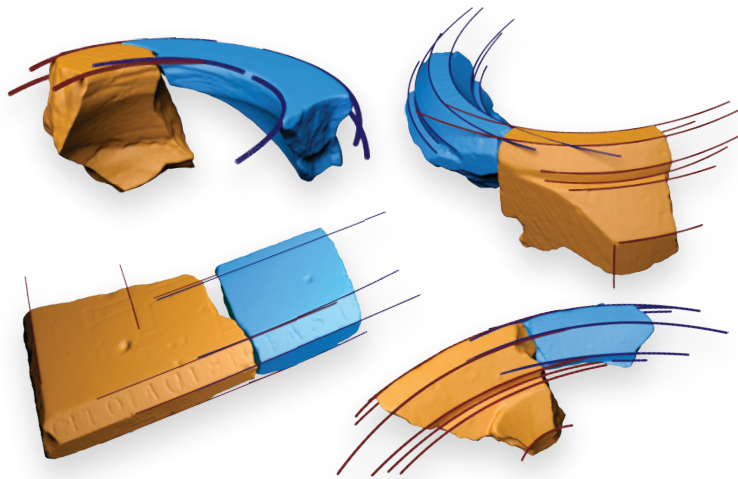


Figure 20: Pairwise matches obtained using the second tier, with the combined feature curve and contact surface registration.



Figure 21: Assembly of a decorative arch segment out of many similar fragments and parts. Both tier 1 and tier 2 registration steps were utilized for this example.

Results of the second tier registration are presented in Figure 20. Potential pairings between fragments with even large gaps or incompatible contact surfaces are detected and reported, based on the shape of the salient morphological regions on the intact surfaces. This stage provides an additional match discovery functionality to the CH expert, which has been proven in practice to work on reliably on ambiguous fragment combinations such as the one presented in Figure 21.

In Figure 22, the tier 2 registration resulted in similar pairings of a fragment with other three parts, non of which can be validated using the contact surface, as there is clearly an incompatible break, possibly caused by a missing chunk. For such cases, it is imperative that the CH expert intervenes and black-lists some of the proposed solutions, according to interpretation and prior knowledge.

2.7 Multi-part Assembly

Combinatorial optimization. Given the matches and respective matching errors generated in the pair-wise alignment stages, in this step of the reassembly, we compute the set of fragment clusters and corresponding global transformations of the fragment

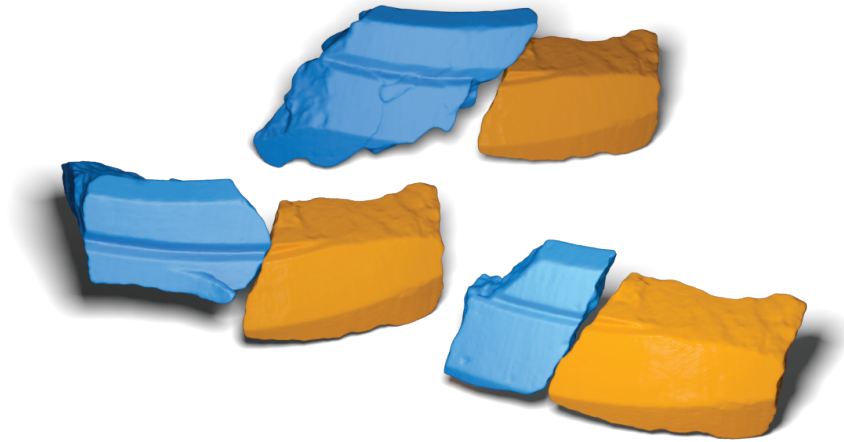


Figure 22: Three pair-wise alignments obtained using our approach. The orange fragment is the same in all cases. Such ambiguities are attributed to the fact, that all objects share similar characteristic feature curves and none of them is fully validated by the contact surfaces.

Related Publications

A. Andreadis, G. Papaioannou, and P. Mavridis. [Generalized digital reassembly using geometric registration](#). In Proc. IEEE/EG Digital Heritage, 2015.

P. Mavridis, Andreadis, A., and G. Papaioannou. [Fractured object reassembly via robust surface registration](#). Eurographics Conference (short paper). 2015.

A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., [Fractured 3D Object Restoration and Completion](#). ACM SIGGRAPH Poster Session, 2015.

meshes. We approach the problem in a similar way to [29] with some modifications. A graph is constructed, where fragments are represented as nodes and pairwise matches are the edges between nodes. Kruskal's algorithm is used in order to compute the Minimum Spanning Tree (or Forest, in the case we need to handle mixed puzzles).

Since pairwise matches are only plausible partial solutions, this process might lead to erroneous results with conflicting geometric transformations causing the overlap of certain fragments. In order to eliminate these occurrences, we employ penetration tests during the merging operations of sub-forests. If a penetration is found, the edge weight is readjusted by a fixed value and the algorithm continues. This is a heuristic and does not guarantee convergence to the correct result. Furthermore, this process might lead to a dead end, which is detected if all other valid edges have been tested and the penetration still occurs. In this case, we backtrack, by re-adjusting all the weights to their original values and modifying the weight of the previously accepted edge. A more exhaustive solution to the problem would be to find all

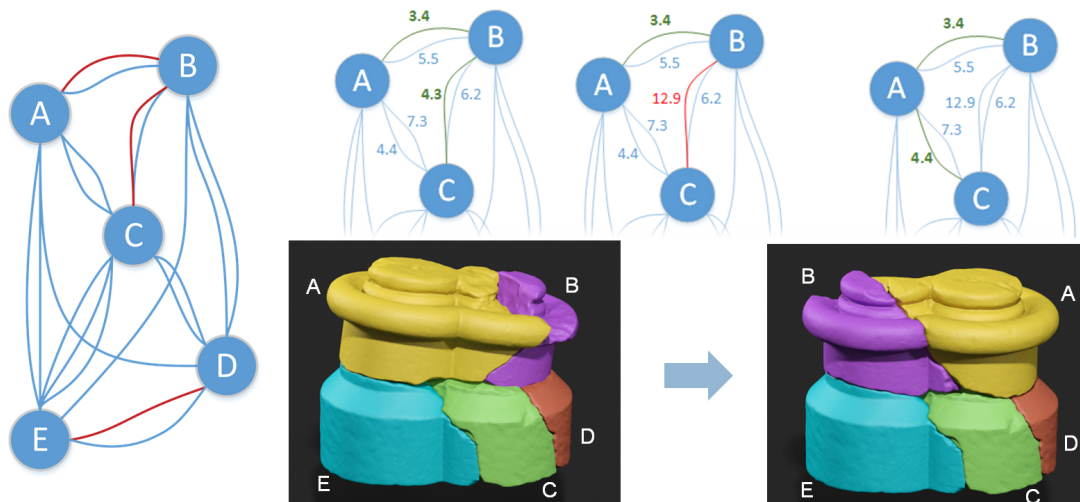


Figure 23: Multi-part assembly. The combinatorial optimization step clusters fragment combinations according to the minimum spanning forest among all fragment facet pairings (left). Middle to right: penetration testing given a current state, penalization of penetrating combination and re-evaluation of clusters.

spanning trees and validate the resulting reassemblies using both penetration testing and total fractured area coverage as there might exist more than one penetration-free reassembly. The problem with this solution is that the number of spanning trees in a complete graph with n vertices is n^{n-2} [14], which renders the approach prohibitive for a large number of fragments.

Figure 23 demonstrates with a simple example the principle of the multi-part registration along with the penetration penalization mechanism. According to an error threshold, fragments may form isolated clusters within the fragment facet combination graph. When a new connection is established (a facet combination between part B and C here), the new fragment is tested against the connecting sub-graph and if a penetration is found, the link (arc) is penalized by increasing its error (shown in bright red). Then, the best path is re-evaluated and the next best link is established, which is also tested for penetration and so on.

Multi-part registration refinement. The final position of the fragments is refined using an iterative multi-part local registration step, that takes into account both feature curves and surface points. At each step of the iteration, for each fragment, initially we locate the subset \mathcal{Y} of all points on its fractured facets, whose distance from other fragments is below a threshold. \mathcal{Y} represents the set of points potentially in contact with other fragments and excludes points near gaps of the assembly (missing geometry). The corresponding set of points of all the other fragments is denoted as \mathcal{Y}' . We need to clarify here, that \mathcal{Y}' consists of points of both fractured and intact surfaces of the other fragments, as we do not want to make assumptions about the underlying segmentation. Using \mathcal{Y} and the complete set of feature curve points \mathcal{Z} ($\mathcal{E} \cup \mathcal{S}$), we solve the minimization problem similarly to Eq. 8:

$$\arg \min_{\mathbf{M}} (c \cdot \mathcal{F}'_{curve} + (1 - c) \cdot \mathcal{F}'_{surf}). \quad (11)$$

\mathcal{F}'_{curve} and \mathcal{F}'_{surf} are defined as:

$$\mathcal{F}'_{curve} = \sum_{j=1}^k w_j \phi(\mathbf{M}\mathbf{z}_j, \mathcal{Z}'), \quad (12)$$

$$\mathcal{F}'_{surf} = \sum_{i=1}^n \phi(\mathbf{M}\mathbf{y}_i, \mathcal{Y}'), \quad (13)$$

where \mathcal{Y}' , \mathcal{Z}' are the complete set of fracture points and the complete set of feature curve points of all the other fragments, respectively. \mathbf{z}_j are the curve samples on the current fragment's curves and \mathbf{y}_i are the points in \mathcal{Y} .

Using this formulation we are able to refine even cases where large parts of the reassembly are missing (see Fig. 24) and where a typical multi-way ICP registration would fail. At this point, we did not take special care to avoid new penetrations, as our experiments did not present such an issue. Nevertheless, one could adopt the point-to-plane ℓ_1 -registration approach of Flöry and Hofer [18] in order to guarantee the multi-part penetration-free alignment.

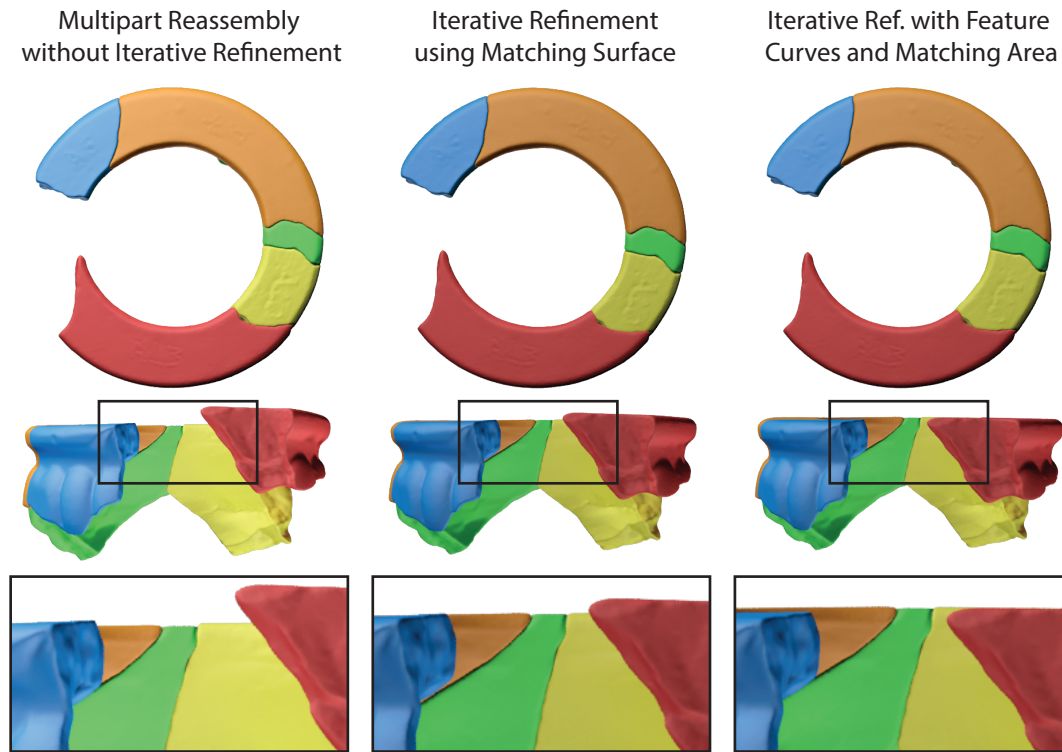


Figure 24: Object reassembly with large missing parts, before iterative refinement (left), using only contact area to refine the solution (middle) and using our combined approach taking advantage of feature curves and contact surface (right). While from the top view all results appear correct the side view shows a skew of the shaped which is fixed with our approach.

2.8 Third Tier - Symmetry-based Fragment Registration

Related Publications

A. Andreadis, G. Papaioannou, and P. Mavridis. [Generalized digital reassembly using geometric registration](#). In Proc. IEEE/EG Digital Heritage, 2015.

I. Sipiran, R. Gregor, and T. Schreck. [Approximate symmetry detection in partial 3d meshes](#). Computer Graphics Forum (proc. Pacific Graphics), vol. 33, no. 7, pp. 131-140, 2014.

The multi-part reassembly step produces objects that are the combination of multiple pairwise alignments, but often, some fragments or clusters of fragments are completely disconnected from the reassembly due to missing parts, bad contact area or the absence of feature curves. In order to address those cases, we proposed a novel approach that exploits object symmetries in order to guide the placement of the remaining fragments (Andreadis et al. [3]).

While symmetry detection is a well-researched area, only a few of the methods focus on objects with large missing data. Sipiran et al. [78] use a surface function based on

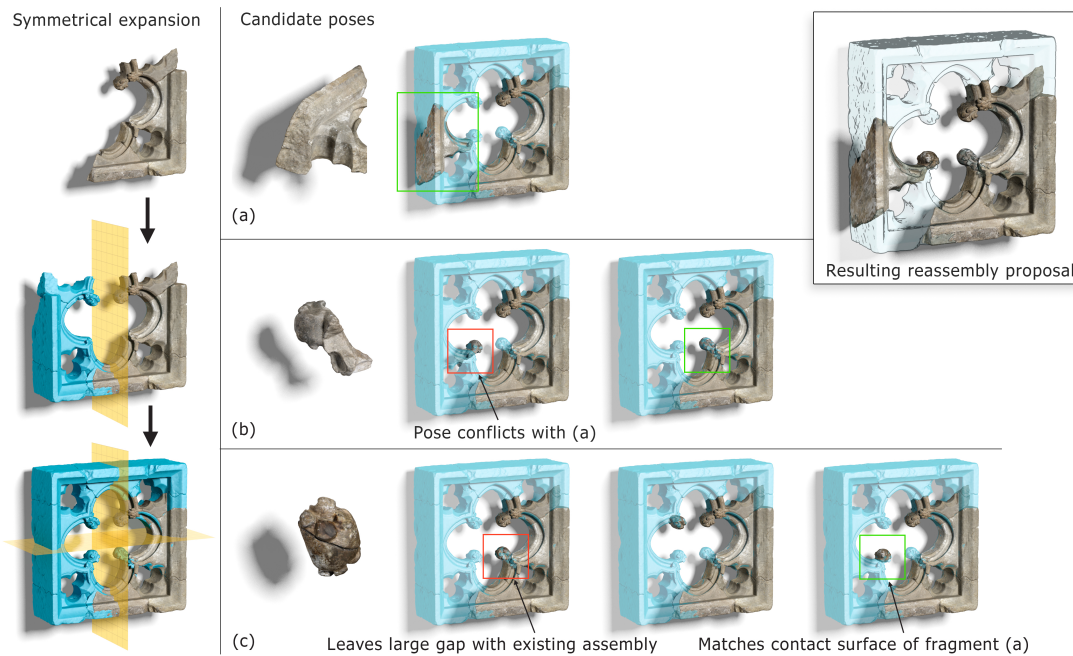


Figure 25: Third tier registration: Symmetry-based fragment registration. Left: using the partial reassembly, we extract a symmetrical expansion of the object. Middle and right: Potential candidate poses for the disjoint parts (a),(b) and (c) are generated and filtered by the CH expert, to result in the final reassembly of the object.

heat diffusion and a partiality-aware voting algorithm to address the problem, even in the challenging case with large missing parts. The method essentially locates the best candidate planes of symmetry on the object and using them one can fill the missing geometry as shown in the application examples in [78].

Since computer-generated "predicted" geometric information of the missing geometry is of little interest to the cultural heritage community, we steer the results of symmetry-based completion to a usable direction in our case; using either the method of [78], or even user-supplied planes of symmetry, under a symmetrical interpretation of the evolving shape, we generate only the missing geometry of an object and use it as a geometric prior (guide shell) to match and align any disconnected fragments. Thus, the symmetrical expansion of the assembly generated so far is never included in the final solution but rather only assists our system suggest potential matches of disjoint fragments, which are inspected by the CH expert. The whole processes is demonstrated in the example of Figure 25.

