Computers & Graphics 99 (2021) 139–152

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on 3DOR 2021

Cross-time registration of 3D point clouds

Evdokia Saiti*, Antonios Danelakis, Theoharis Theoharis

NTNU Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway

ARTICLE INFO

Article history:
Received 31 March 2021
Revised 25 June 2021
Accepted 2 July 2021
Available online 9 July 2021

Keywords:
3D registration
Alignment
Cross-time
Retrieval
Cultural heritage
Erosion

ABSTRACT

Registration is a ubiquitous operation in visual computing and constitutes an important pre-processing step for operations such as 3D object reconstruction, retrieval and recognition. Particularly in cultural heritage (CH) applications, registration techniques are essential for the digitization and restoration pipelines. Cross-time registration is a special case where the objects to be registered are instances of the same object after undergoing processes such as erosion or restoration. Traditional registration techniques are inadequate to address this problem with the required high accuracy for detecting minute changes; some are extremely slow. A deep learning registration framework for cross-time registration is proposed which uses the DeepGMR network in combination with a novel down-sampling scheme for cross-time registration. A dataset especially designed for cross-time registration is presented (called ECHO) and an extensive evaluation of state-of-the-art methods is conducted for the challenging case of cross-time registration.

1. Introduction

Geometric registration (or alignment) is a crucial tool in visual computing with applications in robotics, medical imaging and cultural heritage (CH) analysis, among many others. Registration of datasets and particularly point clouds, has become a key operation in many shape analysis tasks, such as 3D object retrieval [1,2], semantic segmentation and classification [3,4], 3D mapping [5–7], 3D object scanning [8] and 3D model reconstruction [9–11].

Registration aims to find the transformation that optimally aligns two or more similar objects or two or more instances of the same object taken at different times (cross-time data), from different viewpoints (multi-view data) or by different sensors (multi-sensor data), in order to bring the data into a common reference frame [12]. The surface alignment problem is a broad research topic and advances have been made over the years, resulting in a plethora of different strategies and algorithms. However, there are still open problems to be addressed, especially in the context of CH. Archaeological objects differ from mechanical or medical objects in their shape and size (some CH objects can be quite large), articulation and fragility. Moreover, the number of objects digitized and available for experimentation is limited in CH.

Computing has greatly aided the CH field over the last decades, including the restoration, preservation and monitoring processes [13]. In monitoring, microgeometric changes over time are measured and analyzed in order to support conservation strategies [14]. CH objects have been constantly undergoing changes or degradation over time. In this matter, geometric acquisition and measurements of a CH object produce snapshots of 3D models and can be used to track an object through time, in order to document different phases of the conservation pipeline and identify any destructive intervention, or to understand any damages that these modifications may indicate. 3D surface registration can automate the process of monitoring CH artefacts in a non-invasive manner by aligning the objects in such a way that even minute modifications on the object's surface or shape can be automatically detected and measured.

As CH digitization is becoming more widespread, CH object monitoring activities based on the digitized objects are increasingly relevant. Several methodologies have been proposed over the last years, but the contribution is limited due to the relatively small number of digitized CH objects than can be used in the experimentation with the monitoring process. The main reasons are that the conservation process is time consuming and needs to be planed properly so as not to harm the CH object and that the change detected from environmental erosion cannot be easily identified unless several decades pass. The lack of an adequate digital benchmark for deeper analysis and comparison is a major obstacle towards the development of automatic techniques for proper monitoring and documenting different phases of conservation. Such a benchmark is crucial for comparing methodologies and scenaria.

This work is focused on the pairwise cross-time registration problem. We introduce a registration methodology that copes with

* Corresponding author.
E-mail address: evdokia.saiti@ntnu.no (E. Saiti).

https://doi.org/10.1016/j.cag.2021.07.005

big data using a down-sampling scheme that is appropriate for objects that undergo erosion over time and overcomes limitations like the computational complexity of iterative methods, the necessity for point-level correspondence or a coarse pre-alignment step. Moreover, we address the absence of benchmarking data by contributing a dataset of artificially eroded CH objects, including their ground truth transformation. The initial models are taken from the SHREC 2021 dataset for retrieval of CH objects [15] and have been artificially eroded based on weathering conditions resulting from polluted environments and from naturally occurring climatic conditions [16].

