# AMERICAN UNIVERSITY OF BEIRUT

# AUTOMATIC RECONSTRUCTION OF GLASS RELICS USING MANIFOLD LEARNING

by
RABAB ALI KAWTHARANI

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Mechanical Engineering
of the Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
February 2024

AMERICAN UNIVERSITY OF BEIRUT

# AUTOMATIC RECONSTRUCTION OF GLASS RELICS USING MANIFOLD LEARNING

by

RABAB ALI KAWTHARANI

Approved by:

| | |
|---|---|
| Dr. Daniel Asmar, Professor | Advisor |
| Mechanical Engineering | D. Asmar |
| Dr. Imad Elhajj, Professor | Member of Committee |
| Electrical and Computer Engineering | |
| Dr. Samir Mustapha, Associate Professor | Member of Committee |
| Mechanical Engineering | S. Mustapha |
| Dr. Nadine Panayot, Associate Professor | Member of Committee |
| History and Archaeology, Archaeological Museum | Nadine Panayot |

Date of thesis defense: February 2, 2024

# AMERICAN UNIVERSITY OF BEIRUT

# THESIS RELEASE FORM

Student Name: Kawtharani Rabab Ali

Last First Middle

I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of my thesis; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes

___ **As of the date of submission of my thesis**

X **After 1 year from the date of submission of my thesis .**

___ **After 2 years from the date of submission of my thesis .**

___ **After 3 years from the date of submission of my thesis .**

رباب كوثراني Feb 7, 2024

Signature Date

# Acknowledgements

I wish to express my deep gratitude to my thesis advisor, Professor Daniel Asmar, whose mentorship and guidance not only motivated me to complete this research but also provided constructive feedback that significantly shaped the development of this thesis. My sincere appreciation extends to all committee members whose valuable feedback and insightful discussions enriched the quality of this work.

The journey would not have been possible without the support of the MasterCard Foundation Scholars Program at the American University of Beirut. I am truly thankful for the generous scholarship and empowering support that made this endeavor feasible.

I am also genuinely grateful to the Archaeology Museum at AUB and the Digital Cultural Heritage Center (DITCH) for granting me the invaluable opportunity to work on their relics collection alongside an experienced team that imparted enriching skills. Special recognition goes to my colleague Mohammad Karim Yassine, whose dedicated efforts were instrumental in completing this project by constructing a compatible AR platform to guide the renovation process.

Lastly, heartfelt thanks to my family for their unwavering encouragement throughout, driving me toward the achievement of this significant milestone.

# Abstract of the Thesis of

Rabab Ali Kawtharani for Master of Engineering
Major:Mechanical Engineering

Title: Automatic Reconstruction of Glass Relics Using Manifold Learning

The risk imposed by manual reassembly on valuable broken relics necessitates automating this process by leveraging computer vision for 3D data acquisition, and data science to extract features of interest from high dimensional data. This thesis proposes a solution for the automatic reassembly of broken glass relics. The solution first relies on digitizing the broken shards. After that, contours of shards are extracted and segmented. Next, the proposed system maps the segments into the space of manifolds to quantify the similarity in their local geometry and uncover pairwise matches among them. Finally, a global optimization step finds the overall solution of the reassembly problem and the shards are aligned to visualize a digitally reassembled relic. This digital solution is then used in an application that runs on a head-mounted AR device to guide users through the process of sequentially reconstructing the real relic. The focus of this thesis is on the reconstruction part of the problem. The proposed system is discussed and verified over a dataset of broken glassware. Experiments on ten manually broken glass relics validate the success of the proposed approach by estimating the correct position of each shard in the reassembled relic. Moreover, the performance of the proposed system was tested for two extreme scenarios: missing shards and intruder shards. The system showed robust performance in finding matches among shards, however, the alignment of shards around missing pieces was effected.

# Table of Contents

# Illustrations

# Tables

# Abbreviations

| | |
|---|---|
| SFM | Structure from Motion |
| LTSA | Local Tangent Space Alignment |
| AR | Augmented Reality |
| IoU | Intersection over Union |
| MST | Minimum Spanning Tree |
| ICP | Iterative Closest Point |
| NeRF | Neural Radiance Field |
| MLP | Multi-layer Perceptron |

# Chapter 1

# Introduction

The process of reconstructing fractured pottery and ceramics [1], [2], frescoes [3], [4], and manuscripts[5], [6] attracts enormous attention in the archaeology community. In attempt to preserve cultural heritage, renovators are often faced with the challenge of reassembling broken relics from a collection of broken pieces.

Traditionally, object reassembly is done manually and suffers from being difficult and slow. Furthermore, human intervention to renovate fragile fragmented artifacts can impose more damage on the pieces, making the restoration harder. For this reason, an automated digital solution in which the pose of each piece can be first tested on a digital replica sounds appealing. The enhancement in the computer vision domain, including computer-based acquisition and processing of 3D shapes, boosted computer aided reassembly in several areas, especially archaeology, creating what is now known as computer archaeology [7] [8] . Nonetheless, the difficulty of computer aided reassembly of archaeological artifacts depends on the characteristics of the fractured object; for example, the patterns on painted ceramics provide helpful indicators to uncover the matches among the pieces. Pottery's curve profile and symmetry are also helpful in matching sherds and rebuilding artifacts.

Glass, on the other hand, lacks texture and is mostly transparent; making it difficult to extract geometric features for reassembly. The risk imposed by manual reassembly on the valuable broken relics in addition to the minimal useful matching indicators on glass fragments motivated the research proposed here.

This thesis presents a solution for the automatic reassembly of broken glass relics by solving the underlying 3D puzzle that matches the relics' broken shards. The solution first relies on digitizing the broken shards and then matching them using a geometric approach built on projections in the space of manifolds. The overall objective of the project is to find the reconstruction solution and visualize it to a renovator via a head-mounted AR device (Fig. 1). The focus of this thesis is on the automatic reconstruction part of the problem.

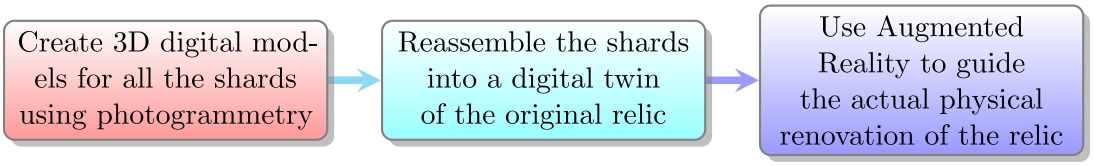



Figure 1.1: Project Overall Pipeline: (1) 3D Model Acquisition; (2) Automatic Solution for 3D Puzzle; (3) AR guided renovation.

The solution demonstrated in this thesis solves 3D puzzles with limited matching indicators. The procedure consists of several steps: 3D model acquisition to build digital twins of shards, preprocessing comprising segmenting break contours and discretizing them, pairwise matching, global optimization, and finally alignment. Structure from motion is used for the 3D model acquisition of shards. Next, the matching process is initiated, where geometric information is extracted from the edges of fragments using manifold learning, and the pair matches are estimated via a matching score that quantifies the similarity in the local geometry of break contours. Finally, a global optimization step is employed to uncover the overall solution of the 3D puzzle. Ultimately, the shards are aligned visualising poses of each one of them in the fully reassembled artifact.

The proposed solution was successfully validated on a dataset of ten manually fractured glass objects of diverse sizes and shapes. While conducting tests on actual historic fragmented artifacts posed challenges, several scenarios were simulated to emulate the contextual complexities encountered in archaeology. These simulated scenarios assessed the solution's performance under conditions such as constructing a relic with missing shards and addressing extrinsic shards erroneously sorted with the wrong group of fragments.

The contributions of this thesis include:

- A robust geometry-based matching evaluation algorithm using mapping into the manifold space,
- An automatic framework for reassembly of broken glassware,
- Experiments on 10 relics validating the proposed approach.

The remainder of this thesis is structures as follows:

Chapter One is the introduction which gives the reader an overview on the importance of cultural heritage and leveraging computer science technologies for building virtual heritage and overcoming the limitations of manual reassembly of broken artifacts. In the introduction, the reader understands the importance of finding a tailored solution for reconstructing glass relics that have limited matching indicators.

Chapter Two reviews the common approaches in solving 3D puzzles along with the existing literature on automatic reconstruction solutions. The related work section analyses various techniques for matching sherds relying on several indicators like color, symmetry, and geometry. In addition, the reader is offered a summary explaining the gaps that highlights the importance of developing a reconstruction solution for glass shards.

In Chapter Three, the methodology is explained. The chapter guides the reader through each stage of the solution from building the virtual models of the shards, to extracting matching information from them, quantifying the geometric similarity, global optimizing, and finally estimating the relative poses of shards in the overall reassembly of the glassware.

In Chapter Four, the experimental validation of the proposed algorithm is discussed. Challenges and limitations of photogrammetry on glass artifacts are explained along the attempt to overcome them. The chapter also covers the description of the dataset that was built and used for validation.

The results are discussed and interpreted in Chapter Five. A table with the output of the algorithm at each stage for every glassware in the validation dataset is presented to the readers. In addition, the performance of the algorithm in failure modes tests is analysed.

Finally, Chapter Six concludes the thesis and provides insights into future work.

# Chapter 2

# Background and Related Work

The problem of automatically reassembling "puzzles", whether jigsaw puzzles or fractured artifacts, is an important problem in the Computer Science Community [9] [6]. Though the approaches to automatic reassembly vary, the pipeline of the approaches is common among most of the work including (1) pre-processing, (2) pairwise matching, and (3) multi-fragment matching [10].

## 2.1 Solving 3D Puzzles

In this section, a brief background is presented on the main three stages of computer-based solutions for 3D puzzles.

### 2.1.1 *Pre-processing*

Pre-processing is performed on images or digital 3D models of fragments to extract features of interest that are later examined to uncover the matches between pairs. We can define pre-processing as the preparatory step to initiate the search for matches between surfaces or descriptors of interest. For instance, in their pre-processing approach, Papaioannou et al [10] segmented the surfaces of the fragments and classified them as either intact or fragmented to limit their search space to surfaces that have useful matching information. Figure 2.1 is an example on pre-processing for solving puzzles. During pre-processing, the outline contours of pieces a (green) and b (blue) are segmented and then classified into intact (purple) and fragmented (yellow). The classification step is beneficial to limit the search space for potential matches in the upcoming steps.

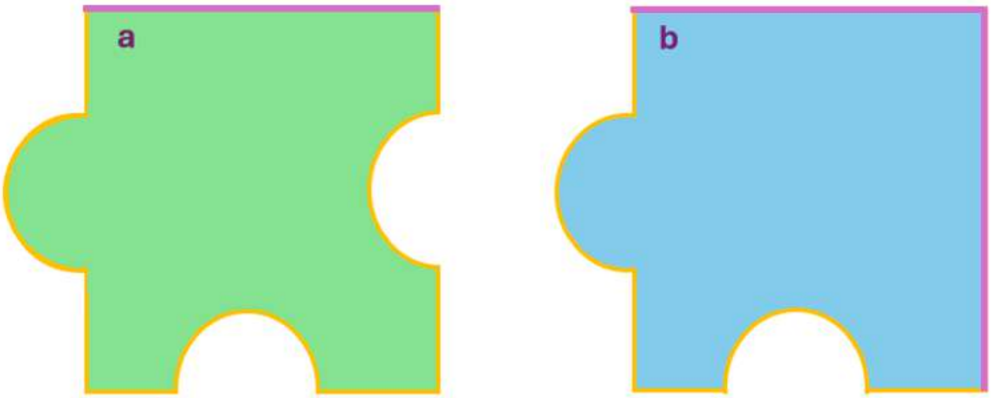



Figure 2.1: The pre-processing step, in this demonstrating example, segments the contours of jigsaw pieces (a and b) and classifies them into intact (purple) and fragmented (yellow).