In this registration step we follow a different surface-based search strategy. The 4PCS method of Aiger et al. [1] was designed to perform robustly in the presence of extensive outliers, using four-coplanar point bases to detect hypotheses (potential alignments) and a RANSAC-style search strategy. Due to the exhaustive coverage of the search space and its linear time complexity, the method is impractical, but its recent improvement (Super4PCS) by Mellado et al. [56] reduces the time complexity to quadratic and presents a good compromise between efficiency and robustness.

Using Super4PCS we generate all the possible registration solutions and sort them according to their registration score. We filter out similar results and present to the user the k -best registrations (where k is specified by the user), after refining them using Sparse ICP. Let us note that, while in the general registration case, Super4PCS requires as input the expected overlap, here we are solving a more specific problem (part-in-whole matching) and this specific parameter is automatically generated using the ratio of the diagonals of the bounding boxes of the two objects.

2.9 Evaluation Methodology, Results and Discussion

2.9.1 Cultural Heritage Datasets

In order to evaluate our proposed methodology we performed several experiments with real cultural heritage data, a subset of which is presented throughout our figures. Keep in mind that for these tests there is no ground truth object, so the validity of the results is based on visual inspection.

Tier 1 pairwise Tests. Pairwise alignment results with fragments that share good contact areas can be seen in Figure 16. These results were obtained with the first-tier contact surface registration, without user input, using the same optimization step parameters for the entire collection. Experiments were performed on a Core i7-3820 processor and on average, each facet combination required 0.6 seconds.

Using the same parameters in Figure 17 we see failure cases of the first tier registration, due to the small contact area between the fragments. Such fragment pairs can be successfully aligned, using the second-tier registration approach that utilizes both contact surface and salient feature curves extracted from the intact surfaces.

Tier 2 pairwise Tests. Successful registrations of fragment pairs with small contact area shown in Figure 20. The two left results were obtained primarily with the feature curves, using higher values of c parameter (see Equation 8) as the contact area of the fragments is not reliable. In contrast, in the two right examples the contribution of the two terms was almost equal.

While the second-tier approach requires more initial processing for the extraction of the salient feature curves, the optimization step is of similar complexity to the first tier and we observed similar times for the computation of each facet combination. This is due to the fact that curve-to-curve distance is measured using a regular point sampling; curves are first converted to a point representation (point sets) and distances are then evaluated with fast point to point set queries, accelerated via a k -d tree.

Multi-part assemblies. Complete object assemblies are presented in Figure 26 and Figure 27, where we also show photos from the real fragments alongside the achieved reassembly. Results in Figure 26 and Figure 27-top were obtained using solely pairwise



Figure 26: Assembly results, using CH datasets.

alignments generated with the first tier registration and the multi-part reassembly step we described earlier, without any user input. The large cluster in Figure 27-bottom was obtained using alignment of both contact surface and feature curves. As we can see from the photograph of the fragment heap, many pairwise results, as also the large reassembled part obtained with our method, had not been discovered prior to the digital reassembly process, mainly due to the small contact area shared between fragments and the large number of structurally similar pieces.

Tier 3 Tests. Finally, in Figure 25 we showcase the use of our third-tier symmetry-based registration on another fractured object. The multi-part reassembly step resulted in one large cluster of fragments, two disconnected fragments and one small cluster of two fragments that was also disconnected from the rest of the reassem-

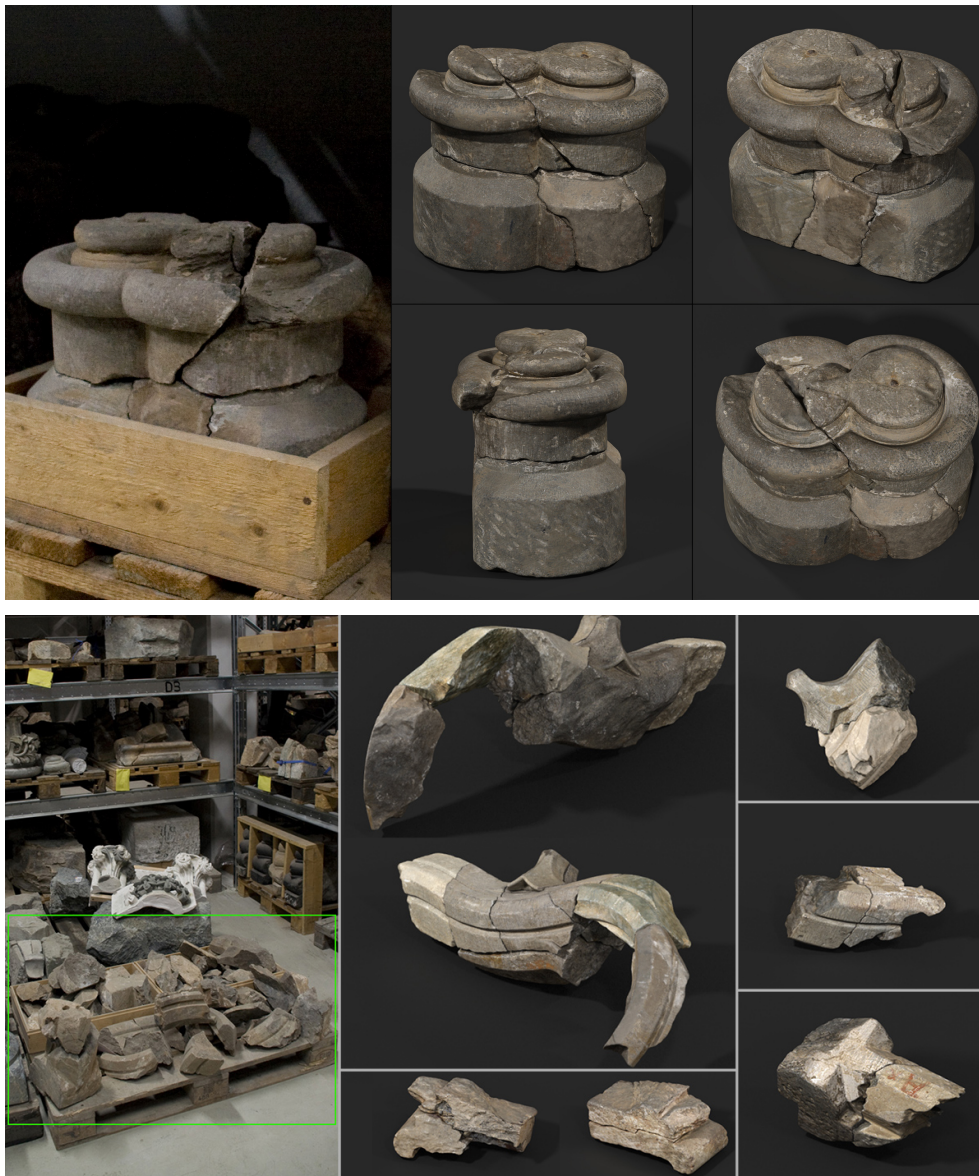


Figure 27: Assembly results, using CH datasets.

bly. On the left of the figure the large cluster is shown, along with evolution of the predicted shape of the entire object. To this end, the expert's hypothesis of a symmetrical shape enabled the exploitation of two planes of symmetry to automatically generate a geometric expansion of the virtual object and use it as a guide to align the disjoint fragments. The symmetrical, complementary expansion can be generated either by the user or by an automated process (in this case the expert user). This complementary part is subsequently used, in order to find plausible positions for the disconnected fragments as show in (a) and (b) and (c). It is crucial to note here that this process is mainly guided by the CH expert as explained in detail in the figure. In this particular example, part (a) was uniquely matched with significant overlap with the complementary shape only in one position. For part (b), two proposals with significant overlap were generated by the modified Super4PCS approach. The first

conflicted with the unique solution for part (a) due to penetration and was rejected. The second placement of part (b) (outlined in green) was also found compatible with the existing geometry of the input reassembly. Finally, for part (c), three positions were suggested: the first leaves a large gap between its fractured surface and the corresponding fracture on the existing reassembly, while the other two do not cause any problems. However, the third solution is also validated for match with part (a) and is accepted as the prevailing pose.

2.9.2 Measurements and quantitative evaluation

To validate our alignment method in a more concise and measurable manner, we performed experiments with non-archaeological models, which we scanned before fracturing (Figure 28). Using the scanned original as a reference model, we generated *reference reassemblies* and measured their RMS Hausdorff distance to our resulting reassemblies and the average penetration of the assembled parts.

In order to measure a meaningful penetration value, using constructive solid geometry, we generate the intersection of the two fragments, let I . The volume of I as a quantity is not very descriptive, as it depends on the actual size of the model. On the other



Figure 28: Non-cultural-heritage objects used for evaluating the algorithms against ground truth data. Top row: The original objects. These were scanned before breaking them. Middle row: The resulting fragments. These were also scanned. Bottom row: The resulting assembled objects.

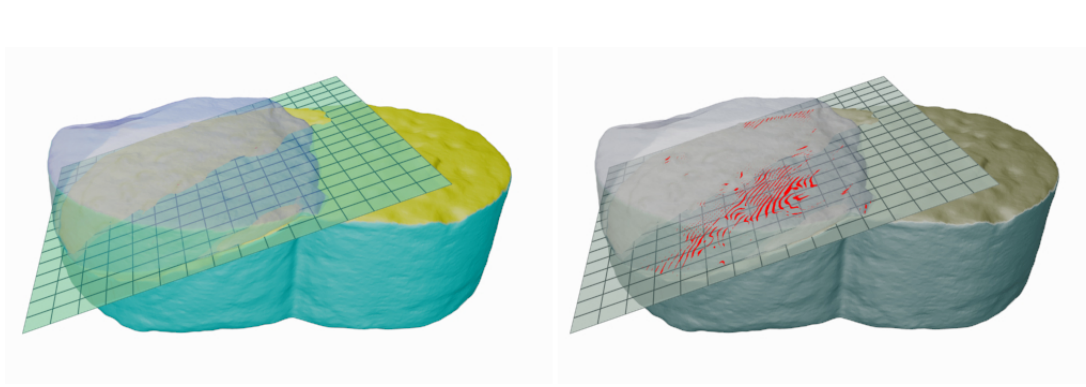


Figure 29: On the left, the projection plane on which the penetrating volume I is projected in order to measure the covered area. On the right with red the penetrating volume.

hand, the ratio of the fragment's volume to the I ratio is also misleading, as thin penetrating segments would report small values, while the actual penetration could be significant. Instead, we project I on a plane P and measure the area it covers. The final penetration score we report is the *volume to projected-area* which accounts for the mean penetration distance. P is generated by using PCA analysis of the covered fractured point set (of both fragments), after alignment (see Figure 29).

To be able to measure the performance of our methodology against ground-truth data, we generate the reference reassemblies and for these data we also present the RMS *Hausdorff distance* of our results to the reference ones. The reason we measure the RMS from the reference reassembly and not reference model is three-fold. During the fracturing procedure a part of the object is fractured into tiny pieces that cannot be digitized and would impose an error in the measurements. Furthermore, the scanning process of objects introduces error in the digitized models and thus there is a deviation between the complete model and its fragments that cannot be measured. 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 nor desirable. 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 alignment where required (symmetrical objects). In both cases, the use of Sparse ICP guarantees the convergence of the registration. Using these *registered fragments* we evaluate the results of our assembly algorithm by aligning one fragment of the reassembly with the registered equivalent, and applying the same transformation on all of the reassembled fragments. The deviation of the reassembled fragments from the registered on the reference model fragments gives us reliably the quality of the achieved assembly.

With regard to complexity, directly performing all fragment facet pair evaluations from scratch prior to driving the combinatorial multi-part stage, can take quite long. 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. In our implementation, this is addressed by caching pair results the first time a combination is evaluated, thus amortizing the cost to multiple experiments and incrementally building the cache each time a new fragment is tested for inclusion in a cluster.

Reassembly of a Single Object. Since in this set of experiments, each time fragments of a single object are used, we expect a single cluster of fragments. The capability to properly group the fragments correctly, without leaving isolated islands in the solution, is evaluated here. Figure 30 presents results from both CH and non-CH datasets, resulting in a single object.

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

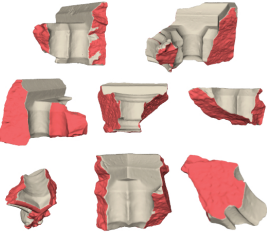

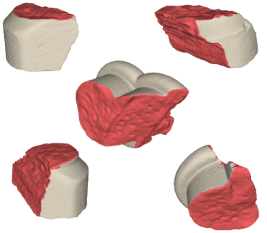
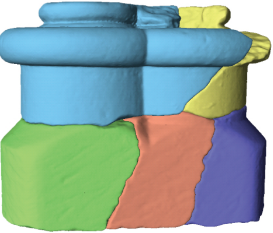
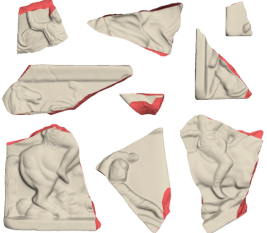
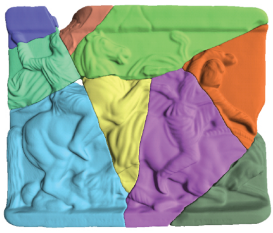
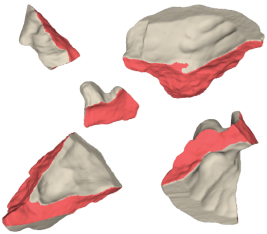
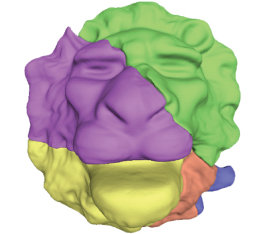
	Fragments	Reassembled Object	Evaluation
DoraEmbrasure			ESA : 420 sec ICP : 171 sec Penetration : 0.46 mm Mean Error : 1.035 Pair-Tests : 526 Hausdorff RMS : -
DoraColumnBase			ESA : 88 sec ICP : 125 sec Penetration : 0.41 mm Mean Error : 1.17 Pair-Tests : 112 Hausdorff RMS : -
Metope1			ESA : 100 sec 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 30: Single object assembly. We report the total time spend in the ESA optimization and the ICP refinement. The mean score of all pairs that drive the reassembly is also reported and the Hausdorff distance RMS from the ground truth, where available.

contrast to the former experiment the expected result is multiple clusters of reassembled objects and potentially isolated fragments.

In Figure 31 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. Here, using the same combination culling threshold E_T , 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 align using the default parameters. 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 typically inspect

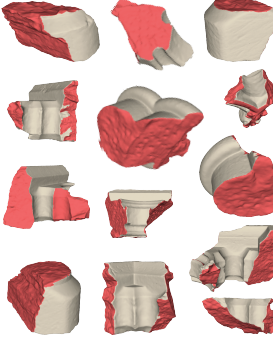
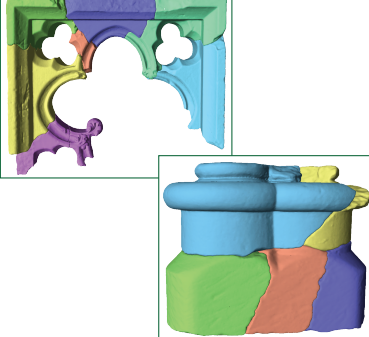
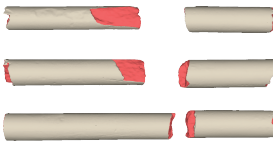
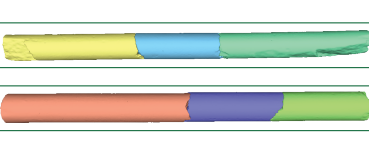
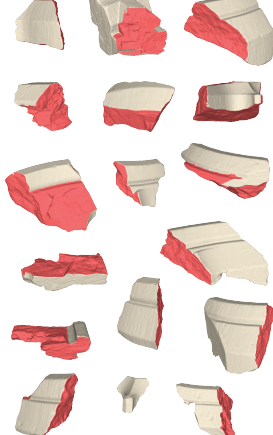
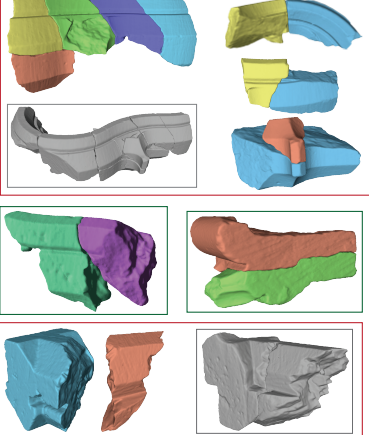
Fragments	Reassembled Object	Evaluation
		ESA : 1100 sec ICP : 320 sec Pair-Tests : 1233 E_T : 2.0 Clusters : 2/2
		ESA : 91 sec ICP : 60 sec Pair-Tests : 149 E_T : 2.0 Clusters : 2/2
		ESA : 2100 sec ICP : 525 sec Pair-Tests : 2620 E_T : 1.5 Clusters : 8/4

Figure 31: Multiple object assembly (mixed puzzle). We report the total time spend in the ESA optimization and the ICP refinement. The mean score of all pairs that drive the reassembly is also reported and the Hausdorff distance RMS from the ground truth, where available. 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).

and enforce/blacklist fragment pairs.