The contributions of this paper are:

- The problem of cross-time 3D registration is formally defined and a framework for cross-time 3D registration is proposed. Publicly available upon publication.
- A down-sampling methodology that detects the most valuable points for cross-time registration is proposed. Publicly available upon publication.
- A benchmark for evaluating both traditional and cross-time registration algorithms is created. Publicly available upon publication.
- An extensive evaluation of both geometry-based and deep learning state-of-the-art approaches on 3D cross-time registration is performed.

The remainder of this paper is organized as follows: In Section 2 related works are discussed while in Section 3 the problem of cross-time 3D registration is defined. In Section 4 the proposed methodology for cross-time 3D registration is introduced while Section 5 presents the proposed evaluation benchmark. Experimental results on cross-time registration are presented in Section 6. The paper is concluded in Section 7.

2. Related work

Since surface registration is fundamental to many visual computing domains, there is a very extensive literature on the subject. However, to the best of our knowledge, there exists no methodology specifically for cross-time registration. Instead, standard point cloud or surface registration techniques have been used, but the results are sub-optimal as we shall see later. In this section, we review the methods that are most related to cross-time registration. For a comprehensive review of general registration methods, the interested reader is referred to [17] and for a survey oriented to cultural heritage applications to [18].

Registration methods can be roughly classified into two broad categories, local and global. Global registration techniques align the source and target objects without any prior information about their relative pose, whereas in local registration, a prior coarse transformation is known and the algorithm tries to refine the solution. In general, local approaches are more accurate but less robust to initial pose than global approaches. Examples of local approaches are the well-known Iterative Closet Point (ICP) [19] and its variants [20], while RANSAC [21] and Fast Global Registration [22] are examples of global methods. Further, registration approaches can rely on point-to-point correspondences between the data or be correspondence-free [23].

In addition to geometry-based registration techniques, there has been a recent wave of deep learning approaches, attempting to overcome the challenge of prolonged running time and aiming to boost accuracy further [12].

2.1. Geometry-based registration

Correspondence-based methodologies are based on the observation that computing the optimal alignment between two surfaces is equivalent to finding corresponding points and then computing the transformation that best aligns them with respect to minimizing a specific distance function. The Iterative Closest Point (ICP) [19] is the best-known and most applied such algorithm for solving rigid registration problems. ICP iteratively alternates between finding point-to-point correspondences and distance minimization to compute the optimal alignment. Given its popularity, a large number of variants have appeared [20,24] but there are some drawbacks. The method is local and, thus, is effective only when the initial pose of the input geometries is close to the global optimum, otherwise it can converge to a local minimum. Moreover, the iterative nature of the algorithm and its point-to-point correspondence nature result in high computational complexity. In addition, real-world data and particularly in the case of cross-time registration where erosion is involved, do not contain exact point level correspondences.

To overcome the issues of point-to-point matching, many strategies try to identify feature-level similarities and correspondences. Approaches like RANSAC [21] and Fast Global Registration (FGR) [22] use feature descriptors and matching combined with robust fitting or optimization techniques to achieve registration. These techniques are much more efficient than point-level methods but are highly dependent on the quality of features. Feature-based techniques generally involve three steps: feature detection, feature description and correspondence estimation. Features are a small group of interest points that can be detected on both objects, due to their distinctiveness or geometric stability under different transformations. Each feature can be delineated by a descriptor that characterizes its geometric information. Two main categories of descriptors exist: global and local. Global descriptors represent the geometric information of an entire 3D object, whereas local descriptors encode the local information at each feature point [25]. Specifically for 3D registration local descriptors are more commonly used, because they can identify similar localities between the two surfaces to be aligned by exploiting the geometric properties around a certain point and its neighborhood.