### 2.1.2 *Pairwise Matching*

After the search space is determined, pairwise matching is addressed by searching for a possible match for each fragmented surface. Pairwise matching approaches fall under four main categories: (1) jigsaw matching, (2) two-dimensional matching, (3) restricted three-dimensional(2.5-D) matching, and (4) three-dimensional matching. Jigsaw matching depends on matching similar patterns in the fragmented pieces like the colors around the break zones. Two-dimensional matching is a solution for thin fragments, such as frescoes, where planar approximation of curves as polygons is an accepted simplification. Restricted matching is extremely helpful because indicators, such as color or symmetry, can be leveraged to restrict the search for matches. Three-dimensional matching considers curves and surfaces in space, but many of the 3D approaches fail to fully automate and require manual initial alignment or region of interest selection before matching starts [11]. Figure 2.2 demonstrates the pairwise matching step where similar contours between pieces a and b are detected (dark blue dashed line).

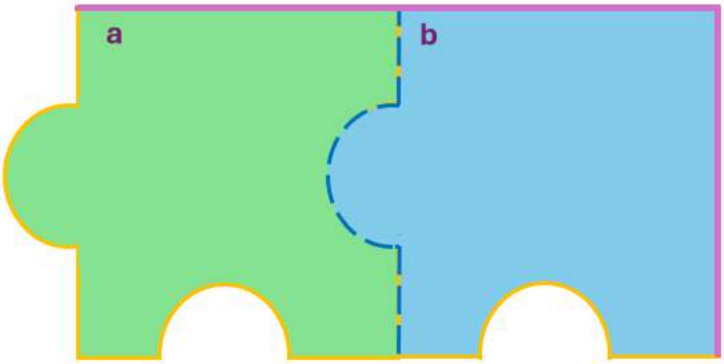



Figure 2.2: The pairwise matching step, in this demonstrating example, detects potential matches (dark blue dashed line) between pairs having similar contours (between pieces a and b).

### 2.1.3 *Multi-fragment Matching*

After matching pairs are detected, global optimization is needed to find the overall solution, a process known as multi-fragment matching. Some methods were developed for specific applications; for example, Goldberg et al. [12] proposed a global automatic solution for jigsaw based on a predefined set of conditions. Another application-specific solution was presented by Zhu et al. [13] for restoring ripped documents. However, in the majority of other cases, a greedy algorithm is utilized to optimize the global reconstruction using a matching metric obtained in the pairwise matching step. In the multi-fragment matching step, all pairwise matches detected in the previous step are optimized into one solution as shown in Figure 2.3.

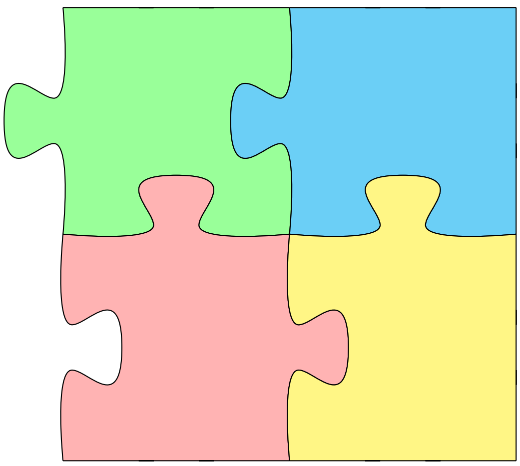


Figure 2.3: The multi-fragment step in this demonstrating example optimizes the matches detected in the pairwise matching step to find the final solution of the puzzle.

## 2.2 Related Work

Solving 3D digital puzzles of archaeological fragments was addressed in the literature for different artifacts and based on various approaches. The indicators of a possible match between a pair of fragments, referred to as descriptors, can vary from color, profile, edge geometry, etc. In this section, we will discuss several approaches of reconstruction, classified based on the descriptors adopted to retain the matches among fragments.

### 2.2.1 *Color*

Color continuity serves as a cue in artifacts reconstruction. It is mainly used in reconstructing paintings [14], painted ceramic pottery [15], and frescoes [3]. In this section, we discuss various approaches that incorporate break area color information, exclusively or among other metrics, to reveal the final reconstructed solution.

Exploiting color continuity, Tsamoura et al. [14] proposed a framework to reconstruct fragmented artwork and images. In the first step of the proposed methodology, famous content-based image retrieval techniques were applied to determine a list of potential spatially adjacent pieces for every image fragment, thus limiting the search space and reducing the computational cost of the following steps. Next, color similarity matrices were calculated and color similarity criteria were employed to retain matching contour segments. For the alignment of pair matches, several ICP variants were examined and evaluated. Finally, the overall reassembly of the image was achieved by utilizing alignment angles exploited from the poses that were estimated by ICP in the previous step.

In another approach, Oxholm and Nishino [15] incorporated color values around contours with torsion and curvature to reconstruct shattered pottery. They represented sherd contours in a two-dimensional form and employed the iterative uniform sampling method from Leitao and Stolfi [16]. The incorporation of stored color values aimed to enhance the precision of aligning the shattered segments. The paper proposes a multi-channel 2D representation of the fragments boundaries encoding shape and color that is followed by 2D partial image alignment techniques to identify matching segments.

Fresco fragments carry useful feature descriptors based on traditional color information and surface normal characteristics that encodes physical characteristics such as brush strokes, string impressions, and erosion [3]. In their multi-feature approach, Toler-Franklin et al. [3] incorporated color and shape information with normal maps as a set of easy to acquire feature descriptors to uncover matches among fragments of historic fresco. Concerning the color data, they started by calculating mean color and color variance that are traditional features used in image-based matching techniques. Additionally, they observed that a pair of adjacent sherds display similar level of pigment decay, thus they utilized color saturation.

Although color information and continuity is a helping cue that makes painted artifact reconstruction easier, it is not always applicable. Glass artifacts usually lack texture, pattern, and painting, which makes incorporating color indicators irrelevant in uncovering matches among shards.

### 2.2.2 *Profile and Symmetry*

In this section, we delve into methodologies that leverage prior knowledge of axially symmetric artifact profiles to achieve the 3D reconstruction of their fragmented shards. Figure 2.4 is an example of an axially symmetric shape that revolves around its axis of symmetry (orange line).

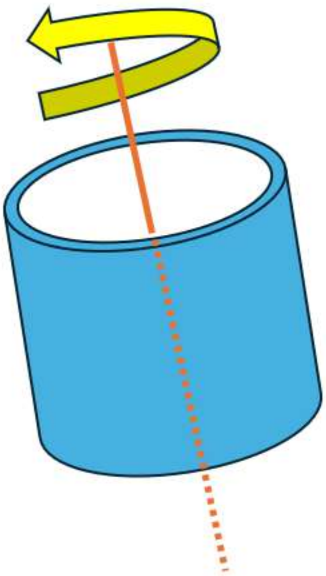


Figure 2.4: A symmetric vessel whose axis of symmetry and profile can be utilized for digital reassembly.

Willis et al. [17] proposed a solution constrained by axial symmetry. Their approach follows a Bayesian framework and employs a Markov Chain Monte Carlo (MCMC) sampling technique for the reconstruction of 3D shapes with axial symmetry from fragmented input data. Prior knowledge of axial symmetry is incorporated in a probabilistic framework which helps in handling uncertinties like missing or noisy data. The global alignment was achieved using a greedy algorithm.

By relying on Pottmann's method and a novel optimization approach, Zhou et al. [18] estimated the rotation axis of a symmetric surface then calculated the entire profile for pottery. The inherent challenges posed by missing shards and holes were addressed by assembling neighboring sherds with validations through human-computer interactions.

Although these approaches can be considered as a valuable contribution to the 3D reconstruction field, the assumption of axial symmetry is not applicable in all cases of glass, such as a jug for example, where the handle of the artifact is only located at one side of the jug.

### 2.2.3 *Edge and Broken Surface Geometry*

The broken surfaces and edges of shards carry important information for reconstruction. In the following, we explore several approaches that utilise the broken edge, also referred to as broken ribbon.

To segment the surfaces of archaeological pieces based on the deviation of polygon normal, region growing was used in the work of Papaioannou et al [19]. Not only was segmentation performed, but the bumpiness of the surfaces was also estimated to eliminate intact surfaces from the search space and keep the broken ribbon. Combination of curve matching and surface matching techniques were used to retrieve the relative position transformation and calculate the respective matching score for pairs of three-dimensional fractured objects. Global optimization via a greedy algorithm resulted in the final assemblage of the fragments [19]. This work is restricted to fragments of thick surfaces with bumps due to fracture, such as stone bricks, which is not the case in glass shards which have thin walls with sharp and slanted edges.

In another work, Li et al. [8] proposed an approach for pairwise matching of fragments by introducing a *"curvedness"* metric, defined for patches of radius $r$ over a neighborhood $N$, describing how smooth or rough a surface curve is. Each patch is embedded into a 3D histogram of curvedness values. From the histograms, matching pairs are obtained and used for alignment via a hierarchical greedy algorithm that chooses the best correspondence at each iteration and avoids backtracking. The curvedness concept presented in the work of Li et al. [8] requires the fragmented surfaces to be bumpy, which is not compatible with the breaking behavior of glass which usually has sharp break facets.

The edge geometry match was translated into a similarity distance metric in the work of Kong and Kimia [20]. For two dimensional matching, the contours were approximated as polygons and aligned to find an optimal 2-dimensional Euclidean transformation between pairs. The similarity score was defined as the residual distance between points. At each epoch of the multi-fragment assemblage, the first best matched pieces with the highest similarity score are removed from the database of potential matches. Even though the matched pieces were removed from the search space, the algorithm allows backtracking in case of failure and reconsiders the excluded pieces. This method was verified experimentally on flat fragments from 2D puzzles and ceramic tiles where the ridges can be projected into polygons in 2D. However, this technique is not applicable for concave glass relics whose ridges are 3D curves.

The thin broken surface of glass shards that is usually sharp and slanted, carries minimal information of roughness. Thus, exploiting the geometry along fragmented glass ribbons is not always useful to explore potential matches due to similarity in bumpiness.

### 2.2.4 *Summary*

It is noteworthy that the reassembly of glass relics is not specifically addressed in the state of the art. The methods available for automatic reassembly fail to address the special characteristics of glass shards such as: (1) thin walls, (2) sharp and slanted break edges with little or no bumpiness, and (3) little texture and recognizable patterns and color information around the break zones, which highlights the need for an automatic reassembly solution dedicated to broken *glass* relics.

# Chapter 3

# Methodology

In this chapter, the methodology to reconstruct broken glass relics is discussed. Figure 3.1 presents a flowchart of the framework starting by building models of shards through 3D model acquisition, extracting contours, segmenting them, quantifying geometric similarities in the pairwise matching stage, optimizing the overall solution, and finally aligning shards and visualizing the reassembled relics.