Reassembly of Heavily Damaged Fragments. This is the final case of our evaluation experiments for the reassembly of objects, where we experimented with objects missing large parts or with fragments that are significantly eroded. 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 frequently utilized the feature curves that extend between fragments.

In Figure 32 we present complete object reassemblies for heavily damaged objects that were obtained using pairwise alignments of both contact surface and feature 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. We also

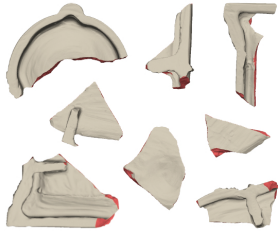
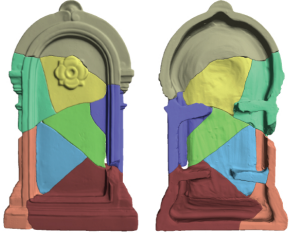
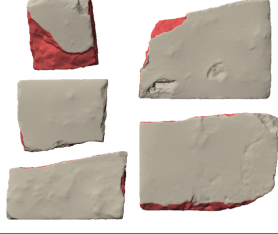
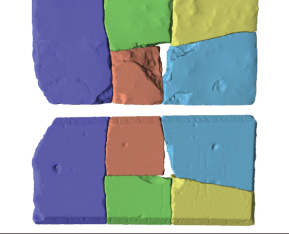

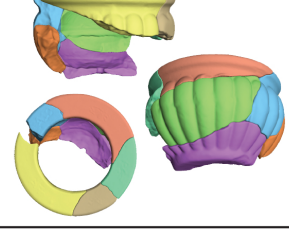
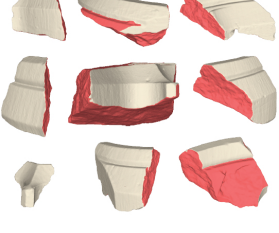
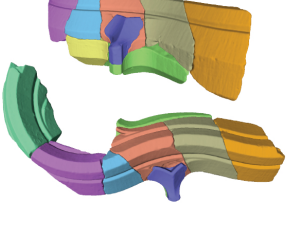
	Fragments	Reassembled Object	Evaluation
Fountain			Penetration : 0.01 mm Pair-Tests : 273 Hausdorff RMS : 1.16
Tombstone			Penetration : 0.81 mm Pair-Tests : 157 Hausdorff RMS : -
OrnatePot			Penetration : 0.46 mm Pair-Tests : 422 Hausdorff RMS : -
DoraArch			Penetration : 0.68 mm Pair-Tests : 772 Hausdorff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

Figure 32: Object assembly using eroded or missing parts. We report the total time spend in the ESA optimization and the ICP refinement. The mean score of all pairs that drive the reassembly is also reported and the Hausdorff distance RMS from the ground truth, where available.

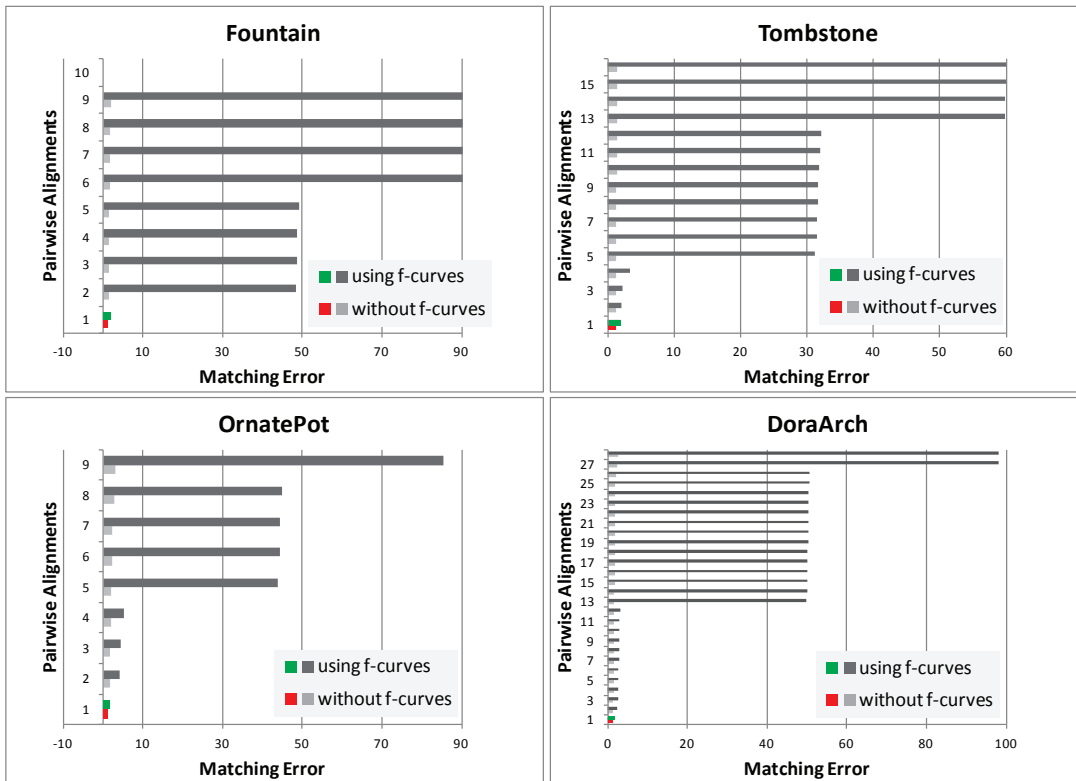


Figure 33: Matching Error discrimination for various fragment pairs with and without utilizing feature curves (tier 2 registration).

have to note here that the score of the alignment when feature curves are used, is not compatible with the score of the registration relying solely on contact surfaces.

An interesting observation on the case of puzzles with eroded parts is that the use of feature curves greatly enhances the discriminating ability of the matching error. Figure 33 confirms this as it clearly shows the difference of the measured error with and without enabling the tier 2 registration for an indicative set of fragment pairs.

3 Object Repair

We here describe the methodology for symmetry and template based-completion. Both completion approaches rely on a common preprocessing methodology and rely on Features based on heat diffusion analysis.

Related Publications

R. Gregor, I. Sipiran, G. Papaioannou, T. Schreck, A. Andreadis and P. Mavridis, [Towards Automated 3D Reconstruction of Cultural Heritage Objects](#). Proc. EUROGRAPHICS Workshop on Graphics and Cultural Heritage, 2014.

I. Sipiran, R. Gregor, and T. Schreck. [Approximate symmetry detection in partial 3d meshes](#). Computer Graphics Forum (proc. Pacific Graphics), vol. 33, no. 7, pp. 131-140, 2014.

A. Andreadis, Gregor, R., Sipiran, I., Mavridis, P., Papaioannou, G., and T. Schreck., [Fractured 3D Object Restoration and Completion](#). ACM SIGGRAPH Poster Session, 2015.

R. Gregor, D. Bauer, I. Sipiran, P. Perakis and T. Schreck, [Automatic 3D Object Fracturing for Evaluation of Partial Retrieval and Object Restoration Tasks - Benchmark and Application to 3D Cultural Heritage Data](#). Eurographics Workshop on 3D Object Retrieval 3DOR'15, 2015.

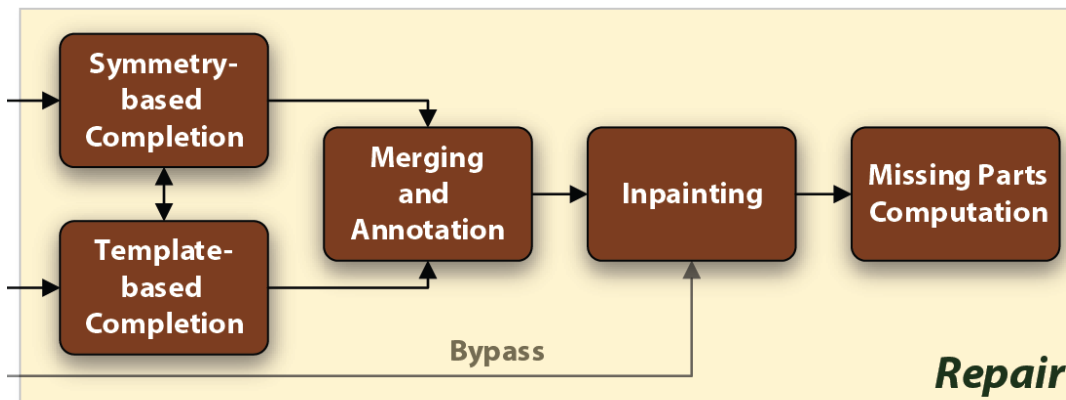


Figure 34: The Object Repair Stage of the restoration pipeline. We propose symmetry- and template-based completion for synthesizing missing geometry. For reasons of clarity, preprocessing steps specific to both completion approaches are omitted in the high level workflow overview.

3.1 Pre-processing

3.1.1 Removal of Fracture Surface

The input to the repair module is a set of reassembled fragments with surface classification information. The classification is encoded in the color channels: blue for intact surface, red for fractured surface and green for the confidence (see Fig. 35 for the classification of Dora embrasure object). Our pre-processing procedure begins with the elimination of the contact surfaces, which are not useful for the analysis of

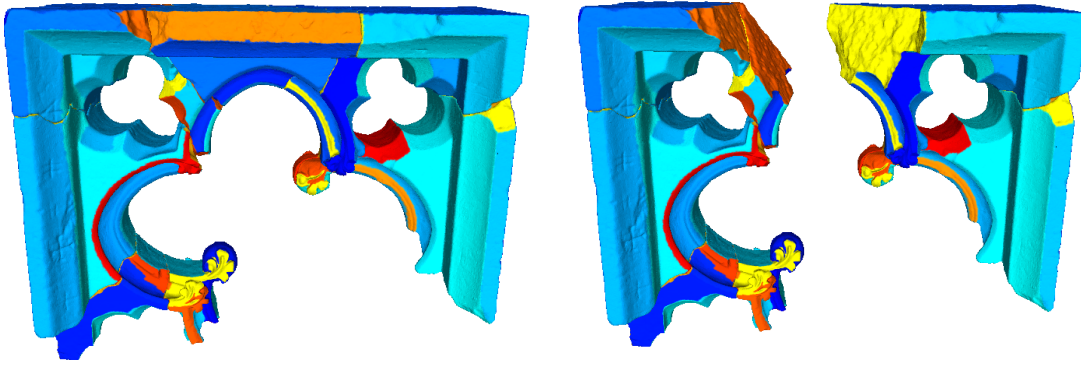


Figure 35: At left, the classification of the surface is encoded in color. Bluish colors correspond to intact surfaces, while reddish and yellowish colors correspond to fractured surfaces. At right, we remove a fragment on purpose in order to show the interior characterization of the breaking edges.

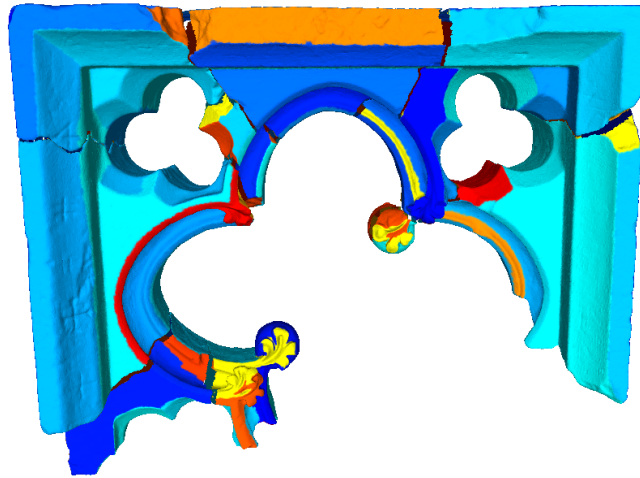


Figure 36: Result of removing fracture surfaces in input fragments.

the symmetry of the object.

The removal method proceeds in a fragment-wise manner, trying to identify the surface of a fragment that needs to be removed. Let f be the analyzed fragment and F the set of remaining fragments. We build a kd-tree with all the points from F in order to perform fast nearest neighbor queries. Subsequently, for each fracture vertex $v \in f$, we evaluate if the nearest neighbor in F is also fractured and close enough to v . If so, v is marked for removal. The closeness evaluation of v and the nearest neighbor can be controlled by a threshold λ . Finally, all the marked vertices are removed, along with the associated triangles in the mesh. The result of applying this procedure can be observed in Figure 36.

3.1.2 Poisson Reconstruction

The goal in this stage is to produce a watertight model as similar as possible to the reassembled object in terms of the overall shape. To cope with this objective, we use the Poisson reconstruction algorithm [39] which is a well known method to obtain 3D surfaces from oriented point clouds.

The reconstruction takes an input point cloud with inward normals and computes a 3D characteristic function χ , defined as 1 at points inside the model and 0 at points outside. The final reconstructed surface can be obtained by extracting an appropriate isosurface. Nevertheless, the main problem lies in the computation of the characteristic function when a set of oriented points are given. The formulation of Kazhdan et al. [39] takes advantage of the fact that the vector field of input normals is a good approximation of the gradient of the characteristic function. For this reason, the application of the divergence operator leads to a Poisson differential equation whose solution can be approximated by making use of an adequate discretization of the 3D space. The preferred discretization is an octree data structure that provides an adaptive partition of the space. Finally, a projection-based optimization step leads to a Laplacian equation that gives the solution χ . As the size of the Laplacian matrix can be prohibitively huge, an optimization was proposed in order to solve the Laplacian equation using a block Gauss-Seidel solver.

The usual parameters of this algorithm are the depth of the octree (hereafter denoted as δ) and the depth for the block Gauss-Seidel solver (hereafter denoted as ρ). The effect of δ is related to the resolution of the obtained model; that is, the higher the value of δ , the higher the resolution of the final surface. In the same way, the effect of ρ is related to the memory usage of the method; that is, the higher the value of ρ , the lower the memory usage. In all our experiments, we found that $\delta = 8$ and $\rho = 8$ provide a good trade-off between resolution and efficiency. In figure 37, we show results obtained with two different configurations of the parameters. Note that a low value of δ is prone to remove details that can be important for the further steps. In contrast, larger values of δ produce high resolution models at expenses of a high memory consumption and a considerable time computation (see Fig. 37).

After applying the Poisson reconstruction to the set of fragments, it is possible to have some topological error (non-manifold edges and vertices) and small connected components as product of the inter-penetration surface of the aligned fragments. To cope with this defects, we remove all the components with area below 0.1 of the area of the entire object. Also, we remove the non-manifold edges and vertices and restructure the local connectivity to convert the object into a watertight model.

3.2 Symmetry-based Completion

Our main assumption in the investigation of symmetry-based methods for the completion of cultural heritage objects is that symmetry is a common characteristic in

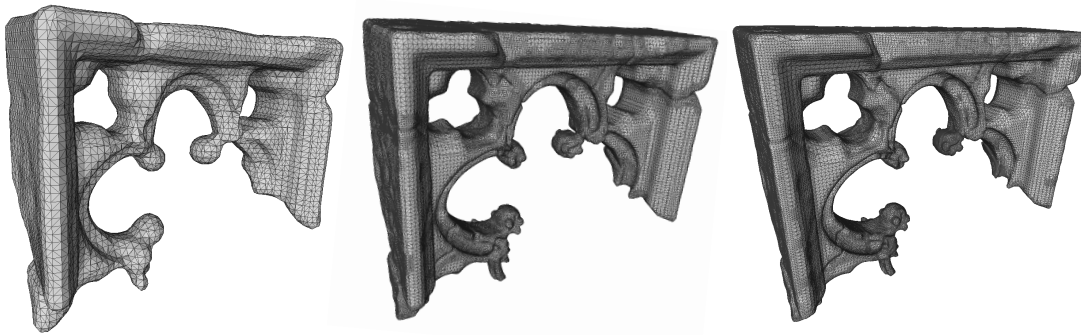


Figure 37: **Left** ($\# = 7292$, $\delta = 6$, $\rho = 6$, $time = 0.5sec.$): Many features have been removed and edges appear rounded. **Center** ($\# = 78071$, $\delta = 8$, $\rho = 8$, $time = 6.04sec.$): Features and edges are well preserved. **Right** ($\# = 116,347$, $\delta = 12$, $\rho = 8$, $time = 14.64sec.$): The shape maintains most of the details at expenses of a large number of vertices. Clearly, the shape in the middle would be the best option in term of details and processing time.

man-made objects. It can be easily verifiable if we look at the objects around us, which will probably exhibit some kind of symmetry. If our goal is to predict missing parts in archaeological objects, chances are that we can get a plausible reconstruction using the symmetry of the existing geometry. This is the motivation behind our methodology to infer (as much as possible) missing geometry using only the information of the objects itself. Although there are a number of methods proposed to analyze the symmetry of a 3D object, the challenge is in the detection of symmetries in objects with (probably large) missing parts, as encountered in the context of cultural heritage. In this section we describe our approach and discuss the main results obtained.