A large number of descriptors have been proposed. Diez et al. presented an analytical review in [26], however not every descriptor is suitable for cross-time registration. Some potentially applicable methodologies are next described. Fast Point Feature Histogram (FPFH) [27] consists of pose-invariant features and is generated as a simplified point feature histogram for each key point and its k-nearest neighbors. Johnson and Hebert introduced the Spin Image (SI) descriptor [28], a rigid transformation-invariant 2D characterization of the surface location around a support region of a specific point. This descriptor obtains competitive results in rigid registration, but is vulnerable to symmetries, noise and clutter. The Radial Intersection Count Image (RICI) descriptor [29], a variation of the SI, has been proposed to overcome the limitations of cluttered scenes and is a 2D histogram of integers that represent the number of intersections of circles centered over the point of interest with the 3D surface. Another variant of the SI is the Scale Invariant Spin Image mesh descriptor (SISI) [30], where the SI descriptor is computed over an estimated local scale at each interest point. The same authors also proposed the Local Depth SIFT (LD-SIFT) [30], a rotation and scale invariant descriptor based on the prior work of Lowe [31]. LD-SIFT represents the vicinity of the each interest point as a depth map by computing a local radial-angular histogram of the pixel value derivatives.

Another approach to registration is based on the branch-and-bound framework [32,33] where the low dimensionality (6DoF) is taken as an advantage to exhaustively search the Special Euclidean Group SE(3) space for the optimal alignment. Although, these methods can achieve a good matching regardless of initial conditions, they often have low efficiency. A popular methodology is the use of statistical models for outlier rejection and geometric

alignment. Specific methods include the use of the Expectation-Maximization (EM) [34] principle for finding accurately and efficiently the alignment transformation [35] and the use of Gaussian Mixture Models (GMMs) to reformulate the point-to-point correspondence problem in a lower dimension resulting in a computationally efficient solution, resistant to noise and outliers [36,37].

2.2. Learning-based registration

Significant recent advances of deep learning methodologies on 3D point clouds provide new opportunities for learning point cloud representations. Milestones like PointNet [38] and DGCNN [3] offer structured representations of 3D point clouds and even if originally designed for point cloud classification and segmentation, they have be transformed and applied to point cloud registration. Learning-based registration has recently shown robustness and efficiency gains over geometry-based techniques.

PointNetLK [39] integrates the Lucas & Kanade (LK) algorithm [40] with the PointNet network for aligning the global features produced by the latter. PointNetLK performs well on shapes unseen in training, but is not robust to noise. PCRNet [41], like PointNetLK, uses PointNet to encode the shape information of the input point clouds but replaces the Lucas-Kanade step by a deep network. DCP [42] is a non-iterative, one-shot algorithm that uses a Siamese DGCNN [3] network to extract the learned correspondences and a differentiable SVD method for registration. RPM-Net [43] tries to improve the robustness to partial visibility by inheriting the idea of the RPM algorithm [44] and incorporating it in a deep network. DeepGMR [45] integrates Gaussian Mixture Model (GMM) registration [36] with neural networks by extracting pose-invariant correspondences between raw point clouds and GMM parameters. Then, these correspondences are fed into the GMM optimization module to estimate the transformation matrix in a single step. The method is efficient and robust to arbitrary displacements and noise. Although, DeepGMR shows highly accurate results, it estimates the correspondence between all points and all components in the latent GMM, which is not suitable for real-life applications and especially in the case of 3D objects that are changing over time.

2.3. Partial registration

A more challenging sub-problem of 3D registration is partial registration, where only subsets of the source and the target object match to one another. Having partially overlapping areas, the alignment is performed by registering the mutually shared patches. Several methods attempt to find correspondences in the area of overlap by identifying keypoints that are common in both source and target. Super4PCS [46], is a variant of RANSAC which iteratively aligns congruent sets of four points taken from the source and the target object. The number of iteration is adaptive, so that when the partial overlap is low, more iterations are performed to reach an acceptable registration result, regardless of initial pose and overlap percentage. Other methods are variants of ICP that deal with noisy data and partial overlap by using general optimization algorithms, like Simulating Annealing [24] and Particle Swarm Optimization [47]. More recently, partial registration has been addressed by PRNet [48], which follows an iterative refinement strategy. It uses deep networks to detect the points of interest followed by estimating the correspondences iteratively in a coarse-to-fine manner to perform the final registration.

Cross-time registration and partial registration share a lot of characteristics. However, there is a crucial difference: in partial registration it is assumed that where overlaps exist, the shape has not changed, while in cross-time registration the objects may encounter considerable shape differences throughout their surface.