## 3.1 3D Model Acquisition

A virtual 3D model of each shard is constructed using Structure from Motion (SfM) by extracting depth information from a series of overlapping offset 2D images captured for a feature of interest (Fig. 3.2). In SfM, there is no need to predefine the geometry of the scene, nor the position and orientation of the camera. An iterative method of features extraction and correspondence from overlapping images is performed to solve for these targets [21]. Features like unique texture, patterns, and other distinctive points are detected for each image. Next, these features are matched among multiple images and related via a matrix known as the Fundamental matrix $F$ which represents the epipolar geometry in an algebraic form. After that, the Essential matrix $E$ is calculated from $F$ and the camera calibration matrix (intrinsic matrix) $K$. $E$ is then used to estimate the camera poses in terms of rotation and translation. The obtained information (camera poses and correspondences) are used to estimate the 3D coordinates of the reconstructed scene via triangulation.

There are various variants and extensions of SfM. Global SfM [22] aims to globally optimize the estimated camera poses and the generated 3D structure by considering the entire image set at once, unlike traditional SfM which follows a sequential approach. On the other hand, Incremental SfM [23] updates the 3D model at each iteration by adding one image making it convenient for real-time applications. Moreover, Dense SfM [24] builds dense point cloud representation of the scene with detailed 3D information. Concerning the extensions, Multi-View Stereo (MVS) [25] is an SfM extension that reconstructs camera poses, sparse 3D data, in addition to dense points through estimating depth maps for each image in the input sequence.Thus, MVS adds more density and geometry details to the output of SfM.

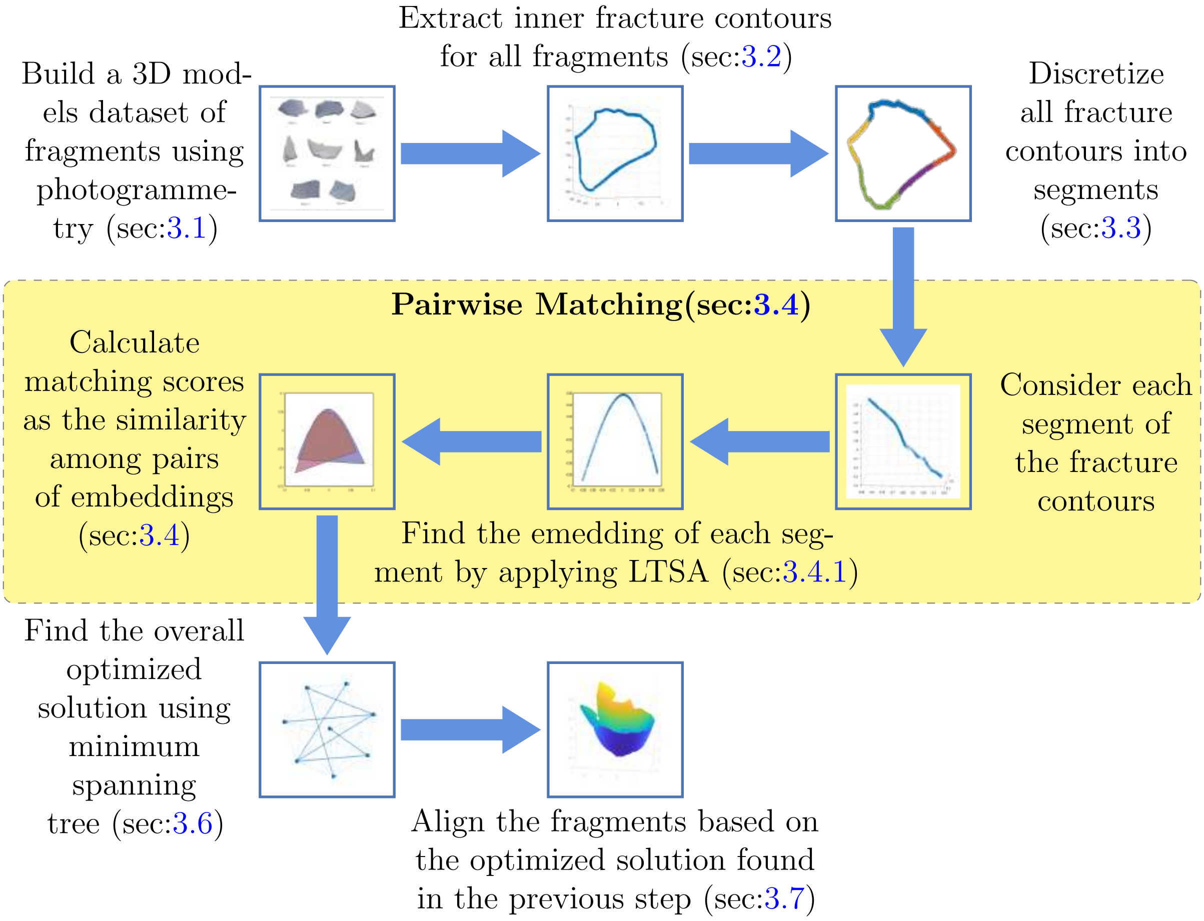



Figure 3.1: Flowchart of our method framework. The principle steps of the method are: (1) Build the database of fragments 3D models; (2) Extract the inner fracture contours; (3) Discretize contours into segments; (4)Pairwise matching: find segments embeddings and quantify similarity among pairs; (5) Multi-fragment matching using minimum spanning tree; (6) Final reassembly; (7) AR for visually guided renovation.

## 3.2 Segmentation

Preprocessing the virtual 3D models of the shards is important to extract geometric information of their underlying break curves. The contour points of each piece are segmented based on curvature, which is calculated as follows:

$$k = \sqrt{\frac{(z"y' - y"z')^2 + (x"z' - z"x')^2 + (y"x' - x"y')^2}{(x'^2 + y'^2 + z'^2)^{\frac{2}{3}}}}, \tag{3.1}$$

where $k$ is the curvature value at a point of coordinates $(x, y, z)$, $(x', y', z')$ are the first derivatives of the coordinates, and $(x", y", z")$ are the second derivatives. Points are classified as cusps or non-cusps based on the curvature at that point. Points with maximum curvature are labeled as cusps and prompted to the user for validation.

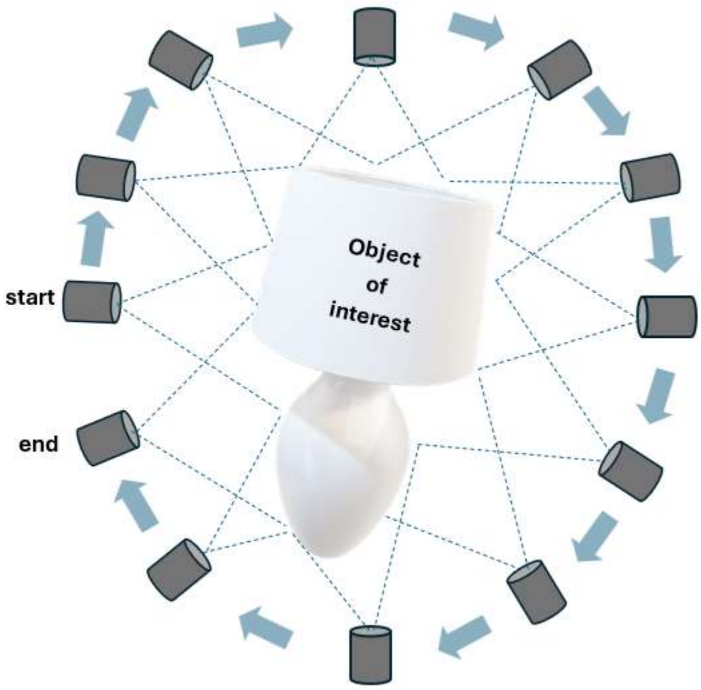



Figure 3.2: Structure from Motion Setup: capture overlapping images of the object of interest and repeat from different camera angles.

## 3.3 Discretization

Along its length, each contour can match to different pieces, thereby justifying the need to discretize the contours into several segments of length ($l_{min}$) which is an initial value that is subject to change to find the actual matching length of each pair as discussed later in Section 3.5 The number of segments ($i$) obtained from discretizing a contour is defined as:

$$i = \frac{\text{geodesic length of the contour}}{l_{min}}, \tag{3.2}$$

where the geodesic length is the distance along the contour from start point to end point. It is estimated in an iterative manner by estimating the Euclidean distance between two consecutive points and summing the value over all the points of the contour as shown in ( 3.3) and visualized in Fig. 3.3 for a contour of $n$-points:

$$i = \sum_{i=1}^{n} \sqrt{(x_i - x_{i+1})^2 (y_i - y_{i+1})^2 + (z_i - z_{i+1})^2}, \tag{3.3}$$

where $(x_i, y_i, z_i)$ and $(x_{i+1}, y_{i+1}, z_{i+1})$ are the coordinates of 2 consecutive points on the contour $(P_i)$ and $(P_{i+1})$ and $n$ is the number of points of the contour.

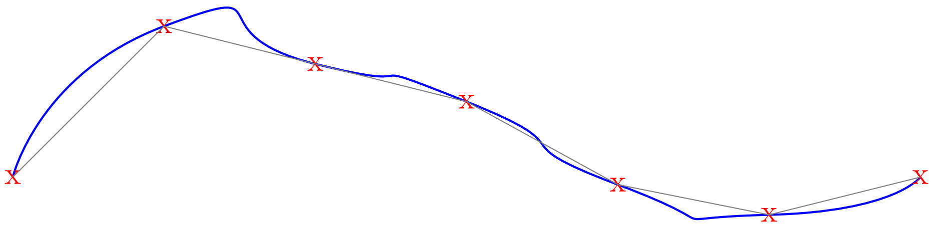


Figure 3.3: Geodesic Distance Approximation Using Euclidean Distance: the blue curve represents the contour, the red x's are points that belong to the curve, the gray lines represent the Euclidean distance between two consecutive points on the curve.

### 3.3.1 *Anchor Point*

Given that different choices for the starting point for the discretization results in a different output, an iterative process is needed in which different starting points are chosen according to this heuristic. First, the starting point ($P_0$) is chosen as the first point in the array of the 3D coordinates of the contour. In the next iteration, a new starting point ($P_1$) is considered such that the distance between the two points is: $d(P_1, P_0) = \mathcal{T}$, where $\mathcal{T}$ is a hyper-parameter.

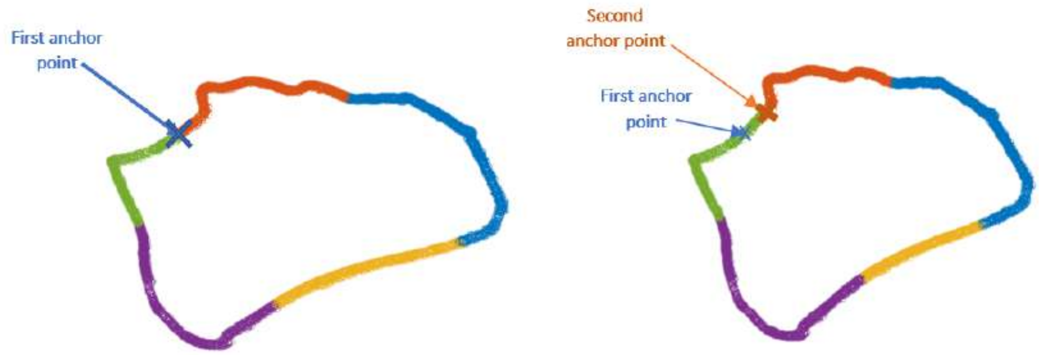



Figure 3.4: Changing the anchor point at each iteration to obtain different discretized segment.