3.2.1 Related work

The study of symmetries has proliferated considerably in recent years in the computer graphics community, giving rise to many proposals to tackle the problem of detection. A comprehensive enumeration of these techniques is out of the scope of this report. Instead, we focus our discussion on methods related to global reflectional symmetries. For readers interested in an overall panorama on 3D symmetry, we recommend the survey by Mitra et al. [61].

A common approach is the evaluation of symmetries as a measure function. In their seminal work, Zabrodsky et al. [94] proposed a symmetry measure based on the exhaustive search in the rotational transformation space. In this direction, Kazhdan et al. [40] formulated an efficient procedure to evaluate the symmetry measure using spherical harmonic coefficients. Similarly, Podolak et al. [71] devised a Monte-Carlo algorithm to sample points in which the symmetry measure evaluation is performed. Subsequently, a refinement step can be applied to get higher precision in transformations that are local maxima. Alike, Martinet et al. [52] used spherical harmonics

to evaluate a symmetry measure based on shape moments. Also, Bermanis et al. [9] proposed to evaluate the symmetry by comparing volumes through an angular difference function which can be efficiently computed using the 3D Pseudo-polar Fourier Transform. Analogously, Kakarala et al. [32] described an optimization method that detects symmetry planes which can be interpreted as a linear phase in the spherical harmonic domain. More recently, Korman et al. [44] proved an upper bound on the sampling density of the transformation group which guarantees to find symmetries with high probability. An obvious limitation of all these techniques is the assumption that the center of mass of the object is a fixed point of the transformation space. Nevertheless, this is not the case when the object is incomplete, where the center of mass is to be shifted.

Another interesting approach to detect symmetries is the vote-based scheme as proposed by Mitra et al. [59]. The idea behind this approach is the use of self-correspondences that may convey evidence about the existence of symmetries. For instance, Mitra et al. [59] proposed to match points with high curvature. Each pair of points induces a transformation which is stored as a high-dimensional point. A final clustering step in this space can provide the more frequent transformations which derive into symmetries. Similarly, Lipman et al. [49] developed a voting algorithm that selects the best transformation as possible for random pairs of points. The best alignment is stored in a symmetric correspondence matrix and its spectral decomposition is exploited to detect symmetries. Analogously, Xu et al. [89] defined an intrinsic symmetry correspondence matrix. A multi-scale analysis over the scale of voting pairs of points is formulated in order to be aware of the scale of the symmetries. More recently, Tevs et al. [81] proposed to quantify the relation between shapes based on the regularities of symmetric parts in 3D objects. The first step is a decomposition for a shape into a set of regions. Then, a graph is used to represent the relation between the regions in terms of symmetric transformations. The authors proposed an algorithm to match two graphs which provides correspondences between the regularities of given shapes.

A few techniques have focused the attention in the problem of symmetry detection with missing data. Xu et al. [90] developed a voting algorithm to detect the intrinsic reflectional symmetry axis. Using the axis as a hint, a completion algorithm for missing geometry was shown. On the other hand, Jiang et al. [31] proposed an algorithm to find intrinsic symmetries in point clouds by using the curve skeleton. A set of filters produces a good set of symmetric correspondences which are finally verified with spectral analysis. Although these two proposals showed results for partial data, the amount of missing geometry is small. In contrast, our approach is carefully designed to deal with the existence of large missing parts, which adds complexity to the problem to solve.

3.2.2 Method overview

The core of our method is the detection of symmetries in shapes with missing geometry. Before going into the details, our intention is to provide a clear scope for our method. According to the classification provided in [61], our method is intended to detect *global*, *approximate* and *extrinsic* symmetries. It is global because the entire object is involved in the symmetry evaluation. It is approximated in the sense that the detected symmetries are not guaranteed to be the exact symmetries. And it is extrinsic because we only focus in rigid objects. Moreover, in light of these observations, we characterize the symmetries through planes in the 3D space. That is, we only consider reflectional symmetries, although the extension to other more general rigid transformations is straightforward. Therefore, our problem can be stated as the detection of reflectional symmetry planes.

The main problem is that we cannot assume that the center of mass of a given object lies in a symmetry plane due to the missing geometry. Obviously, this is the reason about why to use a vote-based approach in our context. However, the missing geometry difficulties the voting process because a point sampling based on simple local features (such as curvature, for instance) could not be robust to noise and other local perturbations.

Our method operates in two stages that try to attenuate the limitations imposed by the partiality. First, we aim at detecting a few reliable features that are effective to find symmetric correspondences and robust to missing data at the same time. We propose a new feature based on the heat diffusion on manifolds which will be proven to be effective to define local diffusion behaviors. More interestingly, pairs of matched points could give rise to potential symmetry planes.

Second, the potential symmetries require a validation in order to compute the final set of planes. We propose a vote-based algorithm that is aware of the partiality. For a potential symmetry plane, we analyze the influence of the partiality in the validation process. We develop the concept of symmetry support for a plane in order to look for evidence for this plane to be a symmetry. A graphic overview of our method can be seen in Fig. 38.

3.2.3 Detection of Symmetry Planes in Partial 3D Meshes

3.2.3.1 Multi-scale Local Features The first step of our algorithm is the detection of robust local features. In order to use these features for symmetry detection of partial data, we must bear in mind two important aspects. First, the features need to preserve certain degree of global awareness in order to enhance the matching of symmetric points. It would reduce the number of false positive matches, and therefore also the number of candidate symmetry planes. Second, the features need to characterize the local geometry in a local enough way so that it is possible to deal with the missing data. Although both requirements seem to be conflicting, we

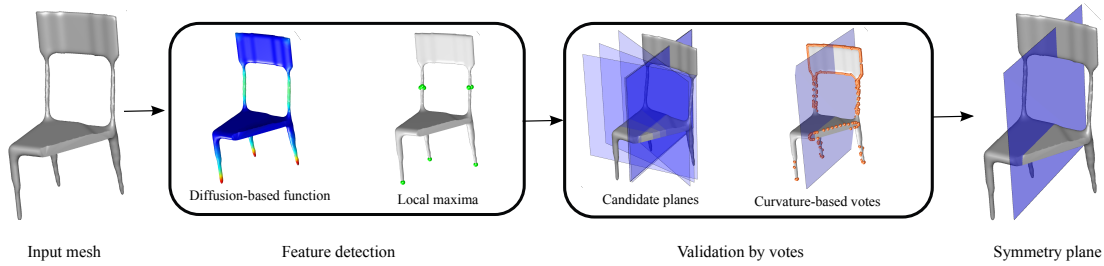


Figure 38: Our algorithm consists of two steps. First, a function is computed on the mesh such that the local maxima of this function are potential symmetric correspondences. Second, a set of candidate symmetry planes are generated and a further voting system determines the planes with the best chances to be symmetry planes. Both steps are designed to deal with the problem of missing geometry, giving a robust method as result.

believe it is possible to formulate a balanced characterization to take advantage of both aspects.

Our formulation is built on top of the theory of heat diffusion on manifolds. The heat diffusion process over a compact manifold \mathcal{M} , possibly with boundary, is governed by the heat equation

$$\Delta_{\mathcal{M}}u(x, t) = -\frac{\partial u(x, t)}{\partial t} \quad (14)$$

where $\Delta_{\mathcal{M}}$ is the Laplace-Beltrami operator of \mathcal{M} and $u(\cdot, t)$ is the heat distribution over \mathcal{M} in time t .

The fundamental solution of the heat equation is $K_t(x, y)$ called the heat kernel. This represents a solution with a point heat source in x and can be considered as the amount of heat transferred from x to y at time t supposing that the heat source is x . For compact manifolds, the heat kernel can be expressed using the eigenvalues and eigenvectors of the Laplace-Beltrami operator as follows:

$$K_t(x, y) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x) \phi_i(y) \quad (15)$$

where λ_i is the i -th eigenvalue and $\phi_i(\cdot)$ is the i -th eigenvector's entry corresponding to a given point.

It is well known that the heat kernel is isometric invariant, multi-scale, and stable against perturbations on the surface. In addition, restricting the heat kernel to the temporal domain and fixing the spatial variables, we can obtain a representation for each point on the manifold [80]:

$$h(x, t) = K_t(x, x) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x)^2 \quad (16)$$

The value $h(x, t)$ can be interpreted as the amount of heat that remains on the point x after time t . This interpretation suggests that points with high values of h for large values of t can be considered as keypoints. However, the use of high values of t is associated with a global behavior, which is not desired in our context. On the other hand, we could conversely evaluate $h(x, t)$ in low values of t , but this would make it sensitive to local perturbations. In order to give a balanced characterization and robustness, we propose a new function $\mathcal{H}(x, t) : \mathcal{M} \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ on the surface as follows

$$\mathcal{H}(x, t) = \int_0^t h(x, t) dt \quad (17)$$

The function $\mathcal{H}(x, t)$ associates the accumulation of heat up to time t to each point in the surface x . The intuition behind our formulation is that for low values of t , the accumulation in different times provides robustness to local perturbations, while still providing a global behavior. More formally, the evaluation of the integral in 17 yields

$$\begin{aligned} \mathcal{H}(x, t) &= \int_0^t h(x, t) dt \\ &= \int_0^t \left(\sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x)^2 \right) dt \\ &= \sum_{i=0}^{\infty} \int_0^t e^{-\lambda_i t} \phi_i(x)^2 dt \\ &= \sum_{i=0}^{\infty} \left(\int_0^t e^{-\lambda_i t} dt \right) \phi_i(x)^2 \\ &= \sum_{i=0}^{\infty} \left(\frac{1 - e^{-\lambda_i t}}{\lambda_i} \right) \phi_i(x)^2 \end{aligned} \quad (18)$$

The final result can be expressed in a simplified form as

$$\mathcal{H}(x, t) = \sum_{i=0}^{\infty} f(\lambda_i, t) \vec{v}_i(x)^2. \quad (19)$$

Note that Eq. 19 has been already reported by Litman and Bronstein [50] as a general formulation for spectral descriptors. In this notation, it is easy to identify the function

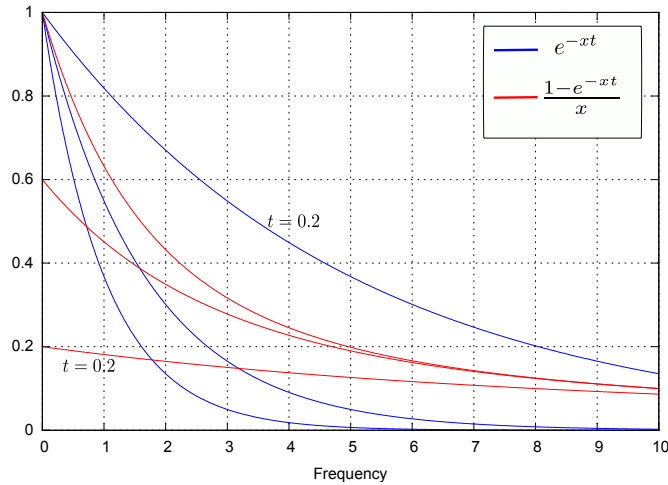


Figure 39: Kernel examples used in the computation of HKS and our feature \mathcal{H} . As t increases, the exponential decay of a low-pass filter is evident. As a result, the interval of frequencies with relevant output values is considerably reduced, which implies a more global awareness. In contrast, our kernel shows a better distribution of values in a more broad domain. We believe this represents a good trade-off between global and local information.

$f(\cdot, \cdot)$ as a filter that weights the contributions of the shape frequencies (related to the eigenvectors of the Laplace-Beltrami operator). For example the Heat Kernel Signatures [80] use a low-pass filter and the Wave Kernel Signatures [6] use a band-pass filter.

In our case, $\mathcal{H}(x, t)$ has a different behavior. The amplitude of the function f changes as t changes. In contrast to the low-pass and band-pass filters in HKS and WKS, our filter penalizes low frequencies when t is small, defining an upper bound over all possible values for f . The next proposition summarizes this point (see Appendix for a proof).

Proposition 1. *Let \mathcal{M} be a compact Riemannian manifold and $\Delta_{\mathcal{M}}$ be its Laplace-Beltrami operator. Let $0 = \lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \dots$ be the eigenvalues of $\Delta_{\mathcal{M}}$. Then*

$$f(\lambda_i, t) = \frac{1 - e^{-\lambda_i t}}{\lambda_i} < t \quad (20)$$

for any $t > 0$ and $\lambda_i \neq 0$.

The most important point to remark is that t controls the influence of the eigenfunctions in the computation of $\mathcal{H}(x, t)$ according to the magnitude of their corresponding eigenvalues. Figure 39 illustrates the behavior of our proposed filter compared to the low-pass filter used in the computation of HKS. As comparison, we can refer to the curves at $t = 0.2$, where it is notorious the attenuation of our filter in small frequencies compared to the values for a low-pass filter. In addition, our filter presents a better distribution of values in all frequencies, compensating the exponential decay in the case of a low-pass filter. This can be traduced in a more balanced behavior with

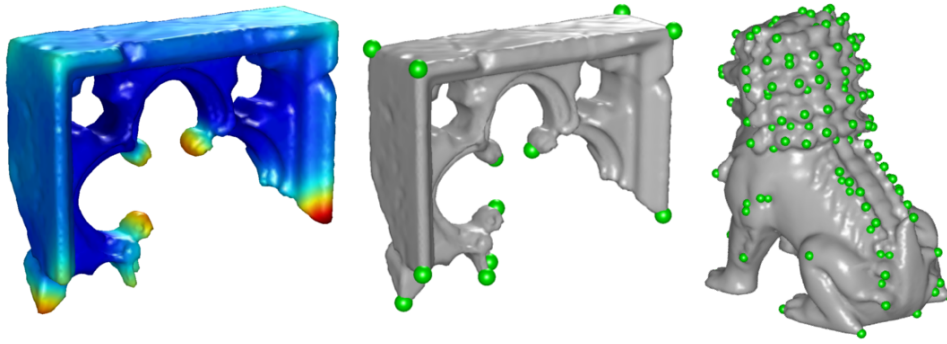


Figure 40: $\mathcal{H}(x, t)$ and its local maxima. Left: The feature values increase as the color goes from blue to yellow and to red. Center: the local maxima are potential references of symmetries, even with missing geometry. Right: local repeatable patterns could be useful in matching tasks.

a good trade-off between globality and locality. This is also the explication of why our feature is multi-scale.

Figure 40 shows the result of computing our feature. The function $\mathcal{H}(x, t)$ is useful to identify local maxima which correspond to points with a high accumulation of local heat. In our example, it is clear that these points can be used to evaluate the symmetry, even when a considerable part of the geometry could be missing. Also as an example, note how our method is able to detect repeatable local structures (dragon shape), which is a desired property to take into account in matching in general.

For our symmetry detection algorithm, once we have computed the function $\mathcal{H}(x, t)$ for every vertex in the mesh, we select the local maxima in the 2-ring neighborhood. For all our experiments, t is set to 0.01 of the surface area.

3.2.3.2 Vote-based Symmetry Detection The second step of our algorithm is the detection of symmetry planes. Let $S = \{s_1, s_2, \dots, s_n\}$ be the set of points detected in the previous stage. Points in S could be evidence of the existence of symmetries, so the matching between them is useful to generate good symmetry hypotheses. For this reason, we consider pairs of points in S as generators of our candidate planes. Let s_i and s_j be two points in S with $i \neq j$. These points define a unique plane as the set of equidistant points to both points. We define a plane with the coefficients of its general form, that is $P_{ij} = [a \ b \ c \ d]$, where the normal vector $\vec{N}_{P_{ij}} = (a, b, c) = (s_i - s_j) / \|s_i - s_j\|_2$. Furthermore, the value d can be obtained as $d = -\langle \vec{N}_{P_{ij}}, (s_i + s_j)/2 \rangle$ since the average point between s_i and s_j belongs to P_{ij} . It is worth to mention that the plane is considered as a candidate if and only if the generating points have a similar feature $\mathcal{H}(x, t)$. That is, P_{ij} is a candidate plane if $|\mathcal{H}(s_i, t) - \mathcal{H}(s_j, t)| \leq \beta$.