3. Problem statement

3.1. 3D registration

In 3D registration we are given two 3D point clouds, the source $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ \ i = 1, 2, \ldots, N\}$ and the target $\mathbf{Q} = \{q_j \in \mathbb{R}^3 |\ \ j = 1, 2, \ldots, M\}$ and the objective is to recover the unknown rigid transformation $\mathbf{T}$ so as to match the source $\mathbf{P}$ into the target point cloud $\mathbf{Q}$.

A rigid transformation in 3D can be represented by a transformation matrix $\mathbf{T}$ which consists of two components; a rotation submatrix $\mathbf{R}$ and a translation vector $\mathbf{t}$. The rigid transformation $\mathbf{T}$ can then be represented by the following homogeneous 4×4 matrix:

$$\mathbf{T} = \left[\begin{array}{c|c} \mathbf{R} & \mathbf{t} \\ \hline 0 & 1 \end{array}\right] \tag{1}$$

where $\mathbf{T} \in SE(3), \mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$. $SE(3)$ is the special Euclidean group of rigid transformations in 3D space (rotations and translations), while $SO(3)$ is the special orthogonal group of rotations in Euclidean Space $\mathbb{R}^3$.

The problem of rigid registration between two discrete point clouds can be formulated as [49]:

$$\arg\min_{\mathbf{R},\mathbf{t}} \sum_{i=1}^{N} d(\mathbf{R}\mathbf{p_i} + \mathbf{t}, \mathbf{Q}) \tag{2}$$

where function $d(\mathbf{p}, \mathbf{Q})$ measures the distance of an arbitrary point $\mathbf{p} \in \mathbf{P}$ to the point cloud $\mathbf{Q}$ and can be defined as:

$$d(\mathbf{p}, \mathbf{Q}) = \min_{\mathbf{q} \in \mathbf{Q}} d(p, q) \tag{3}$$

where $d(p, q)$ is the distance between two points in space.

Eq. (3) is referred to as the distance or error metric. Many methods [24,32] use the squared Euclidean norm as the distance metric and optimize Eq. (2) using least squares:

$$\arg\min_{\mathbf{R},\mathbf{t}} \sum_{i=1}^{N} \|\mathbf{R}\mathbf{p_i} + \mathbf{t} - \widehat{\mathbf{q}}_\mathbf{i}\|^2 \tag{4}$$

where $\widehat{\mathbf{q}}_\mathbf{i}$ is the closest point in $\mathbf{Q}$ to each point $\mathbf{p_i} \in \mathbf{P}$ based on the transformation $\mathbf{T}(\mathbf{R}, \mathbf{t})$.

3.2. The cross-time 3D registration problem

Methods that monitor the geometric variation of an object over time, must try to compare the 3D representations of the same object captured at different points in time. During these time intervals, several modifications like degradation from environmental erosion, cleaning and conservation actions, or even cracking may have occurred on the surface of the object. Therefore, it is not expected that the acquisition process will start at the exact same position at both times; thus the 3D point clouds will not have the same number of points and no perfect correspondences.

Various decay phenomena and alteration processes may occur to the surface of a CH object. Alterations can be due to weathering conditions, physical or chemical aging or human intervention [50]. The material alteration processes can cause local loss of the surface (bursting, chipping, peeling), change in shape (deformation, blistering, delamination, exfoliation, crumbling), cracks (splitting, hair cracks, star cracks) or changes in texture (discoloration, bleaching, staining). Moreover, any conservation process can be considered as an alteration operation to the object, even though it does not imply a worsening of its characteristics and shape (e.g. application of reversible coating, varnish removal or mechanical and chemical cleaning).

Let us define the initial CH object as a set of 3D points $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ and the altered object as $\widehat{\mathbf{P}} = \{\widehat{p}_j \in \mathbb{R}^3 |\ j =$

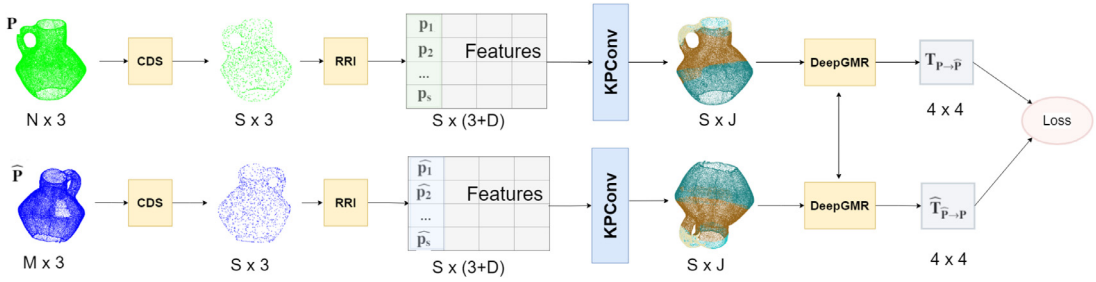

Fig. 1. Overview of the proposed *CrossTimeReg* cross-time 3D registration pipeline.