## 3.4 Pairwise Matching

In this section, the main contribution of the thesis is discussed. In pairwise matching, a search for similarity is edge geometry is launched where similarities are quantified via a matching score. The extracted contours are 3D data points, with curvature being the main feature of interest for estimating potential matches; thereby satisfying the manifold hypothesis in data science. The manifold hypothesis states that high dimensional data lie on a low dimensional manifold carrying sufficient information about the features of interest [26]. For the contours, the lower dimension must carry geometric information about its curvature value over its geodesic distance. In order to find the manifold embedding of the contours, geometric dimensionality reduction techniques that fall under manifold learning are used.

### 3.4.1 *Local Tangent Space Alignment*

Three-dimensional contour segments are embedded in a 2D manifold space. The dimensionality reduction reveals the geometric features of the data using a manifold learning technique known as local tangent space alignment (LTSA) [27], which is a manifold learning technique used to extract geometric features from data for computer vision applications [28]. A tangent space is a space that embodies all possible tangents at a point on a manifold (Fig. 3.5) and carries information about the curvature of the manifold as this point. For example, $\mathcal{P}$ is the tangent space at the point $A$ comprising all tangents (of which $\vec{u}$ and $\vec{v}$) to the manifold $\mathcal{M}$ at $A$ .

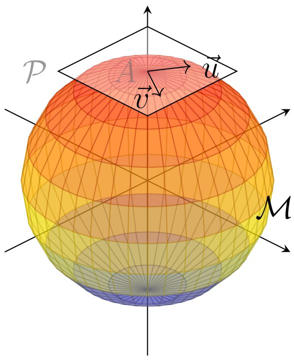



Figure 3.5: Tangent space $\mathcal{P}$ at point $A$ to manifold $\mathcal{M}$.

Local tangent spaces for each point in the dataset are aligned in an optimization step that aims to minimize the difference in local pairwise orientations in the high dimension. The solution to the optimization problem is done through Eigenvalue decomposition and considering top Eigenvectors. A matrix that maps data from higher to lower dimensions is formed from the Eigenvectors.

### 3.4.2 *Matching Score: Intersection Over Union*

All the discretized contours are mapped into the 2D geometric domain using LTSA. Segments with similar geometry possess similar embeddings (Fig.3.6), and to detect this similarity, each segment of a shard's contour is examined versus all other segments of the other shards. Examination is evaluated quantitatively using Intersection over Union (IoU). IoU, also known as Jaccard index, is an extensively used metric in computer vision for evaluating similarities between two arbitrary shapes [29]. In this thesis, IoU is used as a matching score to evaluate the similarity between two contour segments. The obtained 2D representation of each contour is considered as a polygon. IoU is calculated for all the examinations as shown in (3.4), where $Area_1$ and $Area_2$ are the areas of the polygons created by the embedding of two segments A and B respectively.

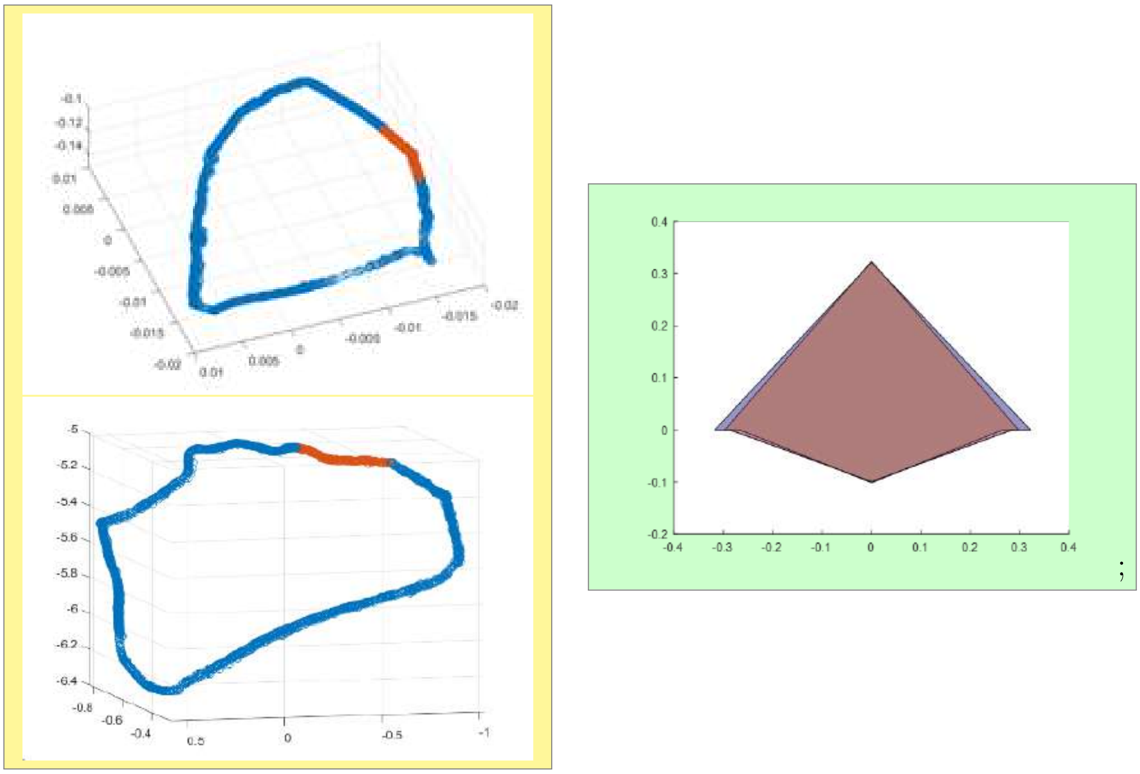


Figure 3.6: The mapping of matching contours from 3D space (orange segment on the left) to 2D manifold space (right) results in similar embeddings.

$$IoU = \frac{Area_1 \cap Area_2}{Area_1 \cup Area_2} \tag{3.4}$$

The matching score is high when a pair is matching (*i.e.*, has similar geometry); and low otherwise.

Figure 3.7 presents the variation in IoU between matching and non-matching segments. To the left, the embeddings are very similar, leading to a high IoU. On the other hand to the right, non-matching segments are mapped to different embeddings leading to a low IoU. Now that IoU is calculated over the search space, scores greater than a tolerance ($\tau$) are considered as possible matches. The discretization step performed earlier for a specific length $l_{min}$ imposes a search for the actual matching contour length.

## 3.5 Search for Length of Matching Contour

For the considered possible matches, the length of matching contour $l_i$ is altered in an iterative manner, such that the actual matching length of the pairs $L_{match}$ is determined. Each iteration $l_i$ increases such that $l_i > l_{i-1}$, thus $l_i$ approaches $L_{match}$ which increases the value of $IoU_i$. Once the segment length approaches the matching length, that is $l_i \approx L_{match}$, the score $IoU_i = IoU_{max}$. When the segment length exceeds the matching length ( $l_i > L_{match}$), $IoU_i$ starts decreasing. The goal is to


(a) Similar embedding for matching segments.

(b) Different embedding for non-matching segments.

Figure 3.7: To the left: the similarity in the embeddings of matching contours results in high IoU. To the right: the difference in the embeddings of non-matching contours results in low IoU.

find $l_i$ for which $IoU_i = IoU_{max}$ and consider it as the best estimation of $L_{match}$. The iterative search process for matching contour lengths comprises two steps. During the initial phase, the potential matching pairs are incrementally extended by five points at one end, and the matching score is calculated. Once the score stops increasing, the second is initiated. In the second phase, the two segments are extended by five points at the opposite end in each iteration until the maximum matching score is achieved. The search algorithm is demonstrated in the flowchart presented in Fig.3.8.

The pairwise matching results in a list of possible matching pairs among the list of the shards. Each match is evaluated with a matching score obtained from the 2D embedding and defined by the matching segments geodesic distances in the 3D space. The output of the pairwise matching acts as the input to the multi-fragments matching for optimal assemblage.

## 3.6 Multi-fragment Matching

Multi-fragment matching is considered as a global optimization step. Among the obtained matching scores, the best matches are found for the overall reassembly. This is a combinatorial optimization problem whose aim is to increase (or decrease) an objective fuction related to a domain of discrete choices. In graph theory, one method to solve this problem is the Minimum Spanning Tree (MST) greedy algorithm [30]. A graph, also known as a tree, whose nodes are the shards and edges are the $non\text{-}matchingscores = 1 - matchingscores$, is built from the results of the previous step. Figure 3.9 is an example of a tree where the nodes represent the shards, and the edges are the non-matching weights. For example, the non-matching weight between Piece 1 and Piece 2 is low, indicating a high possibility of a match. However, the non-matching score between Piece 1 and Piece 3 is high, indicating a low possibility of compatibility between the two shards.

The algorithm results in finding all the matches in the search space while avoiding

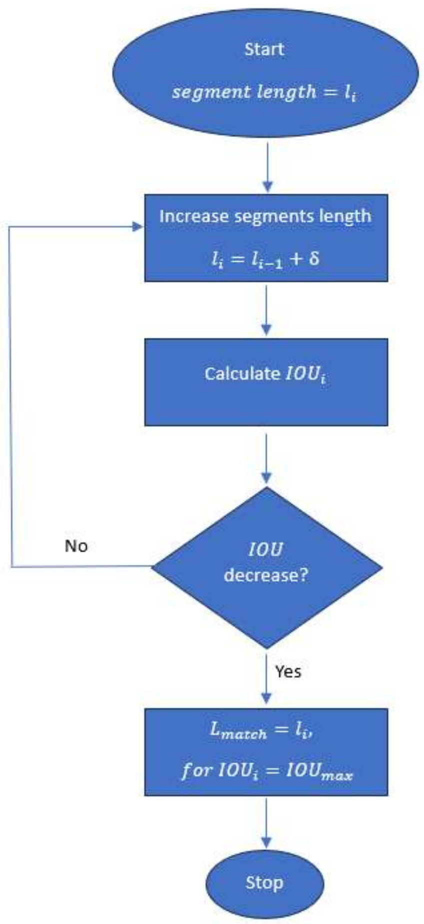



Figure 3.8: The flowchart of the algorithm of searching for the matching length of segments pairs based on the value of their respective matching score.

cycles, which can be formed from three connected pieces. For example, Piece 1 is connected to Piece 2 and to Piece 5, Piece 2 is connected to Piece 1 and Piece 5, and obviously Piece 5 is connected to Piece 1 and Piece 2. Although this type of connection is neglected by the greedy algorithm, it does not affect the result because the algorithm will detect two out of the three matches. For example, the algorithm highlights the match between Piece 1 and Piece 2 and the match between Piece 1 and Piece 5, but not the match between Piece 2 and Piece 5. However, this missing match will be imposed during the alignment and reassembly stage. The position of Piece 5 is constrained by the positions of Piece 1 and Piece 2, thus, the connection between Piece 2 and Piece 5 will eventually appear by default. Figure 3.10 shows the output of the greedy algorithm performed on the graph shown in figure 3.9. The highlighted edges represent the matches selected by the optimization algorithm. Note that the algorithm chooses the lowest non-matching scores while avoiding any cycles.