An important detail about the computation of the symmetry plane is the orientation

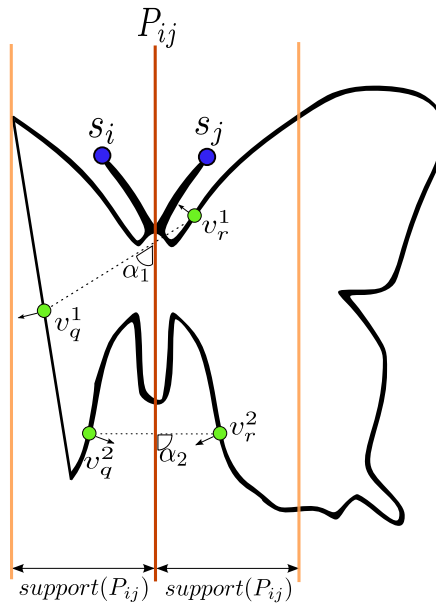


Figure 41: A pair of generating points s_i and s_j defines a bisector P_{ij} . The plane support guarantees the search of valid points for the voting scheme. In addition, voting points should preserve certain properties (as v_q^2 and v_r^2): consistent distance to P_{ij} , consistent normals and angle to bisector. Clearly, the pair formed by v_q^1 and v_r^1 do not hold these properties.

of the plane. Given a pair of points, there are two planes (with opposite directions) that represents the bisector plane. In order to deal with this, we agree to always take a pair s_i and s_j such that $i < j$.

The next step consist of accumulating more evidence for each candidate plane in order to validate whether it is a real symmetry plane or not. For this purpose, we use the curvature in each point on the surface to produce a voting system. In this point, it is necessary to use the information of the candidate plane as well as taking into account that the input shape can have missing data. More specifically, if we assume P_{ij} as a symmetry plane of a partial mesh, it is expected that the amount of geometry on either side of the plane differs considerably. For this reason, we propose to compute a support for the analyzed candidate plane. The support consist of two planes parallel and equidistant to P_{ij} such that the part with less geometry is completely contained in this support. To compute the support, it is enough to compute the distance to the parallel planes and use it as reference to the following steps. Having the plane coefficients $P_{ij} = [a \ b \ c \ d]$, the signed distance of a point $v \in \mathcal{M}$ is defined as $dist_{P_{ij}}(v) = av_x + bv_y + cv_z + d$. The plane support is finally computed as

$$support(P_{ij}) = \min(|\max_{v \in \mathcal{M}} dist_{P_{ij}}(v)|, |\min_{v \in \mathcal{M}} dist_{P_{ij}}(v)|) \quad (21)$$

The support is important to filter the points that will be used to vote for a plane. Our selection of samples to vote is based on a simple rule that uses the curvature

and the support. Given a plane P_{ij} , the samples to vote for it are the vertices v on the mesh such that $(K_1(v) + K_2(v))/2 > \delta$ and $|dist_{P_{ij}}(v)| < support(P_{ij})$, where K_1 and K_2 correspond to the maximum and minimum curvature, respectively. In all our experiments, we set δ to 0.01.

Subsequently, every pair of sample points is tested for evidence of the plane P_{ij} . Let us illustrate the ideas behind these tests before formally defining them. Figure 41 shows a 2D version of our problem. After generating the potential plane and its support, we are interested in evaluating certain characteristics of the voting pairs. For example, some interesting characteristics would be the consistent distance to the plane, the normal consistency, the angle formed by the plane normal and the segment between the voting pair, just to name a few. For example, in Figure 41, the lower pair (v_q^2, v_r^2) fulfills all the aforementioned characteristics, and therefore this pair is a good evidence that the plane could coincide with a symmetry plane. In contrast, the upper pair (v_q^1, v_r^1) is not consistent in distance, normals and angles, hence it can be rejected from the analysis for the evaluated plane.

In our algorithm, given two sample points v_q and v_r , we test the following criteria in order to ensure that the pair is a real evidence:

- **Both sides of the plane.** Points should be in different sides of the plane. That is, the pair is rejected if

$$sign(dist_{P_{ij}}(v_q)) \neq sign(dist_{P_{ij}}(v_r)).$$

- **Coherence on plane distance.** Points should be approximately at the same distance from the plane. That is, the pair is rejected if

$$||dist_{P_{ij}}(v_q)| - |dist_{P_{ij}}(v_r)|| > \theta.$$

- **Orientation to plane.** Points should be disposed in such way that they form a line orthogonal to the plane as much as possible. That is, the pair is rejected if

$$(1 - |\langle \vec{N}_{P_{ij}}, (v_q - v_r) \rangle| / \|v_q - v_r\|_2) > \alpha.$$

- **Consistency of normals.** Normals of points should also be symmetric. Here we only consider the consistency of directions. For instance, if the normal of point v_q is outward the plane P_{ij} , the normal of v_r should also be outward P_{ij} . Therefore, the pair is rejected if

$$sign(\langle \vec{N}_{P_{ij}}, \vec{N}_{v_q} \rangle) = sign(\langle \vec{N}_{P_{ij}}, \vec{N}_{v_r} \rangle).$$

- **Geometric similarity.** Points should have a similar characterization. That is, points are rejected if

$$|\mathcal{H}(v_q, t) - \mathcal{H}(v_r, t)| > \rho.$$

If a pair of samples v_q and v_r passes all test for a plane P_{ij} , the consistency of the tests can be used to compute a weight which can be accumulated as evidence of symmetry for P_{ij} ($w(P_{ij}) = 0$ before running the algorithm). Formally,

$$w(P_{ij}) = w(P_{ij}) + w_d \times w_n \times w_s \quad (22)$$

where w_d , w_n and w_s are respectively related to distance, normal, and similarity consistency, and are defined as

$$\begin{aligned} w_d &= \exp(-(|\text{dist}_{P_{ij}}(v_q)| - |\text{dist}_{P_{ij}}(v_r)|)^2) \\ w_n &= \exp((|\langle \vec{N}_{P_{ij}}, (v_q - v_r) \rangle| / \|v_q - v_r\|_2 - 1)^2) \\ w_s &= \exp(-(\mathcal{H}(v_q, t) - \mathcal{H}(v_r, t))^2). \end{aligned}$$

Once the votes have been computed, we can use them to determine the plane or a set of planes depending on the application. In our results, we specify the selection rule for each shown example. Furthermore, in all our experiments, the thresholds for the test are set to $\theta = 0.1 \times$ the diagonal of the bounding box, $\alpha = \pi/18$ and $\rho = 0.4$.

3.2.4 Reconstruction

The output of the previous step is a set of candidate symmetry planes. For the reconstruction, we need to apply the symmetry transformation associated to each plane. Since the transformation is not trivial (a reflection with an arbitrary plane), we first describe the procedure to obtain the reflection.

A candidate plane P_{ij} obtained from feature points s_i and s_j is characterized by the coefficients in general form $P_{ij} = [a \ b \ c \ d]$. The average point $t = (s_i + s_j)/2$ lies on the plane and therefore can be used to build our final transformation matrix. The steps to perform a reflection transformation with an arbitrary plane can be listed as follows

- Build a translation matrix $T(-t_x, -t_y, -t_z)$ and the associated inverse $T_{inv}(t_x, t_y, t_z)$.
- Build a rotation matrix R with the following definitions

$$\begin{aligned} - \vec{M} &= [a \ b \ c] \\ - \vec{N} &= [0 \ 0 \ 1] \\ - \vec{V} &= \frac{\vec{M} \times \vec{N}}{\|\vec{M} \times \vec{N}\|} \\ - \cos \theta &= \frac{\langle \vec{M}, \vec{N} \rangle}{\|\vec{M}\| \|\vec{N}\|}, \end{aligned}$$

- $\sin \theta = \sqrt{1 - \cos^2 \theta}$, and
- $q = 1 - \cos \theta$.

$$R = \begin{pmatrix} q\vec{V}_x\vec{V}_x + \cos \theta & q\vec{V}_x\vec{V}_y - \sin \theta\vec{V}_z & q\vec{V}_x\vec{V}_z + \sin \theta\vec{V}_y & 0 \\ q\vec{V}_x\vec{V}_y + \sin \theta\vec{V}_z & q\vec{V}_y\vec{V}_y + \cos \theta & q\vec{V}_y\vec{V}_z - \sin \theta\vec{V}_x & 0 \\ q\vec{V}_x\vec{V}_z - \sin \theta\vec{V}_y & q\vec{V}_y\vec{V}_z + \sin \theta\vec{V}_x & q\vec{V}_z\vec{V}_z + \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- The inverse rotation matrix R_{inv} is obtained by changing the sign of the $\sin \theta$ and using the same previous matrix.
- Build the reflection matrix Ref with respect to the plane XY of the 3D coordinate system.
- Build the final transformation matrix

$$T_{inv} \times R_{inv} \times Ref \times R \times T$$

The final transformation matrix can be applied over the input object to obtain the reflected version. However, as the candidate symmetry planes are only approximated, the final reflection could be not as accurate as expected, whereas it is yet a good initialization for a refinement using registration. Consequently, we perform a registration step using the Sparse ICP method [10] with a sparsity parameter $\mu = 0.4$ as proposed in the original paper.

From the visual point of view, we can evaluate our algorithm by showing the original shape and the transformed shape merged. In the first evaluation report D5.4, we used the merged shapes to evaluate the general effectiveness of our method in the PRESIOUS data. An example of such results can be observed in Figure 42.

3.2.5 Evaluation Methodology, Results and Discussion

We performed several experiments using real and synthetic data. The real data was composed of PRESIOUS data and it does not contain a ground truth, with the exception of the Fountain dataset. The synthetic data was obtained by a customized tool to fracture 3D models, and therefore it was possible to have a ground truth for measuring the effectiveness of our method.

For the evaluation, an object is composed of a set of fragments. More formally, let O be the complete object in the benchmark. Let $F = \{f^j\}$ be a set of fragments associated to O . Let $f^* \in F$ be the discarded fragment. After applying our algorithm to the set of fragments $F \setminus \{f^*\}$, we obtain a completed object C and a missing part M . Our criterion to evaluate the robustness of our algorithm with respect to the completion is the ratio of volumes

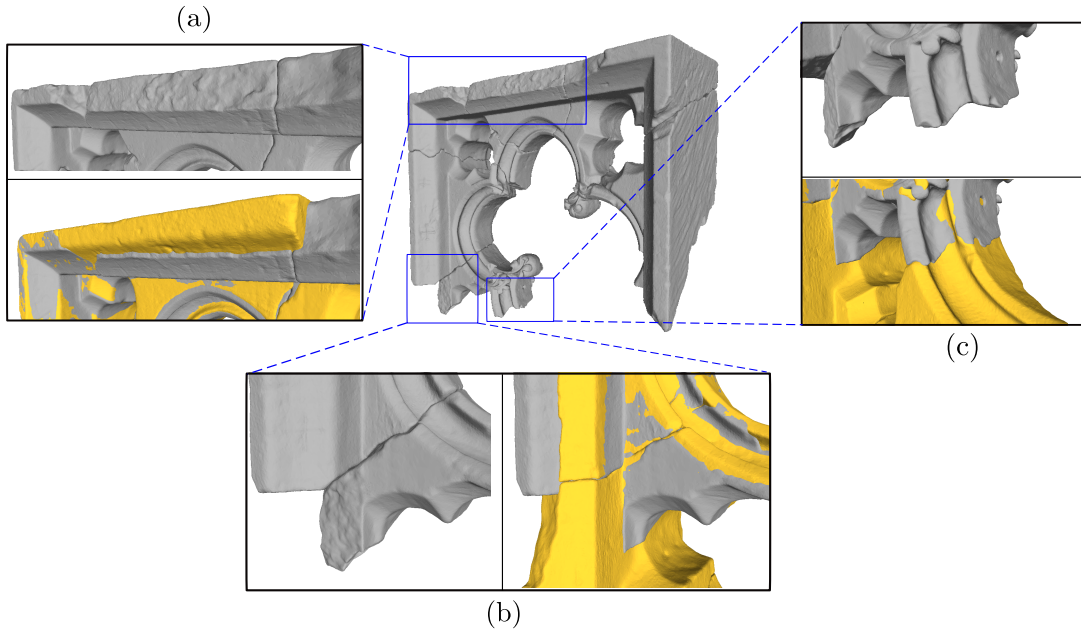


Figure 42: Results of our symmetry-based completion method in the Embrasure data. Our algorithm is able to produce missing geometry while respecting the main features of the original object, feature lines and curves.

$$E_{completion}(O) = \frac{\text{vol}(O \cap C)}{\text{vol}(O \cup C)} \quad (23)$$

It is worth to note that $E_{completion}$ is a conservative measure for the robustness of our algorithm. This is because we penalize the congruent geometry ($\text{vol}(O \cap C)$) with the divergent geometry ($\text{vol}(O \cup C)$). 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 $E_{completion} = 1$.

A summary of our results can be seen in Figure 43, where we removed one fragment at a time and executed our algorithm with the remaining ones. In all the cases our algorithm was able to find the right symmetric transformation. However, the effectiveness of the reconstruction is limited when large parts are missing or the missing part intersects the symmetric plane.

3.3 Template-based Completion

While Symmetry-based Completion might often be the most effective way of synthesizing missing parts, there are cases where the input object is not symmetric, does not provide highly distinctive local features or is too severely damaged for symmetry detection. To address this, the Template-based Completion attempts to retrieve and

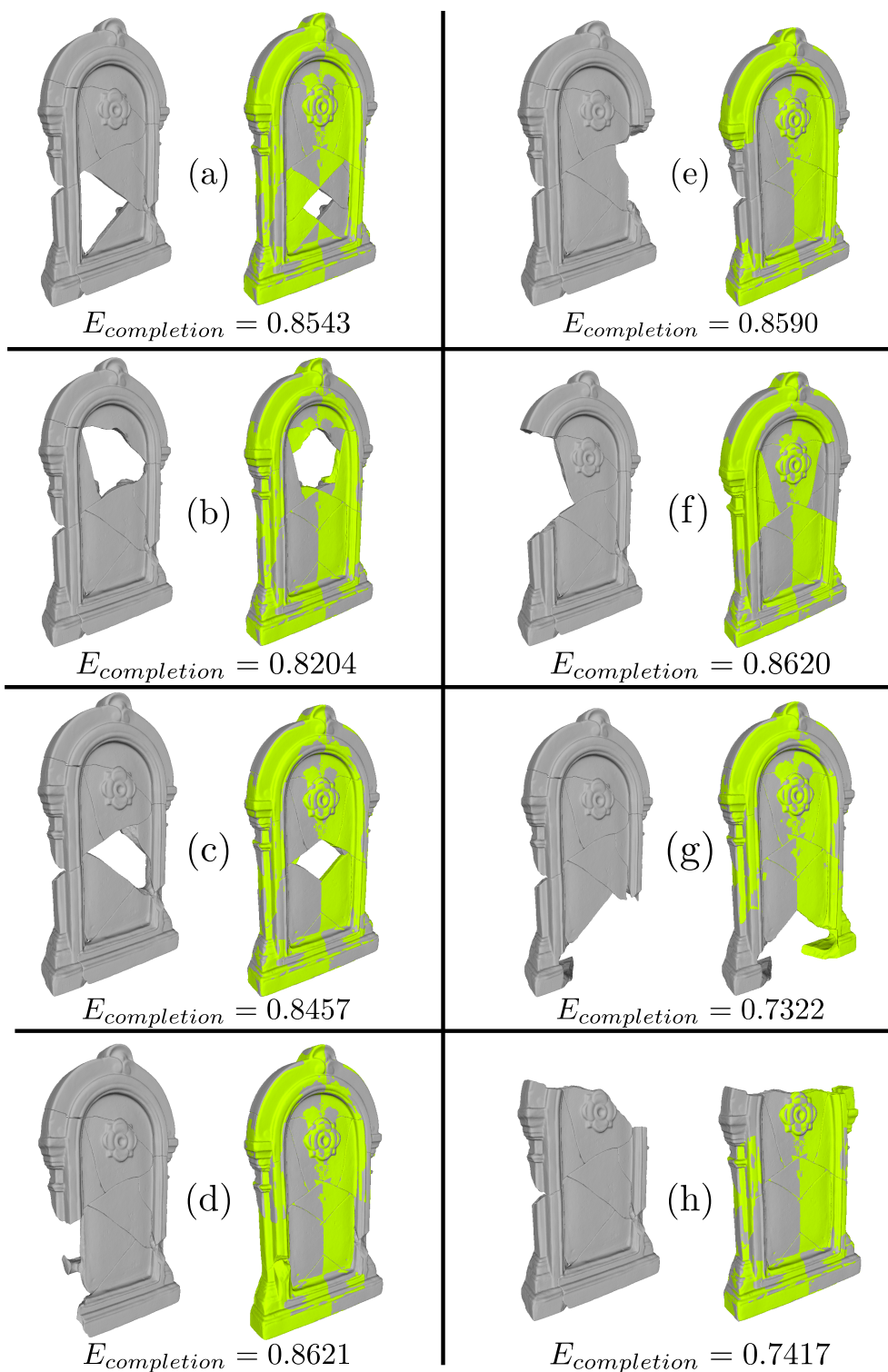


Figure 43: Results of our method and the effectiveness measure applied on different subsets of fragments in the Fountain dataset.