$1, 2, \ldots, M\}$. Without loss of generality, one can assume the existence of a change function f_{ch} that describes the modifications that the initial object has undergone, so that $\widehat{\mathbf{P}} = f_{ch}(\mathbf{P})$. f_{ch} may encompass various types of alterations.

In this framework, the 3D cross-time registration problem can be formulated as: given two 3D point clouds of the same object but captured at different time frames, the source $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ and the target $\widehat{\mathbf{P}} = \{\widehat{p}_j \in \mathbb{R}^3 |\ j = 1, 2, \ldots, M\}$ with $N \neq M$, the aim is to find the unknown rigid transformation $\mathbf{T}$ so as to align the source $\mathbf{P}$ onto the target $\widehat{\mathbf{P}}$ as well as possible for a specific distance metric d:

$$\arg\min_{\mathbf{R},\mathbf{t}} \sum_{i=1}^{N} d(\mathbf{R}\mathbf{p_i} + \mathbf{t}, \widehat{\mathbf{P}}) \tag{5}$$

The problem of cross-time registration can be really challenging if all the different aspects of alterations that a CH object may experience are taken into account. In this work, we focus on the simplified but still challenging case of weathering erosion, where we assume that the objects have been uniformly exposed to environmental effects, both spatially and temporally. We are motivated from the observation of [51] that a typical registration algorithm like ICP [19], will align the source and target point clouds $\mathbf{P}$ and $\widehat{\mathbf{P}}$, so as to minimise the error (i.e. RMSD, Chamfer distance) between them. In doing this, the registration process will often bring the two point clouds close together in certain areas, most probably where the sampling density is higher. This is not ideal where objects have undergone uniform erosion across their surface as shown in the experiments of [52].

This problem has a number of interesting characteristics, especially when considered in the Cultural Heritage domain where the cross-time nature of $\mathbf{P}$ and $\widehat{\mathbf{P}}$ arises after erosion over a long time period:

- A classic registration algorithm will weigh more areas with dense sampling as more points are contributing to the error metric; however an erosion process is more likely to affect the surface of the object evenly and thus a resampling process is required.
- As $\mathbf{P}$ and $\widehat{\mathbf{P}}$ can be assumed to be the same object, we know that there exists an ideal registration $(\mathbf{R}, \mathbf{t})_{\mathbf{ideal}}$. However, as object scans are likely to have been taken across several years, probably with different scanner technology and without external reference points, $(\mathbf{R}, \mathbf{t})_{\mathbf{ideal}}$ is not known. In Section 5 we have created a synthetically eroded dataset where $(\mathbf{R}, \mathbf{t})_{\mathbf{ideal}}$ is known by definition and can be used for training and benchmarking cross-time registration algorithms.

4. Method overview

In this Section, the *CrossTimeReg* framework is presented, see Fig. 1 for the pipeline. The initial and eroded point clouds (also referred to as *source* and *target*) are denoted by $\mathbf{P}$ and $\widehat{\mathbf{P}}$ respectively. $\mathbf{P}$ and $\widehat{\mathbf{P}}$ are first down-sampled using the Curvature Down-Sampling (CDS) block and then rotation invariant features are computed by the Feature Extraction block (RRI). The features along with the point clouds are then sent to a Siamese architecture of KPConv networks. KPConv network is a segmentation network, which estimates for each point the component that it belongs to; it thus determines a point-to-component correspondence. Finally, the registration is performed by aligning the component centroids (weighted by the covariances) using the DeepGMR module, a weighted version of the SVD solution proposed in [42].