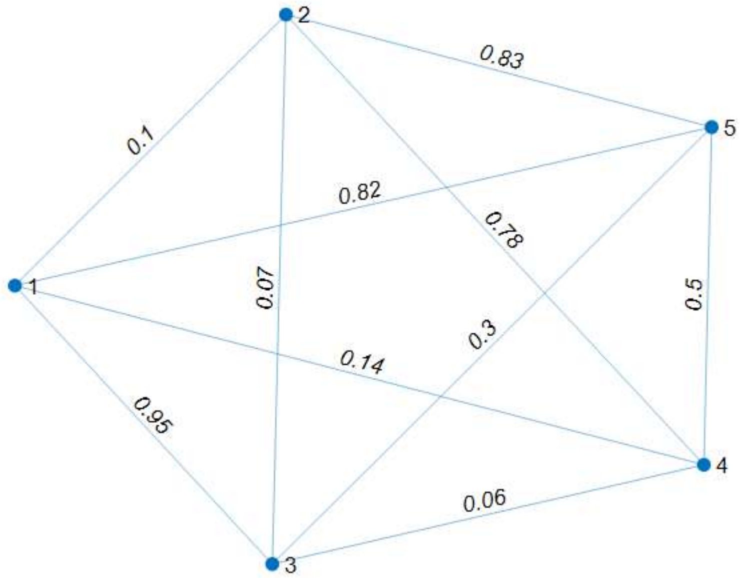



Figure 3.9: Minimum Spanning Tree with each node representing one shard of the fragmented piece and each edge representing the non-matching score between the shards at the nodes.

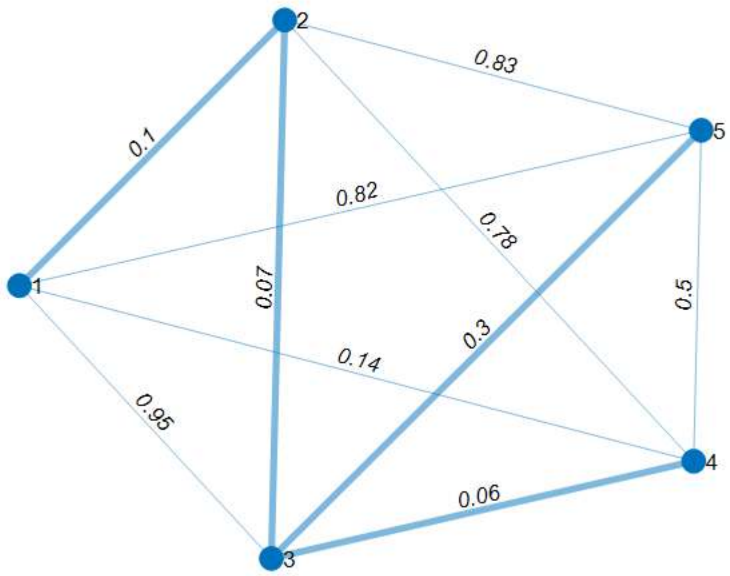



Figure 3.10: The optimized solution obtained by the greedy algorithm. the highlighted edges are the matches that maximizes the overall matching score without allowing overlap in the shards placement.

## 3.7 Alignment

Once the matches among the shards are discovered, the proposed system utilizes this information to visualize the reassembled digital twin. The virtual reassembly is completed using alignment to find the relative poses of the shards using *Iterative Closest Point* (**ICP**) [31].

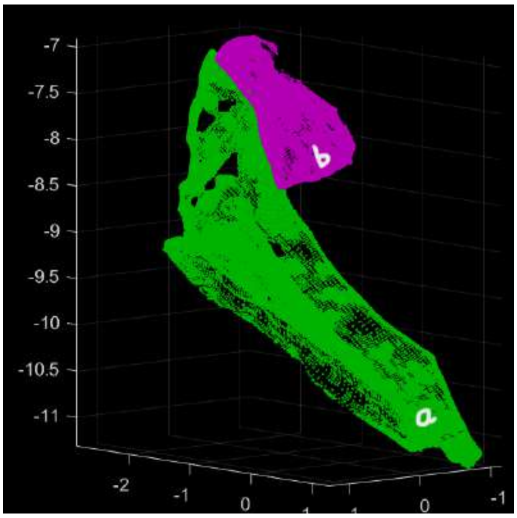
In the alignment stage, a transformation matrix is computed to reveal the relative position between pairs of matching segments connected by a highlighted edge (Fig. 3.10). Subsequently, this matrix is employed to transform the entire shard into the alignment pose.

### 3.7.1 *Iterative Approach*

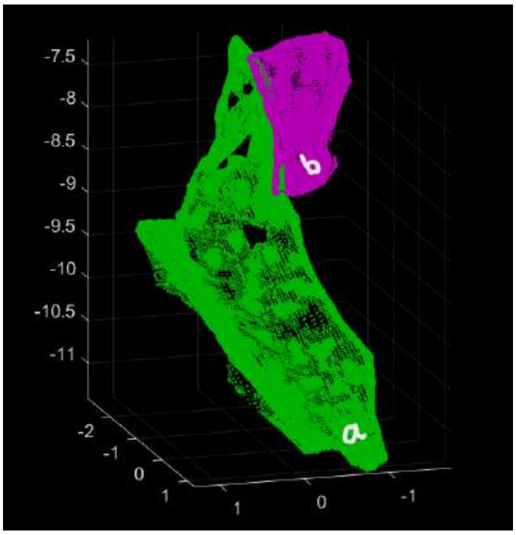
To achieve a seamless visualization through perfect shards positioning, I employed an iterative approach that utilizes positional constraints imposed by neighboring shards. For example, Shard 2 has matching segments with Shards 1 and 3 (Fig.3.10). The iterative alignment approach circulates back and forth between the Pairs 1 and 2, and 2 and 3 until the user verifies the alignment, through Human Computer Interaction.

### 3.7.2 *Sensitivity to Size*

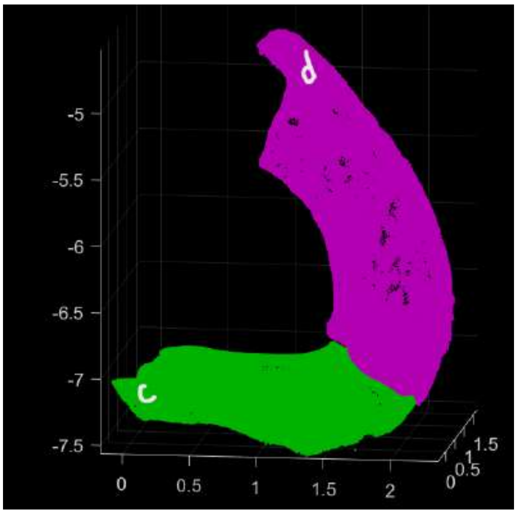
Since SFM does not inherently preserve scale, it is important to scale the 3D models of the shards into the correct dimensions. An approach to resize the shards was scaling the point clouds in MATLAB. However, it was noticed that the alignment algorithm is very sensitive to the sizing of the shards. If the hundredth place of the decimal part of the scaling factor varies, the alignment poses vary. Consequently, obtaining the precise size of the shards is crucial to ensuring a seamless alignment and accurate visualization of the reconstruction. Figure 3.11 shows two examples of pairs being aligned at different scales. The difference between $Scale_1$ and $Scale_2$ is at the hundredth place of the decimal part of the scaling factor.



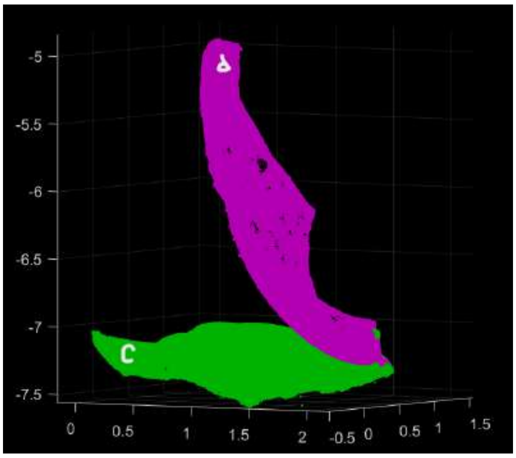

(a) Alignment of two shards *(a and b)* where Shard a is scaled to $Scale_1$.




(b) [Alignment of two shards *(a and b)* where Shard a is scaled to $Scale_2$.




(c) Alignment of two shards *(c and d)* where Shard c is scaled to $Scale_1$.




(d) Alignment of two shards *(c and d)* where Shard c is scaled to $Scale_2$.

Figure 3.11: The sensitivity of the alignment process to the scaling of shards.The change between $Scale_1$ and $Scale_2$ is of order $10^{-2}$.

# Chapter 4

# Experiments

The primary objective of this section is to demonstrate that the method discussed in the earlier chapter holds practical applicability in the reassembly of shattered glassware through edge geometry analysis. By building a dataset of fragmented glass vessels, I aim to evaluate the performance of the proposed system based on accuracy in reconstruction and sensitivity to missing shards.

## 4.1 Validation Dataset

Validating the effectiveness of the algorithm on real broken glass relics was an important goal in this thesis. The Archaeology Museum at the American University of Beirut (AUB) offered a unique opportunity to test the solution on a group of 74 valuable ancient artifacts dating back to the Islamic, Roman, and Byzantine eras [32] that were drastically shattered due to the Beirut Port Explosion (Fig. 4.1).

(a) The display of 74 historic relics [32].

(b) The damage of the glass relics due to the Beirut Blast [32].

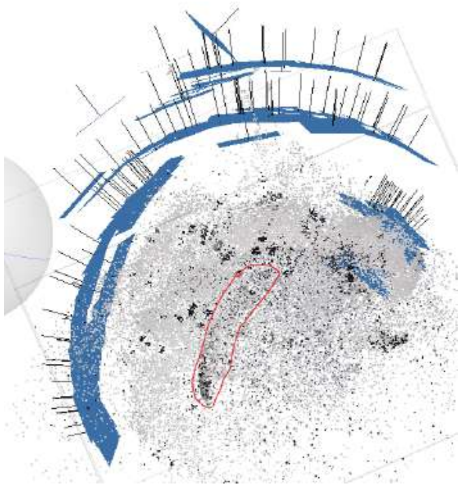
Figure 4.1: To the left: The display of ancient glass vessels at the Archaeology Museum at AUB. To the right: The drastic damage of the vessels after the Beirut Blast [32].

Although there would be a great value in validating the proposed methodology on such historic vessels, several challenges were encountered during the 3D model acquisition stage due to the delicate condition of the fragments.

### 4.1.1 *Limitation of Photogrammetry on Glass Artifacts*

The shards of the ancient relics are transparent in some spots and reflective in others due to iridescence that is technically an age index of historic glassware [33]. Transparency and color change under different light orientations are two main limitations for photogrammetry. The photogrammetric processing failed to reconstruct accurate point clouds of the shards from the Archaeology Museum at AUB (Fig.4.2).