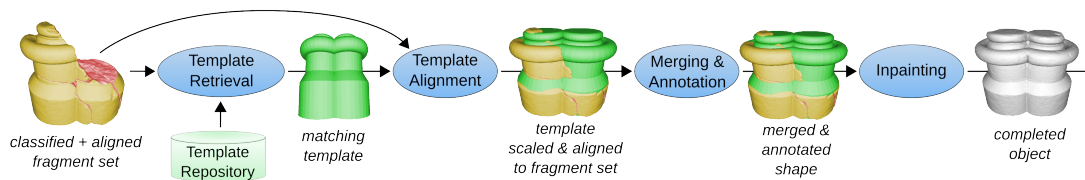


Figure 44: Schematic view of Object repair workflow when using the Template-based Completion approach.

align matching template shapes from a repository to understand the overall shape. (Fig. 44)

3.3.1 Template Shapes

It is easily observable that the effectiveness of the template-approach is highly dependent on the quantity and quality of available template data. To ease this, our approach is designed to cope with templates of varying quality. In particular template retrieval is aimed to make use of template models of different origin as described in the following subsections. For our approach, we make the assumption that an input object shares high partial similarity to such a previously digitized shape (i.e. the template) which is more complete (i.e. has less defects). In turn we assume that partial dissimilarities with respect to global shape of template and input object are very likely to reflect the defects that are encountered in our input. As small-scale degradation varies and (man-made) CH artifacts can be safely assumed to have been created manually our approach has to be insensitive to a high level of dissimilarity on more small-scale levels. While there are many partial retrieval approaches published in the community, our requirements differ⁶ due to the nature of the template shapes and their expected differences to the input shapes.

Retro-Digitized CH Object Models There is a rapidly growing number of digitized 3D CH objects models. Within sets of models from similar excavation sites, one may often observe that the discovered objects can be assorted into different categories by CH domain experts. Hence, given a reasonably large collection of digitized objects, we may assume that for many partially reassembled fragment sets, there already is a less defective digitization of another very similar object that belongs to the same object category. Still our approach has to cope with significant local geometric features between input shapes and templates even though they are (at least partially) very similar concerning their global shape and their semantics.

⁶note that this is also a significant difference to the template retrieval in WP2 where more specific assumptions have to be made either on symmetry in combination with object class or on high similarity of small-scale details, due to the more difficult initial problem statement there

High Quality Synthetic Templates In case suitable templates stemming from digitized CH Objects are not available, the template retrieval should also be able to use other sources instead. In cases where CH objects were not digitized by 3D scanning, there might have been efforts from CH domain experts or 3D modeling artists to create high quality digital 3D representations of CH objects from scratch by using e.g. 3D modeling software. The nature of such models usually does not feature as much small-scale detail information as 3D scanned digitizations of CH objects. In particular, small-scale degradation is not modeled and it can be safely assumed that besides less missing parts, they tend to have more "perfect" overall shapes (i.e. more regular and symmetric overall shapes) than real objects.

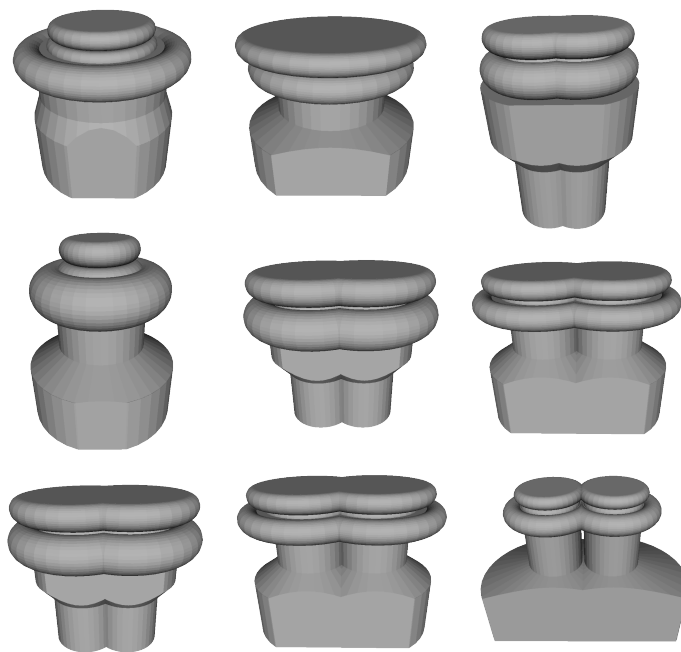


Figure 45: different random parameter configurations for a single parametric surface description of the embrasure.

Low Cost Synthetic Template Generation Both of the previously mentioned approaches to create a new template shape repository imply significant resource requirements (e.g. domain experts, scanning hardware, physical access to large collections of CH objects). In cases where those requirements cannot be met, it is possible to resort to less expensive methods. While populating the template repository with arbitrary shapes from various, freely available shape collections (i.e. not related to CH) might not yield convincing results, it might provide more acceptable results for CH objects that have a rather primitive overall shape but are too heavily damaged for symmetry detection. On the other hand and as mentioned in e.g. D4.1, large collections of models can be synthesized automatically by recombining object parts from smaller

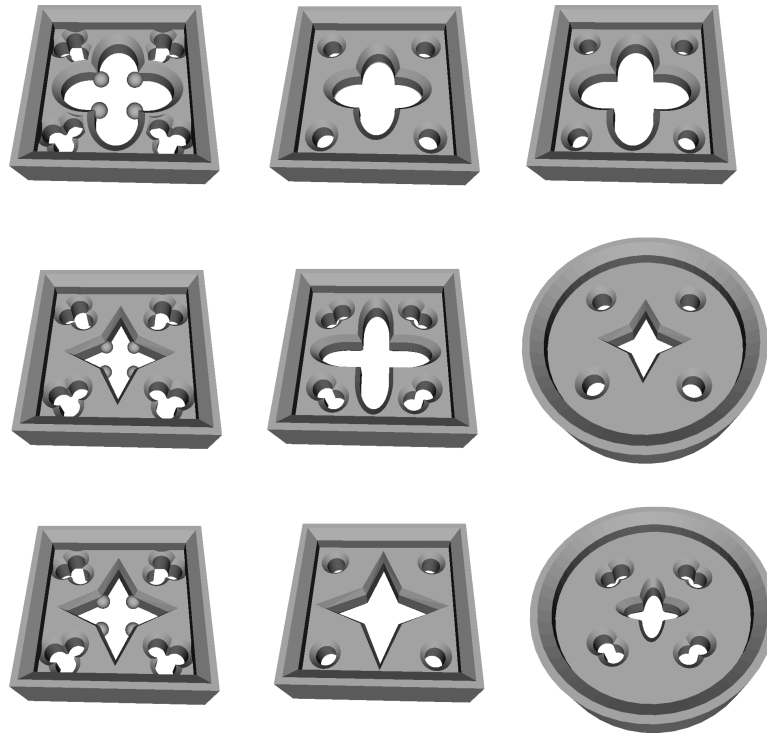


Figure 46: different random parameter configurations for a single parametric surface description of the embrasure. The top left template closely resembles the alignment of the original embrasure fragments, that were obtained in scope of WP2

existing shape repositories. A more simplistic approach to obtain a large collection of models is the use of parametric surface descriptions. As a proof of concept, we created two parametric descriptions which are based on constructive solid geometry⁷. Both parametric surfaces contain in between 10-20 numeric parameters which were assigned randomly generated values in order to obtain two sets of templates (see Fig. 45,46).

3.3.2 Local Features

As stated in Section 3.2.3.1, for compact manifolds, the heat kernel can be expressed using the eigenvalues and eigenvectors of the Laplace-Beltrami operator as follows

$$K_t(x, y) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x) \phi_i(y) \quad (24)$$

where λ_i is the i -th eigenvalue and $\phi_i(\cdot)$ is the i -th eigenvector's entry corresponding to a given point. Sun et al. [80] proposed to restrict the heat kernel to the temporal domain, which yields a point characterization that depends on the time value t . This characterization is the well-known Heat Kernel Signature (HKS)

⁷CSG, i.e. boolean recombinations of (non-uniformly) scaled primitives

$$K_t(x, x) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x)^2. \quad (25)$$

Given a shape \mathcal{S} , we need to compute the Heat Kernel Signature for every point in \mathcal{S} . In practice, the HKS of a point $s \in \mathcal{S}$ is a n -dimensional descriptor vector with each bin corresponding to some value of t :

$$p(x) = (p_1(x), p_2(x), \dots, p_n(x)) \quad (26)$$

$$p_i(x) = c(x)K_{t_i}(x, x), \quad (27)$$

where $c(x)$ must be selected in order to have $\|p(x)\|_2 = 1$. Sun et al. [80] suggested to use a logarithmic scale over the time values ($t_i = \alpha^{\tau_i}$) in the corresponding i -th bin. However, it is well known that the HKS is not invariant to local scalings and perturbations of otherwise similar objects. To overcome this problem, Bronstein and Kokkinos [?] proposed a scale-invariant version of the HKS. The first step is to provide scale invariance to the heat kernel signatures.

$$\hat{K}_t(x, x) = \frac{-\sum_{i \geq 1} \lambda_i \alpha^{\tau} \log \alpha e^{-\lambda_i \alpha^{\tau}} \phi_i(x)^2}{\sum_{i \geq 1} e^{-\lambda_i \alpha^{\tau}} \phi_i(x)^2} \quad (28)$$

that changes the scale influence into a shift in the logarithmic scale. To remove the shift, we can apply the Fourier Transform $H(\omega)$ of $\hat{K}_t(x, x)$ with respect to τ and thus take the absolute value in some frequencies as descriptor.

$$\hat{p}(x) = (|H(\omega_1)|, |H(\omega_2)|, \dots, |H(\omega_q)|). \quad (29)$$

3.3.2.1 Extensions In addition to the existing standard HKS and the SI-HKS and based on finding from extensive testing and search for good computational parameters, we introduce a modified Heat Kernel Signature, which can be regarded as an adaption of the intermediate step when computing the SI-HKS from HKS (see Equation 28, which will hereby simply be denoted as DL-HKS). Our experimental tests have shown that the Scale-Invariance of SI-HKS has several drawbacks. First, the normalization schemes and parametrization, as e.g. the time sampling used to obtain the signatures are actually sensitive to the global scale of the input objects. Second and more severe, while the Fourier-Transformation enables a vast reduction of descriptor dimensionality, it apparently discards a significant amount of potentially useful information so that SI-HKS are not very effective to determine intra-object

similarity⁸. Even though the FFT has the desirable side-effect of also introducing invariance to time shifts, the most significant problem here is that the resolution along the time domain is lost entirely. In turn, local SI-HKS features lose the multi-scale property when compared to standard HKS. This issue also translates to the partial retrieval problems when e.g. the goal is to find accurate correspondences between local features of different objects. Virtually the set of local features can hardly be separated into consistent, different classes anymore, even if the local features are actually similar in their original scale within the same object or in between two globally similarly scaled objects. From a Signal Processing perspective, discarding of the high frequency components and the fixed (i.e. data independent) sampling-time parametrization both produce noise in the computed SI-HKS that is close in magnitude to the total variance of the local descriptors within an object. While this surprisingly doesn't impair global retrieval performance when used in conjunction with a suitable encoding scheme and appropriately preprocessed shapes (as in [11]), it is a design disadvantage for the retrieval of template shapes from incomplete input objects as in our use case. Due to the missing of larger parts of the input object it is in general impossible to perform equivalent pre-scaling to corresponding templates and input objects. In other words, while SI-HKS is largely invariant to local scaling of geometric features in an object, its published way of computation requires careful global scale-normalization of the objects due to fixed parametrization of the sampling time. There is no proposal on how to derive the time parametrization in a data-driven way when such a normalization is not an option. Furthermore for our test shapes SI-HKS is not a suitable descriptor to perform inter-shape matching of local features in partially similar shapes and in our experiments, it was also not well suited to be used for reliable intra-shape matching (e.g. to find local correspondences for symmetry detection). Fig. 47 illustrates the issues of fixed timing when no global scale normalization is performed before feature extraction.

We propose a data-driven way to obtain the time parametrization by deriving them from the magnitude of the first $n_\lambda = 15$ smallest eigenvalues of a shapes discrete Laplace-Beltrami Operator approximation. Note that this also differs from the normalization scheme that was proposed to obtain invariance to (global) isometric scaling for the original HKS in [80]. However, when observing isometrically scaled objects, the magnitude of the eigenvalues relates inversely proportional to the squared scaling factor. In order to normalize the eigenvalues to obtain the Shape DNA descriptor, Reuter et al. [74] initially and among other methods proposed to fit a linear function to the sequence of the smallest eigenvalues to obtain a normalization factor. Generally, the slope of the linear function fitted to the sequence of eigenvalues is highly related to the overall scale of an object. In our early experiments for data-driven time parametrization, however, we used the sum of the first n_λ Eigenvalues λ_i as normalization factor instead, which can be assumed to be roughly proportional to the slope of such a fitting line⁹. As the shift of a Heat Kernel Signature $h_t(x)$ along the log-scaled time axis is also inversely proportional to the scaling factor of the object, we

⁸e.g. SI-HKS are less effective than standard HKS which in turn are less effective than the customized HKS proposed for the symmetry-based completion

⁹This is based on the condition that λ_{n_λ} is several magnitudes larger than λ_1

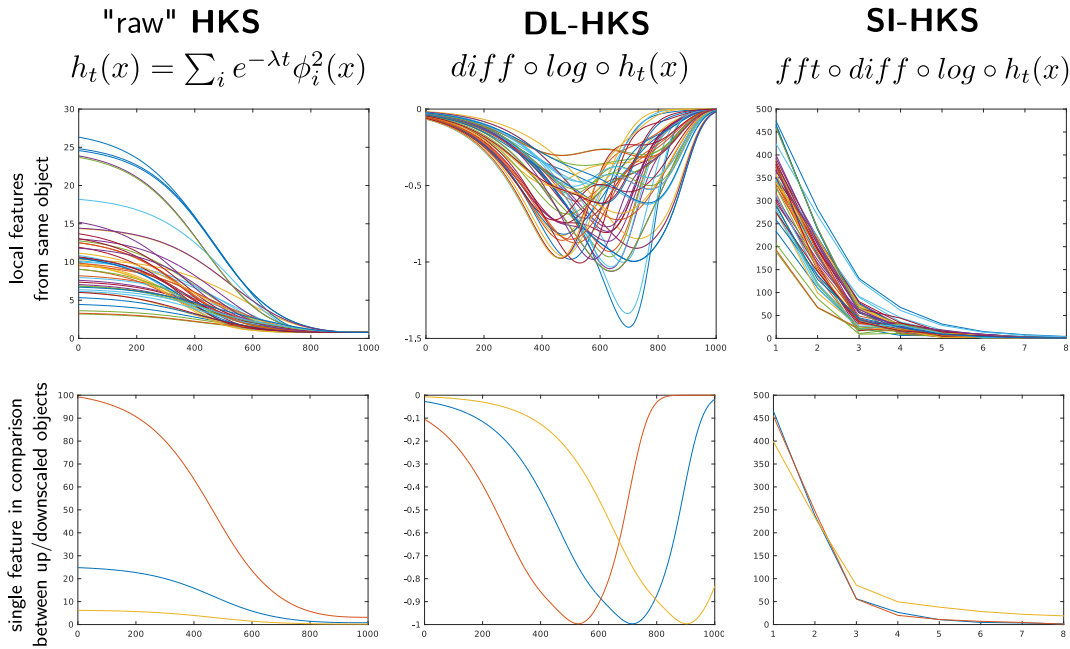


Figure 47: Comparison of HKS variants with a (random) subset of local features from the same object (top row) and a comparison of a single local feature among three isometrically scaled versions of the same model (bottom row) with fixed (logarithmic) time sampling. Scaling factors are 0.5 (red), 1.0 (blue), 2.0 (yellow). Due to the shift of the DL-HKS within the sampling time interval (center bottom), the frequency spectrum of the SI-HKS (bottom right) is not identical. Subjective similarity of the signatures is highest for the DL-HKS plot. In comparison to the original parametrization proposed in [12] the time parametrization used here already covers a more broader range here

conducted experiments where the logarithmic time sampling was obtained according to 30 and 31. For isometrically scaled objects, this eliminates the shift along the time axis for isometrically scaled objects. In our early practical experiments, the relative error between signatures of isometrically scaled objects obtained by such an adaptive time-sampling is less than 10^{-6} and could likely be further reduced by a direct computation of the slope of the eigenvalue sequence. Essentially and within a certain range determined by the numerical stability of e.g. the Laplace-Beltrami approximation and the Eigenvalue solver, pre-scaling the objects is no longer necessary to obtain DL-HKS and SI-HKS invariant to the global scale of the object. This simplifies the overall pipeline and also leads to slightly increased efficiency. However in the case of missing parts on a query, invariance to isometric scaling is not enough, we expect that relying on the entire sequence of eigenvalues is more robust in case of missing parts than a normalization based on e.g. the magnitude of a single eigenvalue¹⁰. As the effects of data-driven time parametrization in principle resembles a isometric scale normalization, it is guaranteed that the query objects has likely been normalized to a size that is not identical to the scale of the normalized (and more complete) template shape. When comparing similar local geometric features of both objects, this effect resembles local scaling of the geometric feature. Thus, a descriptor invariant to local scalings is still desirable. Additionally, when comparing our time parametrization proposal in [12, 80], our time parametrization covers a far larger range, to avoid the issue that the informative time interval of a signature is not covered by $[t_1, t_{n_t}]$.

$$t_1 = \frac{0.025 * n_\lambda}{\sum_{i=1}^{n_\lambda} \lambda_i} \quad (30)$$

$$t_{n_t} = 4000 * t_1 \quad (31)$$

Instead of performing a Fourier-Transform on the differential of the DL-HKS as for the SI-HKS, there are several alternative ways to obtain invariance to shifts along the time domain. First, instead of using the L_2 norm for distance computation, Dynamic Time Warping (DTW) could be used. However, DTW can only be computed with quadratic time complexity. In addition, DTW could reduce effectiveness as it is aimed to cope with non-uniform shifts in time when comparing two time-series. In our case, we can expect that if there are shifts in time between local features, most of the detected shifts between correspondences can be expected to be scattered around a certain value that could serve as an indication to estimate the global scaling difference between query and template. Hence we propose a simple encoding scheme to only reduce the impact of uniform, linear time shifts in our descriptor when using more efficient metrics such as L_2 . First, we simply detect the first local extrema and inflection point of the DL-HKS samples. As our experiments on DL-HKS and SI-HKS indicate, the energy in high-frequencies is very low for the DL-HKS. This implies that there is also no significant amount of high-frequency noise which could otherwise have severe

¹⁰which was also proposed in [74] to obtain invariance to isometric scaling for the Shape DNA descriptor

impact on a simple numerical search for the first local extrema on the DL-HKS and values near zeros on its second derivative. Such a search¹¹ can be performed in $O(n)$ which is lower than $O(n \log(n))$ of the FFT used to obtain the SI-HKS. Further experimentation will be required to determine to which amount the number of time samples can be reduced here to trade-off efficiency and effectiveness for this step. Once these interest points of a single signature have been detected, they are sorted according to the time domain. These points can be used to adapt the final local feature encoding so that it is possible to normalize individual signatures according to the location of interest points with respect to the time domain. Furthermore, the multi-scale property of the signature can be preserved, which allows for more fine-grained control of the impact of more local or global characteristics of a vertex. Further testing and experimentation is required to determine the optimal approach here.