Curvature down-sampling (CDS): Registration algorithms often use a down-sampling pre-processing step on the input point clouds to accelerate the registration process. Some methods [30] detect the most interesting points and compute a descriptor for each of them while others [45,48] keep the nearest or farthest S points to the centroid of the object. In traditional registration, these methods may be sufficient as the local shape of the source and target object is not expected to vary. However in cross-time registration, the target object's local shape is expected to be modified due to erosion and other effects and the aforementioned down-sampling approaches may fail. To address this, we propose a down-sampling approach for cross-time registration that takes into consideration the points that are less likely to be significantly altered by erosion. We expect these points to be those with the minimum principal curvature [53,54]. The intuitive reason behind this is that such points are less exposed to erosion/degradation processes or conservation activities. Thus they are considered to be a robust representation of the object across such processes or activities [55]. We thus compute the principal curvature of each point of $\mathbf{P}$ and down-sample by retaining the S points with the minimum principal curvature values. We have selected $S = 1024$ (see Section 6).

To compute the principal curvature λ_i of a point $p_i \in \mathbf{P}$, the neighborhood covariance matrix $\mathbf{C_i}$ is first computed and then Eq. (6) is resolved with respect to scalar λ_i (eigenvalue of $\mathbf{C_i}$) and matrix $\mathbf{u}$ (eigenvectors of $\mathbf{C_i}$) [56]:

$$\mathbf{C_i}\mathbf{u} = \lambda_i \mathbf{u} \tag{6}$$

The symmetric 3×3 covariance matrix $\mathbf{C_i}$ of a point p_i is calculated based on its local neighborhood of κ nearest points $q_j, j = 1, 2, \ldots, \kappa$:

$$\mathbf{C_i} = \frac{1}{\kappa} \sum_{j=1}^{\kappa} \begin{bmatrix} q_j^x q_j^x & q_j^x q_j^y & q_j^x q_j^z \\ q_j^y q_j^x & q_j^y q_j^y & q_j^y q_j^z \\ q_j^z q_j^x & q_j^z q_j^y & q_j^z q_j^z \end{bmatrix} \tag{7}$$

where q_j^x, q_j^y, q_j^z correspond to the x,y and z coordinates of neighboring point q_j respectively.

Eigenvectors $\mathbf{u}$ represent the principal axes of the neighborhood:

$$\mathbf{u} = \begin{bmatrix} A_x & A_y & A_z \\ B_x & B_y & B_z \\ C_x & C_y & C_z \end{bmatrix} \tag{8}$$

and their eigenvalues λ are:

$$\lambda = \begin{bmatrix} \lambda_A & 0 & 0 \\ 0 & \lambda_B & 0 \\ 0 & 0 & \lambda_C \end{bmatrix} \tag{9}$$

Then the principal curvature λ_i of p_i is:

$$\lambda_i = \frac{min(\lambda_A, \lambda_B, \lambda_C)}{\lambda_A + \lambda_B + \lambda_C} \tag{10}$$

Feature extraction (RRI): We adopt the RRI (rigorous rotation invariant) descriptors for the point cloud [57] which creates features that remain fixed under different orientations. For each point $p_i \in \mathbf{P}$, the RRI module searches for its K-nearest neighbors and constructs a $K-NN$ graph. Then a combination of distance, angle, sin and cos features are computed for p_i based on the local neighborhood of the $K-NN$ graph.

Thus, the outcome of the RRI module is a feature matrix $\mathbf{F} = \{f_i \in \mathbb{R}^D |\ i = 1, 2, \ldots, S\}$ of dimension $S \times D$, where $D = 4 * K$ (K neighbors with 4 features each). The features $\mathbf{F}$ are then combined with the points $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, S\}$ that resulted from down-sampling and the concatenated matrix of dimension $S \times (3 + D)$ is output to the next stage.

Model segmentation (KPConv): We next estimate point-to-component correspondences, by segmenting each point cloud with the KPConv network [58]. We chose the deformable KPConv (KP-FCNN) presented in the same work, as our segmentation backbone for its good performance in learning local shifts effectively by deforming the convolution kernels to make them fit into the point cloud.