(a) A reflective and iridescent shard from a broken relic.



(b) The photogrammetric reconstruction of the iridescent shard.

Figure 4.2: To the left: A reflective and iridescent shard from a broken relic from the Archaeology Museum at AUB. To the right: The failure of photogrammetry to build a 3D point cloud of the shard shown to the left.

Dusting the shattered glass would have added texture and limited the reflectivity and transparency of the shards, which could have been helpful to enhance the results of phorogrammetry. However, the fragile and sensitive nature of the fragments did not allow dusting, and thus, obtaining 3D models through photogrammetry was not possible. Neural Radiance Field (NeRF) [34] is a novel technology to reconstruct objects using a fully connected neural network that utilizes light information in 2D images, which gave me hope in an efficient alternative in acquiring 3D models of transparent glass.

### 4.1.2 *Neural Radiance Field (NeRF)*

Neural Radiance Field (NeRF) [34] is a fully connected deep network that maps a sparse images set of an object/scene of interest with known camera poses into a synthesized 3D view by optimizing a continuous volumetric scene function. NeRF represents the 3D scene as a continuous function whose input is a 5D query of spatial location coordinates $(x, y, z)$ and viewing direction $(\theta, \phi)$ representing the inclination and azimuth angles. From the camera poses, 3D coordinates are generated for their corresponding sampled points in the images. This correspondence is fed into multi-layer perceptron (MLP) that outputs a query of color intensity $c$ and opacity $\sigma$. The network is trained to estimate the radiance values of points in the images. Back propagation and gradient descent are used to optimize the mapping function and minimize the difference between the predicted radiance values and the ground truth ones in the input images. After the training is complete, rendering new views of the scene starts. In the volumetric rendering stage, the model traces rays into the scene and integrates the radiance along each ray to eventually generate the final scene.

The fact that NeRF performs volume rendering with radiance fields, which describes the interaction of light with the scene's geometry and material at each point in 3D, demonstrates the applicability of NeRF in different fields as it handles transparent and refractive objects accurately [35]. Although NeRF requires intensive computation capacity for training and inference, several adaptations [36], [37] offer faster results. For example, Instant NeRF developed by NVIDIA [37] take few minutes to train an accurate 3D view compared to the original NeRF model that takes hours to output a good-looking scene.

The results of NeRF on the shards of historic relics from the Archaeology Museum (Fig. 4.3) and even on totally transparent fragments (Fig. 4.4) were impressively successful in capturing geometry details. However, the major restriction of utilizing the NeRF output is the 3D mesh quality (Fig. 4.5). Even though using radiance field to build 3D scenes of transparent, reflective, and iridescent objects is possible using NeRF, meshing the obtained scene was not successful.

Since the output mesh of the NeRF scene did not provide satisfactory representation, the decision was to test and validate the algorithm on a dataset of non-historic glassware that are manually broken and which I could dust. NeRF and photogrammetry can be combined to produce high quality meshes of iridescent and reflective glass relics, after dusting them, as a direction for future work to validate and utilize the proposed system for archaeological artifacts. After dusting the shards, a detailed and accurate mesh can be obtained via photogrammetry and then combined with the 3D model built using NeRF. This combination leverages the strengths of both techniques and can offer a high quality 3D reconstruction of the shards.

### 4.1.3 *Dataset Description*

A validation dataset of ten glass vessels of different shapes, thickness, and sizes was collected. The glassware collection was shattered manually and randomly into fragments of arbitrary geometry. The crushing process resulted in totally destroying the area of impact leading to missing shards.

Table 4.1 lists the glass pieces used for testing and validation while providing details on their type and the number of shards obtained after fracture.

(a) The 2D image of an iridescent shard with transparent patches.


(b) The NeRF scene of a historic glassware shard from the AUB Archaeology Museum shows fine details despite transparency and iridescence.

Figure 4.3: At the top: A transparent and iridescent shard from a broken relic from the Archaeology Museum at AUB. At the bottom: The 3D reconstruction of the shard using NeRF with successful representation of features and geometry.

(a) A fully transparent glass fragment.


(b) The NeRF scene of the glass fragment shown to the left.

Figure 4.4: At the top: A fully transparent glass fragment . At the bottom: The NeRF scene of the transparent glass fragment (left) which accurately captures the geometry.

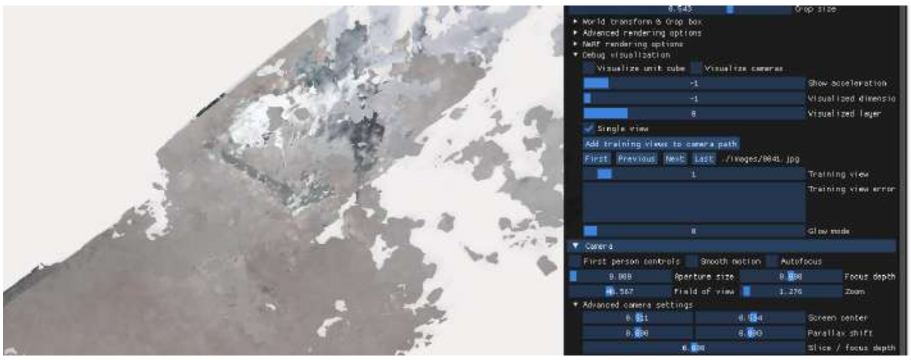


Figure 4.5: Instant NeRF fails to provide an accurate reliable mesh of a glass shard despite capturing an accurate NeRF scene.

Table 4.1: Table of information on the glassware used for the validation dataset.

| Glassware number | Type | Number of shards |
|---|---|---|
| 1 | jug | 11 |
| 2 | bowl | 7 |
| 3 | jar | 13 |
| 4 | jug | 10 |
| 5 | jar | 8 |
| 6 | jar | 14 |
| 7 | cup | 5 |
| 8 | cup | 4 |
| 9 | cup | 3 |
| 10 | vase | 8 |

# Chapter 5

# Results and Discussion

In this chapter, I will delve into the insights of each set of results outlining the robustness of the algorithm through empirical evidence. The presentation of results will be accompanied by thorough interpretations on the performance of the proposed solution in different scenarios.

## 5.1 Results

In Table 5.1, the output of each stage of the reassembly pipeline for each example in the validation dataset is presented. The original pieces with the fragmented shards and their segmented inner contours, the relative non-matching scores yielding to the global optimization of fragments positioning, and the end aligned visualization of the reconstructed pieces are presented. As shown in the table, each glassware is fragmented into shards which are dusted (painted for color and enhanced with texturing doodles). In the third column, the final segmentation of the shards is shown. In column four the non-matching scores among shards are presented in the form of a graph whose optimized solution is shown in the next column. The final alignment that visualizes the poses and relative positions of shards in the overall solution is presented in column five.

By comparing the output in column five to the original form of each glassware in images of column one, the robustness of the proposed system is verified. Each glassware was accurately reconstructed where the position of its shards and the matching areas among them is estimated and visualized for the user. It is noteworthy that the jug in row four (Glassware 4) is not fully reconstructed because of the deconstruction of the rim shards during photogrammetry. The processing time for each glassware depends on the size of the item, the size of its shards, and the quality of the 3D point cloud. On average, the proposed system takes around one hour to estimate the matches after downsampling all the contours.

In the following sections, the proposed systems in tested for complex scenarios that are common in the archaeology field and associated with the surveying and sorting stages when the shards of broken artifacts are found and classified into groups based on their features.

## 5.2 Testing for Failure Modes

In this section, several scenarios are investigated to test the proposed system's robustness to failure modes, such as dealing with missing shards and extrinsic shards.

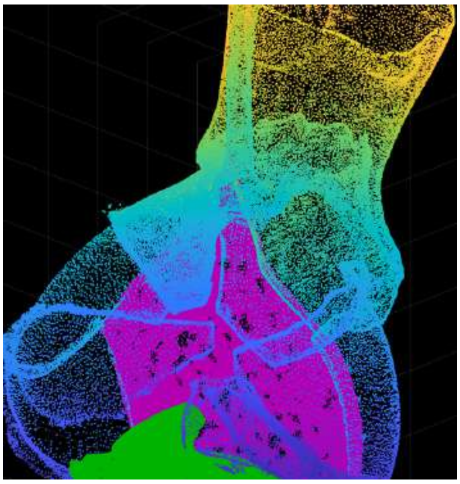
### 5.2.1 *Missing Shards*

The effectiveness of the algorithm in handling missing shards is a critical aspect of performance assessment. For the solution to be practical for archaeological and cultural applications, the proposed solution should show little or no sensitively to missing fragments. As mentioned earlier, the manual fragmentation of the objects led, in most of the cases, to local destruction of the point of impact allowing for a critical assessment of the algorithms sensitivity and applicability. To thoroughly evaluate the performance around the destroyed region where shards are missing, I decided to closely examine two aspects: the effect on the overall solution, and the accuracy of shards alignment around the impact area.

#### 5.2.1.1 The Effect of Missing Shards on the Overall Solution

The algorithm showed high accuracy in uncovering matches despite missing shards. Figures 5.1, 5.2, 5.3, and 5.4 show that geometric similarities were estimated and global optimization successfully determined the matches among the fragments of four different relics even though there are missing shards. As the obtained results are inspected closely, it is realized that the missing shards didn't have any effect on detecting the matches; however, in specific cases, the alignment around the impact area was effected (Fig. 5.4).

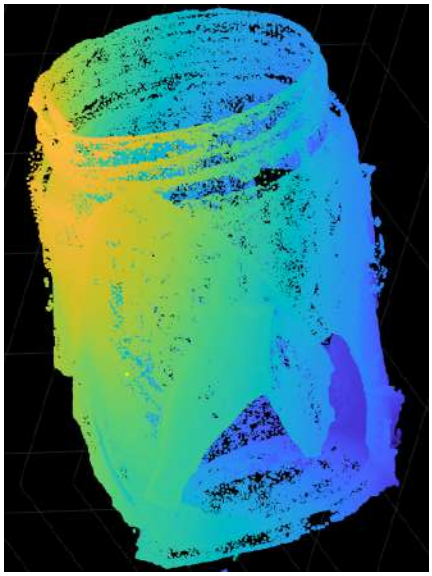
(a) Glassware One with missing shards in impact area.



(b) Reassembly of Glassware One with missing shards in impact area.

Figure 5.1: The results of the overall reassembly of fractured Glassware One with missing shards.

(a) Glassware Three with missing shards in impact area.

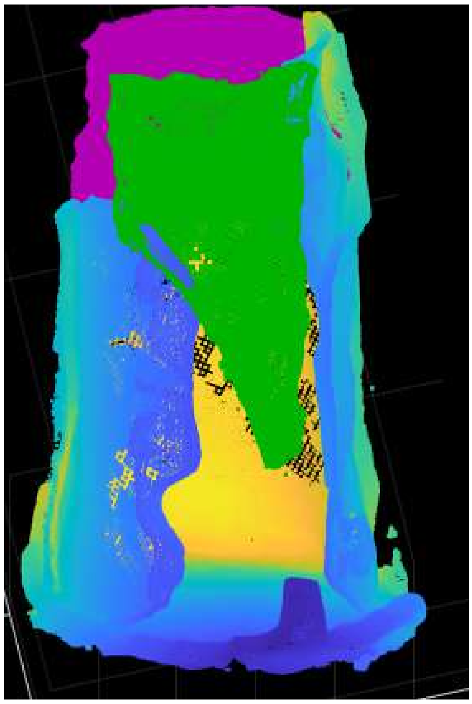


(b) Reassembly of Glassware Three with missing shards in impact area.