3.3.3 Global Feature Encoding Scheme

The SS-BoF from [11] is extended by a weighting scheme and an additional composite distance function to exploit the previously established surface classification (section ??). The weighting scheme is applied to the computation of a second global descriptor of the query object. Identical to the first descriptor, it resembles a matrix that encodes the occurrence frequency of nearby combinations of feature vectors (i.e. *geometric words*) from the vocabulary¹². Every geometric word that is found close to¹³ a breaking edge has its weight reduced accordingly. After appropriate normalization, the resulting global descriptor is more insensitive to areas around breaking edges.

3.3.4 Quantization

Once we compute the descriptors for each shape in the database, these must be grouped in a huge collection of local descriptors which will be called the descriptor space. Next, it is necessary to quantize the n -dimensional descriptors space. The idea is to find a point set in the descriptor space in order to better cluster the whole descriptor set. Unsupervised techniques from machine learning field can be used such as k -means and its variants. In order to make this section self-contained, we briefly describe k -means clustering on the descriptor space.

Let D be the huge set of n -dimensional descriptors and k be the number of clusters we want to find. The algorithm can be summarized as follows:

1. **Initial centroids selection:** Select k points in the n -dimensional space. This step can be performed in different ways, for instance, selecting random points

¹¹based on fixed thresholds and a sliding window

¹²the vocabulary is extracted by k -means clustering of the signatures of all templates in the repository, the signatures of the query object are quantized to their nearest neighbor in the vocabulary

¹³due to the nature of the HKS, a diffusion distance is expected to lead to better results than simple euclidean distance

in the n -dimensional space, selecting random descriptors from D , or using information about the distribution of descriptors in D , just to name a few. Let $M = (m_1, \dots, m_k)$ be the set of selected centroids.

2. **Cluster assignment:** Assign each descriptor in D to the closest cluster C_i

$$C_i = \{d \in D : \|d - m_i\| \leq \|d - m_j\|, \forall j = 1 \dots k\} \quad (32)$$

3. **Centroids update:** Compute the new centroids for each cluster

$$m_i = \frac{1}{|C_i|} \sum_{d \in C_i} d \quad (33)$$

4. **Stop criterion:** If centroids remain unchanged after update step, stop and return M . Go to step 2, otherwise.

Using the set of centroids M and the heat kernel signatures previously calculated for a shape P , we need to compute a single descriptor for P , so it is necessary to combine the local descriptor in a shape descriptor. To tackle this issue, we calculate the feature distribution in a vertex $x \in P$ as $\theta(x) = (\theta_1(x), \dots, \theta_k(x))^T$ where

$$\theta_i(x) = c(x) \exp\left(\frac{-\|p(x) - m_i\|_2}{2\sigma^2}\right) \quad (34)$$

where $c(x)$ is a constant selected such that $\|\theta(x)\|_2 = 1$, $p(x)$ is the heat kernel signature of x , m_i is the centroid of cluster C_i , and σ is constant. Each bin in $\theta(x)$ can be considered as the probability that x belongs to the cluster corresponding to such bin. This is a soft version of quantization because the classic bag of features approach considers placing a one in the bin corresponding to the closest cluster and zeros in the rest. Although the classic way can be performed here, the soft version has proved to be effective in experiments.

To obtain a shape descriptor, the feature distributions are simply added to obtain a shape descriptor of size k , the vocabulary size:

$$f(S) = \sum_{x \in S} \theta(x) \quad (35)$$

and the matching between two shapes S and T is performed by using the L_1 distance

$$d(S, T) = \|f(S) - f(T)\|_1 \quad (36)$$

Nevertheless, during the quantization process, the spatial information is lost. Obviously, this information could be useful in the matching process. To address this

problem, we can consider a feature distribution among pairs of descriptors regarding a weighting factor related to spatial information. Again, we can use the heat kernel $K_t(x, y)$ as a spatial factor which will have high values for close points x and y . Therefore, the following definition for descriptors should be used:

$$F(S) = \sum_{x \in S} \sum_{y \in S} \theta(x) \theta^T(y) K_t(x, y) \quad (37)$$

This descriptor results in a $k \times k$ matrix and the distance between two shapes can be done with L_1 distance as usual. For further optimization, we add a minor extension here: Since the resulting matrix is symmetric, we can safely omit all coefficients above the principal diagonal to reduce the length of the finally encoded descriptor to $\frac{k^2+k}{2}$, which reduces the required storage space for large dictionary sizes by nearly 50 percent.

3.4 Scaling of retrieved part & alignment to input

The retrieval step delivers a template from the repository which is similar to the analyzed object. Nevertheless, the similarity is evaluated in a scale invariant way, so the scale of the template is not necessarily equal to the scale of the query. Therefore, we need to correct the scale of the template. Bear in mind that this process is not trivial because the query object could be incomplete, and a unit cube normalization strategy could not properly work. Instead we address the problem using a similarity-guided strategy. The inspiration for this method is the technique proposed by Ovsjanikov et al. [65], where a elegant relation between transformation and certain descriptor space was found.

Let Q be the query object and T be the template. The idea is to find the scale transformation E applied to T which makes the descriptors of both shapes similar. Let S be a description function, the problem can be formulated as the following optimization task

$$\underset{E}{\text{margin}} \|S(E(T)) - S(Q)\|^2 \quad (38)$$

In other words, the problem is to find a scale transformation which makes the descriptors of two shapes similar. To solve the above optimization problem, we require a dependency between the description function and the scale in order to apply a gradient method. Ovsjanikov et al. [65] proposed to use a smoothed version of the Shape Distribution descriptor [64]. Given an object Q , we compute N random distances between pairs of point in the surface $(q_{j1}, q_{j2}), j = 1, \dots, N$. Then, the descriptor $S(Q)$ is a m -dimensional vector, where the i -th coordinate is defined as

$$S_i(Q) = \frac{1}{N} \sum_{j=1}^N \exp \left(-\frac{(\mu_i - d(q_{j1}, q_{j2})/R)^2}{2\sigma^2} \right), \quad (39)$$

where R is the average distance of all pairs and μ_i is set to $\mu_i = 3i/m$. With the smooth definition of the descriptor space, the requirement to find a solution to Eq. 38 is the existence of the gradient. More formally, the solution can be found as

$$\begin{aligned} \nabla(\|S(E(T)) - S(Q)\|^2) &= 0 \\ 2(S(E(T)) - S(Q))J &= 0 \end{aligned} \quad (40)$$

where J is the Jacobian matrix, such that $J(k, i) = \frac{\partial S(E(T))_k}{\partial E_i}$, the partial derivative of the k -th coordinate in the descriptor space with respect to the i -th scale parameter. The scale $E = (E_1, E_2, E_3)$ is composed of three parameters, one per each three-dimensional coordinate. In addition, the computation of the Jacobian matrix is straightforward if we figure out that the scale E transform a point q into a point $q(E) = (E_1q_x, E_2q_y, E_3q_z)$. In our implementation we used a Cholesky solver for the non-linear optimization on J . An example of our algorithm can be seen in Figure 48.

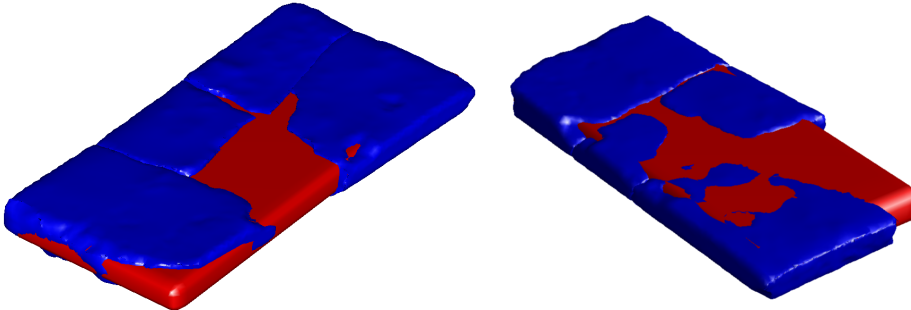


Figure 48: First results of our alignment approach: After initial pose estimation, a template shape (red) is scaled non-isometrically until optimal similarity to reassembled fragments of a tombstone (blue) is reached.

3.5 Merging & Annotation

Both, symmetry completion and The previous processing steps (i.) aligned fragments and synthesized or retrieved additional shapes. In contrast to the simplified model, in which the reassembled fragments and the mirrored parts (section 3.2) were already merged in low resolution, the aligned high-resolution fragments and generated parts are spatially aligned but still completely disconnected, separate shapes. In the merging step, all aligned parts are combined into a single watertight mesh. Merging

could be implemented by a Poisson Reconstruction (a higher octree depth should be used to preserve small scale features) or by directly computing a boolean union of the volumetric representations¹⁴. Subsequently, the resulting vertices have to be annotated by their distance to the various encountered surface types that were used for the merge (i.e. surface areas classified as defective, exterior, stemming from a template shape or having been synthesized based on symmetry).

3.6 Automated Inpainting

In this step, we aim at transferring geometry and additional information such as textures to the synthesized data. Our idea is to use self-similarity between HKS signatures to determine similar local regions which will be used as basis for the transfer. The use of heat diffusion has been successfully applied for inpainting of holes in [22]. Although our problem is a bit different due to the nature of the defects we want to recover, we believe that self-similarity is still a useful tool to this stage. The result of the inpainting step is regarded as the completed object.

3.7 Missing Parts Computation

The final step in our workflow can be used to export the predicted missing parts of the object. These could in turn be supplied to a 3D printer to produce suitable parts for a physical reconstruction of the CH object. Given the previously computed result in the workflow, missing parts can basically be obtained by computing the difference of the completed object (i.e. result of the inpainting step) and the aligned fragment set (i.e. result of the reassembly stage). To address limitations of current 3D printers, a subsequent post-processing step is required to discard parts with too small diameter.

Nevertheless, a simple boolean difference between the volumetric representations of two shapes is not suitable for the determination of accurate missing parts, since it depends on the resolution of the volumetric representation. Instead, we devise a method based on point cloud processing to get the as-accurate-as possible missing geometry.

The inputs are the original shape S and the symmetric counterpart S' . Our method is divided in two parts depending on the two types of surfaces we need in the missing part: intact surface and fractured surface. To compute the intact surface of the missing part, we proceed in the following way:

- **Filtering geometry by distance.** For each vertex v' in S' , search the nearest neighbor v in S . If $dist(v', v) < \rho$ (where ρ is enough small), then v' is marked for removal. Alternatively, if $dist(v', v) > 10 \times \rho$ (points are far enough) and

¹⁴e.g. using a readily available, efficient volumetric data structure as described in [62]

the normals of v and v' are similar, then point v' is also marked. Finally, marked vertices and their corresponding triangles are removed from S' .

- **Filtering by fracture support.** The previous filtering could break the connectivity of the input shape, so it is mandatory to check which connected components should remain. First, we identify the connected components of the resulting shape in the previous step. For each connected component, we store how many nearest neighbor points in the original shape are fracture vertices. It means we want to record how much fracture surface is covered by each connected component. We only keep connected components with more than 50% of fracture support.

Likewise, to compute the fractured surface of the missing part, we proceed in the following way:

- **Computing rough missing part.** The first step is to compute an approximation of the missing part by applying a boolean difference $B = S' - S$. It is worth mentioning that resolution is not important, as we only need a very rough approximation which will be used to find a more accurate part. Note that the boolean operation guarantees that B and S are disjoint, and therefore the only shared surface between both is the fracture surface we are looking for.
- **Extracting the fractured surface.** We perform the extraction applying a nearest neighbor approach in point clouds. For each vertex in S , we only keep the vertices with a close nearest neighbor in B . Note that this is feasible given the observation in the previous step.

Figure 49 depicts an example of missing part computed by our algorithm in the Embrasure dataset.

4 Conclusions

This report described the core reassembly and repair methodologies developed in WP4 of PRESIOUS.

First, we categorized various types of object defects often encountered in Archeology artifacts, the study of which was instructive in forming techniques to repair these kinds of defects. Also, a careful literature analysis in 3D object processing added to informing a choice of methods. We proposed a workflow that combines existing with newly developed methods, having in mind a mostly automatic process to offload the Archaeologist from manual processing steps. The methodology starts with a set of (possibly incomplete, eroded) input fragments. Fragment reassembly (see Section



Figure 49: Missing part computation. With the identification of the intact and fractured surface of the missing part, we can proceed to close the shape and produce a manifold output.

2) operates by segmentation of fracture surfaces, which are matched to connect fragments. Furthermore, based on external features, fragments which are only weakly or not connected to other fragments, may be located to plausible positions based on external object features. Object repair (see Section 3) methods aim to finish a complete object, based on a possibly incomplete reassembly result. To this end, a method based on symmetry detection using a small number key points is proposed. As an alternative, we also support completion based on retrieval of partially similar shapes from a repair repository. Our report also proposed methods to evaluate the performance of the various reassembly and repair approaches.

This methodology represents the state after 30 month of research in PRESIOUS WP 4. Methods may be refined during the last phased of the project until month 36, however the core methodology as presented will remain stable.

During our work, we also identified promising extensions which could be addressed in future work. For one, evaluation based on additional benchmarks and test cases is interesting. The methodologies developed were informed in part, by the types of objects acquired within PRESIOUS. Different types of objects, e.g., larger architectural structures, may require additional reassembly and repair methods. Besides evaluation, several extensions to individual steps of the methodology could lead to further improvements. The support for an automated segmentation step that divides the reassembled model into more primitive parts before they are processed by the Template-based Completion could reduce the required number of template shapes in case of complex non-symmetric objects. External data needs and quality of results could also be improved by support for more generic non-rigid alignment. Symmetry-based completion could benefit from support of radial or point-based symmetry. Also,

a completion based on intrinsic symmetry could be useful if performed after the template shape retrieval or directly integrated into the Inpainting step.