Figure 5.2: The results of the overall reassembly of fractured Glassware Three with missing shards.

It is noteworthy that, in all conducted tests, the absent shards position was non-critical. These missing shards did not divide the relic fragments into two groups that could not be connected without the presence of the missing shard.

(a) Glassware Five with missing shards in impact area.



(b) Reassembly of Glassware Five with missing shards in impact area.

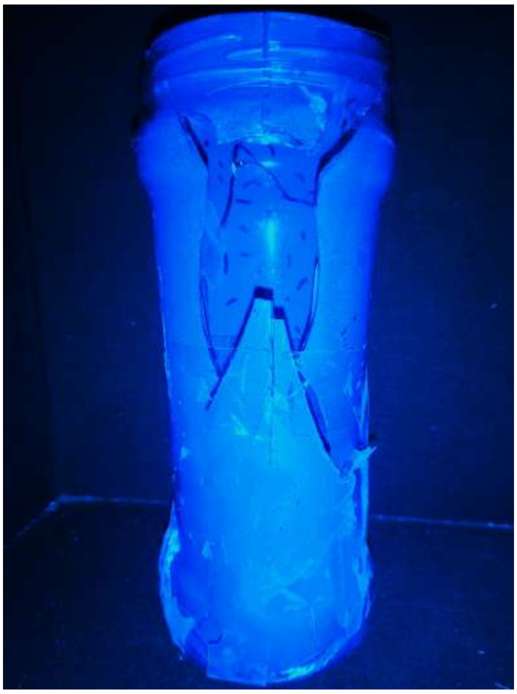
Figure 5.3: The results of the overall reassembly of fractured Glassware Five with missing shards.

#### 5.2.1.2 The Accuracy of Shards Alignment Around the Impact Area

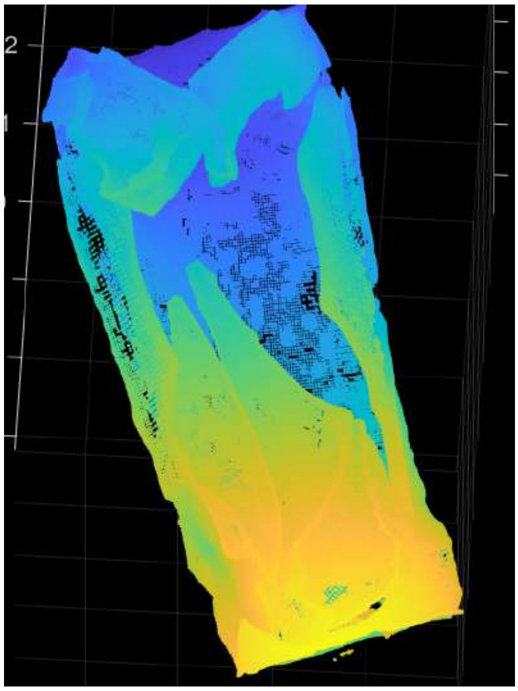
Although the algorithm in a robust manner dealt with missing shards and successfully uncovered the matches among fragments, the alignment of matches around the impact area where shards are missing showed some sensitivity. Recalling the iterative approach outlined for the alignment (Section 3.7), the algorithm navigates back and forth between matches to attain the perfect positioning and alignment. However, in the case of missing shards, this iterative process is less effective for fragments adjacent to the impact area. The limitation arises from the absence of positional constraints on all sides. Figure 5.4 demonstrates how the alignment of the shards of the rim of glassware six was not accurate. The misplaced shard's position is a result of the lack of restrictions imposed by neighboring shards.

### 5.2.2 *Intruder Shards*

It was interesting to explore the performance of the proposed system in handling intruder shards. Excavating for historic fragments is followed by sorting, where archaeologists attempt to group shards based on features like material and texture. This process might include shards that do not belong to a group. At this stage, it is important for the proposed system to be open to exclude the shards that have no matches instead of imposing them to the end solution.



(a) Glassware Six with missing shards in impact area.



(b) Reassembly of Glassware Six with missing shards in impact area.

Figure 5.4: The results of the overall reassembly of fractured Glassware Six with missing shards.

To evaluate the algorithm's efficacy in managing extrinsic shards, two tests where completed. Intruder shards where added to the input query of fragments 3D-models. In the first test (Fig.5.5), an intruder piece was added to the query of shards of Glassware 2, which has no missing fragments. Glassware 2 consists of 7 shards and the intruder shard was numbered 8. The output of the overall optimisation shows that Shard Number 8 was excluded from the graph. This means that there was no matching score calculated between the intruder piece and any other shard high enough to nominate it to the final optimization step. Thus, the proposed system successfully eliminated the intruder shard.

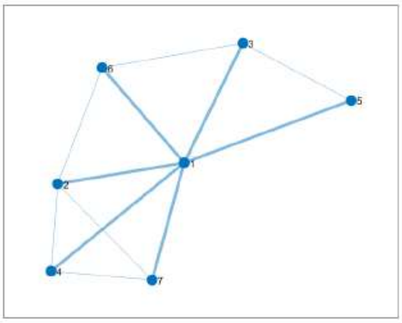
In the second test (Fig.5.6), an intruder piece was added to the shards list of Glassware 3, which has missing fragments. Glassware 3 consists of 13 shards and the intruder shard was numbered 14. Again, the output of the optimization shows that the intruder shard was successfully excluded from the final solution.

## 5.3 Discussion of Robustness and Failure

As presented in the reconstruction results (Sec.5.1) and the performance of the proposed system under several tests (Sec.5.2), the proposed system proves to be robust in exploring matches among shards and finding the overall solution to reassemble broken glass relics. In the case of missing shards, the proposed system successfully

(a) Shards of Glassware Two with no missing shards.

(b) Intruder shard added to the query of Glassware Two.




(c) The MST showing the exclusion of non-belonging shards in the reassembly of Glassware Two.

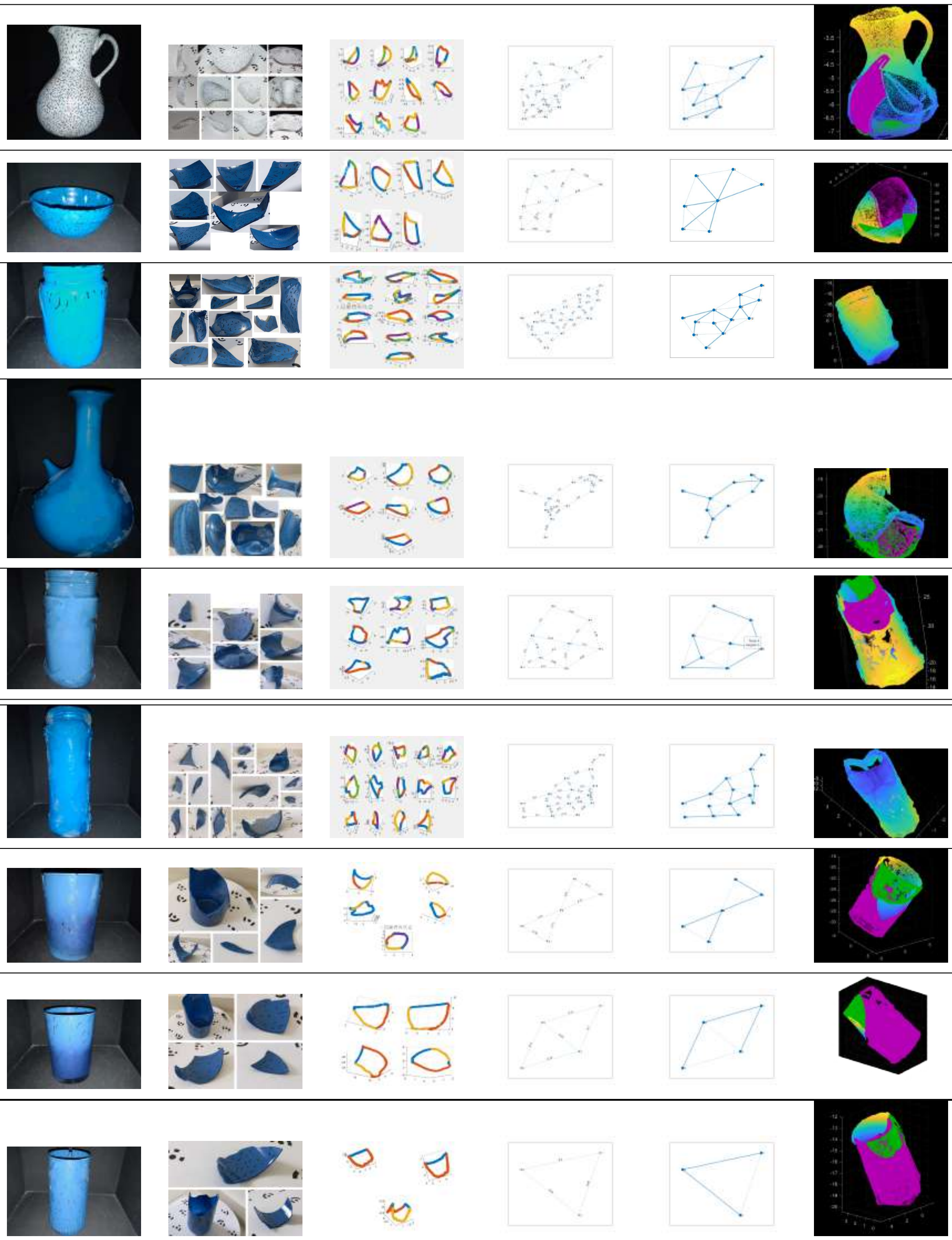
Figure 5.5: The test that was done to evaluate the performance of the algorithm in handling extrinsic shards added to the group of shards belonging to Glassware Two which has no missing shards.

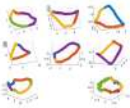
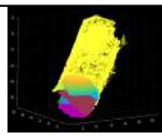
quantifies similarities among pairs and optimizes matches to obtain the overall solution. However, the alignment of the shards around the impact area, where shards are missing, shows noticeable sensitivity that affect the final orientation of the shards poses but not the correct position of the shards.

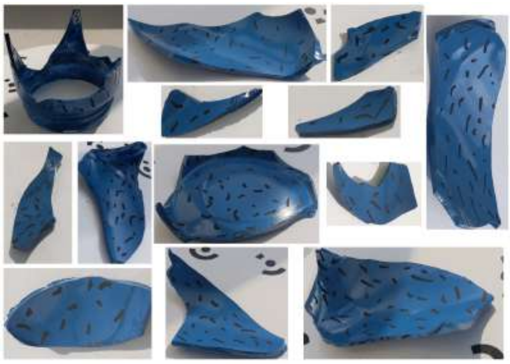
In addition, the proposed system showed great robustness in dealing with intruder shards which don't belong to the relics under construction and thus have no matching contours with any of the other shards. As the system quantifies the contours geometric similarities of the intruder shard with all the other shards to be a low score, it automatically dismiss the intruder shard without including it in the global optimization stage and thus in the alignment stage.

Table 5.1: Table of results at each stage of the reassembly pipeline for 10 fractured glassware. The Table features the original objects with the processing at consecutive steps leading to the visualized solution estimated by the proposed algorithm.

| Original Object | Shards | Segmented Contours | Non-Matching Score | Global Optimization | Alignment |
|---|---|---|---|---|---|

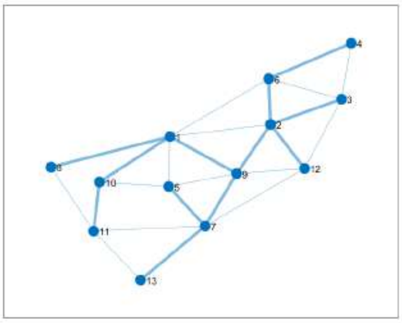


| Original Object | Shards | Segmented Contours | Matching Score | Global Optimization | Alignment |
|---|---|---|---|---|---|
|  |  |  |  |  |  |



(a) Shards of Glassware Three with missing shards.

(b) Intruder shard added to the query of Glassware Three.




(c) The MST showing the exclusion of non-belonging shards in the reassembly of Glassware Three.

Figure 5.6: The test that was done to evaluate the performance of the algorithm in handling extrinsic shards added to the group of shards belonging to Glassware 3 which has missing shards.

# Chapter 6
# Conclusion and Future Work

Leveraging technology and digital tools provides valuable solutions for preserving cultural heritage. This thesis integrates computer vision techniques along with manifold learning to develop an algorithm focused on uncovering local geometry. Specifically, the algorithm addresses the challenge of solving 3D puzzles by exploiting similar edge geometry among 3D models of historic glass relics shards.

The proposed system was tested on ten manually shattered glassware items with varying profiles. The system showed robust performance in estimating and quantifying geometric similarities among pairs of shards based on exploiting their contours, in addition to a successful global solution for the optimization problem of the multi-fragment matching. To examine the performance of the proposed system in depth, failure mode tests were done simulating different scenarios such as missing shards and intruder shards, which are common in archaeology. Dealing with missing shards was done successfully without affecting the pairwise matching and the multi-fragment matching stages. However, it is noteworthy that in all the tests that were done to evaluate the performance of the system when dealing with missing shards, the position and the size of the missing shards was not critical in a way that splits the glassware into two unconnected pieces. Even though the system uncovered the matches successfully, the absence of certain shards impacted the alignment shards close to the impact area because some positional constraints imposed by neighboring shards were lost. On another note, the proposed system was also tested against intruder shards. It dealt with foreign shards by estimating very low matching scores with other pieces during the pairwise matching stage, thus eliminating them in the global optimization and alignment stages.

While the algorithm was not directly tested on historic relics, considering the merge of NeRF technology with SFM presents a promising plan for future research. This approach aims to acquire accurate 3D point clouds of ancient shards, enhancing the possibility to apply the developed algorithm in cultural heritage applications.

# Bibliography


[1] M. Kampel and R. Sablatnig, "Virtual reconstruction of broken and unbroken pottery," in *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.*, 2003, pp. 318–325. DOI: 10.1109/IM.2003.1240265.

[2] N. A. Rasheed and M. J. Nordin, "A survey of computer methods in reconstruction of 3d archaeological pottery objects," *International Journal of Advanced Research*, vol. 3, no. 3, pp. 712–714, 2015.

[3] C. Toler-Franklin, B. Brown, T. Weyrich, T. Funkhouser, and S. Rusinkiewicz, "Multi-feature matching of fresco fragments," *ACM Trans. Graph.*, vol. 29, no. 6, 2010, ISSN: 0730-0301. DOI: 10.1145/1882261.1866207. [Online]. Available: https://doi.org/10.1145/1882261.1866207.

[4] G. Song and H. Wang, "Artificial intelligence-assisted fresco restoration with multiscale line drawing generation," *Complexity*, vol. 2021, pp. 1–12, 2021.

[5] R. Lotus, J. Varghese, and S. Saudia, "An approach to automatic reconstruction of apictorial hand torn paper document.," *Int. Arab J. Inf. Technol.*, vol. 13, no. 4, pp. 457–461, 2016.

[6] F. Kleber and R. Sablatnig, "A survey of techniques for document and archaeology artefact reconstruction," in *2009 10th International Conference on Document Analysis and Recognition*, 2009, pp. 1061–1065. DOI: 10.1109/ICDAR.2009.154.

[7] B. Liu, M. Wang, X. Niu, S. Wang, S. Zhang, and J. Zhang, "A fragment fracture surface segmentation method based on learning of local geometric features on margins used for automatic utensil reassembly," *Computer-Aided Design*, vol. 132, p. 102 963, 2021, ISSN: 0010-4485. DOI: https://doi.org/10.1016/j.cad.2020.102963. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010448520301561.

[8] Q. Li, G. Geng, and M. Zhou, "Pairwise matching for 3d fragment reassembly based on boundary curves and concave-convex patches," *IEEE Access*, vol. 8, pp. 6153–6161, 2020. DOI: 10.1109/ACCESS.2019.2961391.

[9] A. R. Willis and D. B. Cooper, "Computational reconstruction of ancient artifacts," *IEEE Signal Processing Magazine*, vol. 25, no. 4, pp. 65–83, 2008. DOI: 10.1109/MSP.2008.923101.

[10] G. Papaioannou, T. Schreck, A. Andreadis, *et al.*, "From reassembly to object completion: A complete systems pipeline," *J. Comput. Cult. Herit.*, vol. 10, no. 2, 2017, ISSN: 1556-4673. DOI: 10.1145/3009905. [Online]. Available: https://doi.org/10.1145/3009905.

[11] G. P. R. G. T. S. Anthousis A. Pavlos M., "State of the art report on object matching, reassembly and completion methods," Trondheim, Norway, Tech. Rep., 2013.

[12] D. Goldberg, C. Malon, and M. Bern, "A global approach to automatic solution of jigsaw puzzles," ser. SCG '02, Barcelona, Spain: Association for Computing Machinery, 2002, 82–87, ISBN: 1581135041. DOI: 10.1145/513400.513410. [Online]. Available: https://doi.org/10.1145/513400.513410.

[13] L. Zhu, Z. Zhou, and D. Hu, "Globally consistent reconstruction of ripped-up documents," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 1, pp. 1–13, 2008. DOI: 10.1109/TPAMI.2007.1163.

[14] E. Tsamoura and I. Pitas, "Automatic color based reassembly of fragmented images and paintings," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 680–690, 2010. DOI: 10.1109/TIP.2009.2035840.

[15] G. Oxholm and K. Nishino, "Reassembling thin artifacts of unknown geometry," in *Proceedings of the 12th International Conference on Virtual Reality, Archaeology and Cultural Heritage*, ser. VAST'11, Prato, Italy: Eurographics Association, 2011, 49–56, ISBN: 9783905674347.

[16] H. da Gama Leitao and J. Stolfi, "A multiscale method for the reassembly of two-dimensional fragmented objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1239–1251, 2002. DOI: 10.1109/TPAMI.2002.1033215.

[17] A. Willis and D. Cooper, "Bayesian assembly of 3d axially symmetric shapes from fragments," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, 2004, pp. I–I. DOI: 10.1109/CVPR.2004.1315017.

[18] M. Zhou, G. Geng, Z. Wu, and W. Shui, "A virtual restoration system for broken pottery," 2010.

[19] G. Papaioannou, E.-A. Karabassi, and T. Theoharis, "Segmentation and surface characterization of arbitrary 3d meshes for object reconstruction and recognition," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 1, 2000, 734–737 vol.1. DOI: 10.1109/ICPR.2000.905491.

[20] W. Kong and B. B. Kimia, "On solving 2d and 3d puzzles using curve matching," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, pp. II–II, 2001. [Online]. Available: https://api.semanticscholar.org/CorpusID:17259657.

[21] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511811685.

[22] Z. Cui and P. Tan, "Global structure-from-motion by similarity averaging," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[23] M. Klopschitz, A. Irschara, G. Reitmayr, and D. Schmalstieg, "Robust incremental structure from motion," in *Proc. 3DPVT*, vol. 2, 2010, pp. 1–8.

[24] F. Ernst, P. Wilinski, and K. van Overveld, "Dense structure-from-motion: An approach based on segment matching," in *Computer Vision — ECCV 2002*, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 217–231.

[25] S. N. Sinha, "Multiview stereo," in *Computer Vision: A Reference Guide*, K. Ikeuchi, Ed. Boston, MA: Springer US, 2014, pp. 516–522, ISBN: 978-0-387-31439-6. DOI: 10.1007/978-0-387-31439-6_203. [Online]. Available: https://doi.org/10.1007/978-0-387-31439-6_203.

[26] L. Cayton, "Algorithms for manifold learning," *Univ. of California at San Diego Tech. Rep*, vol. 12, no. 1-17, p. 1, 2005.

[27] Z. Zhang and H. Zha, "Nonlinear dimension reduction via local tangent space alignment," in *Intelligent Data Engineering and Automated Learning: 4th International Conference, IDEAL 2003, Hong Kong, China, March 21-23, 2003. Revised Papers 4*, Springer, 2003, pp. 477–481.

[28] J. Park, *Manifold learning in computer vision*. The Pennsylvania State University, 2005.

[29] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 658–666.

[30] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proceedings of the American Mathematical society*, vol. 7, no. 1, pp. 48–50, 1956.

[31] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International journal of computer vision*, vol. 13, no. 2, pp. 119–152, 1994.

[32] M. Asser, *Aub archaeological museum counts cost of beirut blast*, 2020. [Online]. Available: https://aub.edu.lb/articles/Pages/archaeological-museum-beirut-blast.aspx?fbclid=IwAR2cQU7qdlpP9rxJ_f_9U3k75373QfNiK-o53hOEvzNQQ9JqGRZ4R4Emmcw.

[33] M. Emami, S. Nekouei, H. Ahmadi, C. Pritzel, and R. Trettin, "Iridescence in ancient glass: A morphological and chemical investigation," *International Journal of Applied Glass Science*, vol. 7, no. 1, pp. 59–68, 2016. DOI: https://doi.org/10.1111/ijag.12182. eprint: https://ceramics.onlinelibrary.wiley.com/doi/pdf/10.1111/ijag.12182. [Online]. Available: https://ceramics.onlinelibrary.wiley.com/doi/abs/10.1111/ijag.12182.

[34] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, *Nerf: Representing scenes as neural radiance fields for view synthesis*, 2020. arXiv: 2003.08934 [cs.CV].

[35] J. Ichnowski, Y. Avigal, J. Kerr, and K. Goldberg, "Dex-nerf: Using a neural radiance field to grasp transparent objects," *arXiv preprint arXiv:2110.14217*, 2021.

[36] M. Tancik, E. Weber, E. Ng, *et al.*, "Nerfstudio: A modular framework for neural radiance field development," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023, pp. 1–12.

[37] T. Müller, A. Evans, C. Schied, *et al.*, "Instant neural radiance fields," in *ACM SIGGRAPH 2022 Real-Time Live!* 2022, pp. 1–2.