References

- [1] D. Aiger, N. J. Mitra, and D. Cohen-Or. 4points congruent sets for robust pairwise surface registration. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 85:1–85:10, New York, NY, USA, 2008. ACM.
- [2] E. Altantsetseg, K. Matsuyama, and K. Konno. Pairwise matching of 3D fragments using fast fourier transform. *The Visual Computer*, 30(6-8):929–938, 2014.
- [3] A. Andreadis, G. Papaioannou, and P. Mavridis. Generalized digital reassembly using geometric registration. In *Proc. IEEE/EG Digital Heritage*, 2015.
- [4] A. Andreadis, G. Papaioannou, and P. Mavridis. A parametric space approach to the computation of multi-scale geometric features. In *International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing (CGVCVIP)*, 2015.
- [5] M. Attene, M. Campen, and L. Kobbelt. Polygon mesh repairing. *ACM Computing Surveys*, 45(2):1–33, 2013.
- [6] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *IEEE Int. Conf. in Computer Vision Workshops*, pages 1626–1633, 2011.
- [7] G. Bendels, M. Guthe, and R. Klein. Free-form modelling for surface inpainting. In *Proc. of the 4th Int. Conf. on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa (Afrigraph 2006)*, pages 49–58, 2006.
- [8] G. H. Bendels, R. Schnabel, and R. Klein. Fragment-based Surface Inpainting. Technical report, 2005.
- [9] A. Bermanis, A. Averbuch, and Y. Keller. 3-D Symmetry Detection and Analysis Using the Pseudo-polar Fourier Transform. *Int. Journal of Computer Vision*, 90(2):166–182, 2010.
- [10] S. Bouaziz, A. Tagliasacchi, and M. Pauly. Sparse iterative closest point. *Computer Graphics Forum (Symposium on Geometry Processing)*, 32(5):1–11, 2013.
- [11] A. Bronstein, M. Bronstein, L. Guibas, and M. Ovsjanikov. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. on Graphics*, 30(1):1–20, 2011.
- [12] M. Bronstein and I. Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. of Conf. Computer Vision and Pattern Recognition*, pages 1704–1711, 2010.
- [13] B. J. Brown, C. Toler-Franklin, D. Nehab, M. Burns, D. Dobkin, A. Vlachopoulos, C. Dumas, S. Rusinkiewicz, and T. Weyrich. A system for high-volume acquisition and matching of fresco fragments: Reassembling theran wall paintings. *ACM Trans. Graph.*, 27(3):84:1–84:9, Aug. 2008.
- [14] A. Cayley. *The collected mathematical papers of Arthur Cayley*, volume 7. The University Press, 1894.

- [15] S. Chaudhuri, E. Kalogerakis, L. Guibas, and V. Koltun. Probabilistic reasoning for assembly-based 3D modeling. *ACM SIGGRAPH 2011*, 30(4):1, 2011.
- [16] C.-S. Chen, Y.-P. Hung, and J.-B. Cheng. RANSAC-based DARCES: A new approach to fast automatic registration of partially overlapping range images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(11), 1999.
- [17] J. Daniels II, T. Ochotta, L. K. Ha, and C. T. Silva. Spline-based feature curves from point-sampled geometry. *Vis. Comput.*, 24(6):449–462, May 2008.
- [18] S. Floery and M. Hofer. Surface fitting and registration of point clouds using approximations of the unsigned distance function. *Computer Aided Geometric Design*, 27(1):60 – 77, 2010.
- [19] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin. Modeling by example. *ACM Trans. on Graphics*, 23(3):652–663, 2004.
- [20] R. Gregor, I. Sipiran, G. Papaioannou, T. Schreck, A. Andreadis, and P. Mavridis. Towards Automated 3D Reconstruction of Cultural Heritage Objects. *Proc. EUROGRAPHICS Workshop on Graphics and Cultural Heritage*, 2014.
- [21] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
- [22] G. Harary, A. Tal, and E. Grinspun. Context-based coherent surface completion. *ACM Trans. on Graphics*, 33(1), 2014.
- [23] K. Hildebrandt, K. Polthier, and M. Wardetzky. Smooth feature lines on surface meshes. In *Proceedings of the Third Eurographics Symposium on Geometry Processing*, SGP '05, pages 85–90, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.
- [24] H. Huang, M. Gong, D. Cohen-Or, Y. Ouyang, F. Tan, and H. Zhang. Field-guided registration for feature-conforming shape composition. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2012)*, 31:171:1–171:11, 2012.
- [25] H. Huang, K. Yin, M. Gong, D. Lischinski, D. Cohen-Or, U. Ascher, and B. Chen. "mind the gap": Tele-registration for structure-driven image completion. *ACM Transactions on Graphics (Proceedings of SIGGRAPH ASIA 2013)*, 32:174:1–174:10, 2013.
- [26] H. Huang, S. Wu, D. Cohen-Or, M. Gong, H. Zhang, G. Li, and B. Chen. L1-medial skeleton of point cloud. *ACM Trans. Graph.*, 32(4):65:1–65:8, July 2013.
- [27] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. In *ACM SIGGRAPH, SIGGRAPH '06*, pages 569–578. ACM, 2006.
- [28] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. *ACM Trans. Graph.*, 25(3):569–578, July 2006.
- [29] D. F. Huber. *Automatic Three-dimensional Modeling from Reality*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2002. AAI3076866.

- [30] D. Hulin and M. Troyanov. Mean curvature and asymptotic volume of small balls. *The American Mathematical Monthly*, 110(10):947–950, 2003.
- [31] W. Jiang, K. Xu, Z.-Q. Cheng, and H. Zhang. Skeleton-based intrinsic symmetry detection on point clouds. *Graphical Models*, 75(4):177–188, July 2013.
- [32] R. Kakarala, P. Kaliamoorthi, and V. Premachandran. Three-Dimensional Bilateral Symmetry Plane Estimation in the Phase Domain. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, CVPR '13*, pages 249–256. IEEE Computer Society, 2013.
- [33] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun. A probabilistic model for component-based shape synthesis. *ACM Trans. on Graphics*, 31(4):1–11, July 2012.
- [34] E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3D mesh segmentation and labeling. *ACM Trans. on Graphics*, 29(3), 2010.
- [35] M. Kampel and R. Sablatnig. An automated pottery archival and reconstruction system. *Journal of Visualization and Computer Animation*, 14(3):111–120, 2003.
- [36] M. Kampel and R. Sablatnig. On 3D mosaicing of rotationally symmetric ceramic fragments. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 265–268. IEEE, 2004.
- [37] N. Kawai, T. Sato, and N. Yokoya. Surface completion by minimizing energy based on similarity of shape. In *IEEE Int. Conf. on Image Processing*, pages 1532–1535, 2008.
- [38] N. Kawai, T. Sato, and N. Yokoya. Efficient surface completion using principal curvature and its evaluation. In *Proc. Int. Conf. Image Processing*, pages 521–524, 2009.
- [39] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proc. of Symposium on Geometry Processing*, pages 61–70, 2006.
- [40] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Symmetry Descriptors and 3D Shape Matching. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '04*, pages 115–123, 2004.
- [41] J.-H. Kim. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, 53(11):3735–3745, 2009.
- [42] S.-K. Kim and C.-H. Kim. Finding ridges and valleys in a discrete surface using a modified MLS approximation. *Comput. Aided Des.*, 37(14):1533–1542, Dec. 2005.
- [43] D. Koller and M. Levoy. Computer-aided reconstruction and new matches in the forma urbis romae. *Bullettino Della Commissione Archeologica Comunale di Roma*, 2, 2006.
- [44] S. Korman, R. Litman, S. Avidan, and A. Bronstein. Probably Approximately Symmetric: Fast 3D Symmetry Detection with Global Guarantees. *arXiv preprint arXiv: . . .*, 2014.

- [45] J. Lee and T. Funkhouser. Sketch-based search and composition of 3D models. In *Proc. Fifth Eurographics Conf. on Sketch-Based Interfaces and Modeling*, 2008.
- [46] Q. Li, M. Zhou, and G. Geng. Pairwise matching of 3D fragments. In *Information Management, Innovation Management and Industrial Engineering (ICIII), 2012 International Conference on*, volume 3, pages 479–482. IEEE, 2012.
- [47] X. Li, Z. Yin, L. Wei, S. Wan, W. Yu, and M. Li. Symmetry and template guided completion of damaged skulls. *Computers & Graphics*, 35(4):885–893, 2011.
- [48] H. Lin, J. Gao, Y. Zhou, G. Lu, M. Ye, C. Zhang, L. Liu, and R. Yang. Semantic decomposition and reconstruction of residential scenes from lidar data. *ACM Trans. Graph.*, 32(4):66:1–66:10, 2013.
- [49] Y. Lipman, R. M. Rustamov, and T. a. Funkhouser. Biharmonic distance. *ACM Trans. on Graphics*, 29(3):1–11, June 2010.
- [50] R. Litman and A. Bronstein. Learning Spectral Descriptors for Deformable Shape Correspondence. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(1):171–180, Jan 2014.
- [51] Y. Liu, H. Yang, and W. Wang. Reconstructing b-spline curves from point clouds—a tangential flow approach using least squares minimization. In *Shape Modeling and Applications, 2005 International Conference*, pages 4–12. IEEE, 2005.
- [52] A. Martinet, C. Soler, N. Holzschuch, and F. X. Sillion. Accurate Detection of Symmetries in 3D Shapes. *ACM Trans. Graph.*, 25(2):439–464, Apr. 2006.
- [53] P. Mavridis, A. Andreadis, and G. Papaioannou. Efficient sparse icp. *Computer Aided Geometric Design (Proc. Geometric Modeling and Processing)*, 2015.
- [54] P. Mavridis, A. Andreadis, and G. Papaioannou. Fractured object reassembly via robust surface registration. In *Eurographics 2015 (Short Papers track)*. Eurographics Association, May 2015.
- [55] M. McGuire, B. Osman, M. Bukowski, and P. Hennessy. The alchemy screen-space ambient obscurance algorithm. In *Proc. of the ACM SIGGRAPH Symp. on High Performance Graphics*, HPG '11, pages 25–32. ACM, 2011.
- [56] N. Mellado, D. Aiger, and N. J. Mitra. Super 4pcs fast global pointcloud registration via smart indexing. *Computer Graphics Forum*, 33(5):205–215, 2014.
- [57] N. Mellado, P. Reuter, and C. Schlick. Semi-automatic geometry-driven reassembly of fractured archeological objects. In *Proceedings of the 11th International Conference on Virtual Reality, Archaeology and Cultural Heritage*, VAST'10, pages 33–38, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association.
- [58] M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and mathematics III*, pages 35–57. Springer, 2003.
- [59] N. Mitra, L. Guibas, and M. Pauly. Partial and approximate symmetry detection for 3D geometry. *ACM Trans. on Graphics*, 25(3):560–568, 2006.

- [60] N. J. Mitra, M. Pauly, M. Wand, and D. Ceylan. Symmetry in 3D Geometry: Extraction and Applications. *Computer Graphics Forum*, 32(6):1–23, 2012.
- [61] N. J. Mitra, M. Pauly, M. Wand, and D. Ceylan. Symmetry in 3D Geometry: Extraction and Applications. *Computer Graphics Forum*, 32(6):1–23, 2013.
- [62] K. Museth. VDB: High-resolution sparse volumes with dynamic topology. *ACM Trans. on Graphics*, 32(3), 2013.
- [63] Y. Ohtake, A. Belyaev, and H.-P. Seidel. Ridge-valley lines on meshes via implicit surface fitting. *ACM Trans. Graph.*, 23(3):609–612, Aug. 2004.
- [64] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Trans. Graph.*, 21(4):807–832, Oct. 2002.
- [65] M. Ovsjanikov, W. Li, L. Guibas, and N. N. J. Mitra. Exploration of continuous variability in collections of 3D shapes. *ACM Trans. on Graphics*, 34(4):1, 2011.
- [66] G. Palmas, N. Pietroni, P. Cignoni, and R. Scopigno. A computer-assisted constraint-based system for assembling fragmented objects. In *Proc. of Digital Heritage 2013 International Congress*, volume 1, pages 529–536. IEEE, 2013.
- [67] G. Papaioannou and E.-A. Karabassi. On the automatic assemblage of arbitrary broken solid artefacts. *Image and Vision Computing*, 21(5):401–412, 2003.
- [68] G. Papaioannou, E.-A. Karabassi, and T. Theoharis. Virtual archaeologist: Assembling the past. *Computer Graphics and Applications, IEEE*, 21(2):53–59, 2001.
- [69] D. Parikh, R. Sukthankar, T. Chen, and M. Chen. Feature-based part retrieval for interactive 3D reassembly. In *Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision, WACV '07*, pages 14–14, Washington, DC, USA, 2007. IEEE Computer Society.
- [70] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces. *Computer Graphics Forum*, 22(3):281–289, 2003.
- [71] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser. A Planar-reflective Symmetry Transform for 3D Shapes. *ACM Trans. Graph.*, 25(3):549–559, 2006.
- [72] Y. Poley and S. Peleg. Alignment and mosaicing of non-overlapping images. In *Computational Photography (ICCP), 2012 IEEE International Conference on*, pages 1–8, April 2012.
- [73] H. Pottmann, J. Wallner, Q.-X. Huang, and Y.-L. Yang. Integral invariants for robust geometry processing. *Comput. Aided Geom. Des.*, 26(1):37–60, Jan. 2009.
- [74] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-spectra as fingerprints for shape matching. *SPM '05 Proceedings of the 2005 ACM symposium on Solid and physical modeling*, 1(212):101–106, 2005.
- [75] J. D. Rodriguez, A. Perez, and J. A. Lozano. Sensitivity analysis of k-fold cross validation in prediction error estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(3):569–575, 2010.

- [76] C. Sanchez Belenguer and E. V. Vidal. Archaeological fragment characterization and 3D reconstruction based on projective gpu depth maps. In *Virtual Systems and Multimedia (VSMM), 2012 18th International Conference on*, pages 275–282. IEEE, 2012.
- [77] P. Siarry, G. Berthiau, F. Durdin, and J. Haussy. Enhanced simulated annealing for globally minimizing functions of many-continuous variables. *ACM Trans. Math. Softw.*, 23(2):209–228, June 1997.
- [78] I. Sipiran, R. Gregor, and T. Schreck. Approximate symmetry detection in partial 3d meshes. *Computer Graphics Forum (proc. Pacific Graphics)*, 33:131–140, 2014.
- [79] K. Son, E. B. Almeida, and D. B. Cooper. Axially symmetric 3D pots configuration system using axis of symmetry and break curve. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 257–264. IEEE, 2013.
- [80] J. Sun, M. Ovsjanikov, and L. Guibas. A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. *Computer Graphics Forum*, 28(5):1383–1392, 2009.
- [81] A. Tevs, Q. Huang, M. Wand, H.-P. Seidel, and L. Guibas. Relating shapes via Geometric Symmetries and Regularities. *ACM Trans. Graph.*, 33(4), 2014.
- [82] B. Thuswaldner, S. Flory, R. Kalasek, M. Hofer, Q.-X. Huang, and H. Thür. Digital anastylosis of the octagon in ephesos. *J. Comput. Cult. Herit.*, 2:1:1–1:27, July 2009.
- [83] C. Toler-Franklin, B. Brown, T. Weyrich, T. Funkhouser, and S. Rusinkiewicz. Multi-feature matching of fresco fragments. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 29(6), dec 2010.
- [84] T. Ullrich and D. W. Fellner. Generative object definition and semantic recognition. In *3DOR*, pages 1–8, 2011.
- [85] W. Wang, H. Pottmann, and Y. Liu. Fitting B-spline curves to point clouds by curvature-based squared distance minimization. *ACM Trans. Graph.*, 25(2):214–238, Apr. 2006.
- [86] L. Wei, W. Yu, M. Li, and X. Li. Skull Assembly and Completion Using Template-Based Surface Matching. *Proc. Int. Conf. on 3D Imaging, Modeling, Processing, Visualization and Transmission*, (1):413–420, May 2011.
- [87] A. Willis, X. Orriols, and D. B. Cooper. Accurately estimating sherd 3D surface geometry with application to pot reconstruction. In *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on*, volume 1, pages 5–5. IEEE, 2003.
- [88] S. Winkelbach and F. Wahl. Pairwise matching of 3D fragments using cluster trees. *International Journal of Computer Vision*, 78(1):1–13, 2008.
- [89] K. Xu, H. Zhang, W. Jiang, R. Dyer, Z. Cheng, L. Liu, and B. Chen. Multi-scale Partial Intrinsic Symmetry Detection. *ACM Trans. Graph.*, 31(6):181:1–181:11, Nov. 2012.

- [90] K. Xu, H. Zhang, A. Tagliasacchi, L. Liu, G. Li, M. Meng, and Y. Xiong. Partial intrinsic reflectional symmetry of 3D shapes. *ACM Trans. on Graphics*, 28(5):1, Dec. 2009.
- [91] S. Yoshizawa, A. Belyaev, and H.-P. Seidel. Fast and robust detection of crest lines on meshes. In *Proceedings of the 2005 ACM Symposium on Solid and Physical Modeling*, SPM '05, pages 227–232, New York, NY, USA, 2005. ACM.
- [92] S. Yoshizawa, A. Belyaev, H. Yokota, and H.-P. Seidel. Fast and faithful geometric algorithm for detecting crest lines on meshes. In *PG '07: Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 231–237, Washington, DC, USA, 2007. IEEE Computer Society.
- [93] S. Yoshizawa, A. Belyaev, H. Yokota, and H.-P. Seidel. Fast, robust, and faithful methods for detecting crest lines on meshes. *Computer Aided Geometric Design*, 25(8):545–560, 2008.
- [94] H. Zabrodsky, S. Peleg, and D. Avnir. Symmetry as a continuous feature. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(12):1154–1166, Dec 1995.