Given the down-sampled point cloud $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, S\}$ and its D corresponding features at each point $\mathbf{F} = \{f_i \in \mathbb{R}^D |\ i = 1, 2, \ldots, S\}$, the convolution of a kernel g at a point $x \in \mathbb{R}^3$ is defined as:

$$g(x, \mathbf{P}, \mathbf{F}) = \sum_{x_i \in N_x} g(x_i - x) f_i \tag{11}$$

where $N_x = \{x_i \in \mathbf{P} |\ \|x_i - x\| \leqslant r \in \mathbb{R}\}$, is the radius neighborhood of point x [59]. This neighborhood creates a sphere $\mathcal{S}_r^3$ around the point of interest x, and K kernels are spread in this sphere. Let $\{\tilde{x}_k |\ k = 1, ., K\} \subset \mathcal{S}_r^3$ be the kernel points and $\{W_k |\ k = 1, ., K\}$ be their associated weight matrices; then the kernel g can be defined in association with the linear correlation h between the kernel points $\tilde{x}_k$ and any point $(x_i - x)$ of sphere $\mathcal{S}_r^3$, as:

$$g(x_i - x) = \sum_{k=1}^{K} h(x_i - x, \tilde{x}_k)\ W_k \tag{12}$$

where

$$h(x_i - x, \tilde{x}_k) = \max\left(0, 1 - \frac{\|(x_i - x) - \tilde{x}_k\|}{\sigma}\right) \tag{13}$$

and σ is the influence distance between the kernel point and the selected point of the sphere that is related on the input density.

Combining equations Eqs. (12) and (11) we get the standard KPConv layer:

$$g(x, \mathbf{P}, \mathbf{F}) = \sum_{x_i \in N_x} \left(\sum_{k=1}^{K} h(x_i - x, \tilde{x}_k)\ W_k \right) f_i \tag{14}$$

Even though the standard KPConv produces sufficiently good results, we concluded that the deformable KPConv [58] suits the cross-time registration even better, because the network learns the kernel point positions. Instead of defining the kernel g on the kernel points $\tilde{x}_k$, the network generates a set of K shifts $\Delta(x)$ for every point $x \in \mathbb{R}^3$. Then the deformable KPConv layer is defined as:

$$g(x, \mathbf{P}, \mathbf{F}) = \sum_{x_i \in N_x} \left(\sum_{k=1}^{K} h(x_i - x, \tilde{x}_k + \Delta(x))\ W_k \right) f_i \tag{15}$$

The KPConv module estimates the point-to-component correspondences of both source and target point clouds, essentially performing a segmentation. The registration is done by the GMM-based DeepGMR module, which learns a consistent GMM representation of J components in order to recover the optimal transformation between the segmented point clouds. Given the desired number of segmentation components J, KPConv produces a respective segmentation of the input points in the form of an $S \times J$ association matrix $\mathbf{\Gamma} = \{\gamma_{ij}\}$ whose elements represent the probability of a point p_i belonging to a component $j \in J$. These J components are used to express the point cloud as a Gaussian Mixture Model (GMM) of J Gaussian distributions.

Final alignment (DeepGMR): The association matrix $\mathbf{\Gamma}$, representing the point-to-component correspondence, is used to estimate the transformation matrix $\mathbf{T}$ that aligns $\mathbf{P}$ and $\widehat{\mathbf{P}}$. To this end, we employ the optimization module of the DeepGMR network [45], where two differentiable blocks M_Θ and M_T are used to calculate the Gaussian mixture model (GMM) parameters from the association matrix $\mathbf{\Gamma}$ and transformation matrix $\mathbf{T}$ respectively.

M_Θ block converts the given point cloud $\mathbf{P} = \{p_i \mid i = 1, ., S\}$ and its association matrix $\mathbf{\Gamma} = \{\gamma_{ij} \mid i = 1, \ldots, S\ \&\ j = 1, \ldots, J\}$ to GMM parameters Θ as:

$$\Theta_j = (\pi_j, \mu_j, \Sigma_j) \tag{16}$$

where: $\pi_j = \frac{1}{S} \sum_{i=1}^{S} \gamma_{ij}$ is a scalar mixture weight, μ_j is the mean vector and $\mathbf{\Sigma}$ the covariance matrix of the j-th component, computed by solving the equations:

$$S\pi_j \mu_j = \sum_{i=1}^{S} \gamma_{ij} p_i \tag{17}$$

$$S\pi_j \Sigma_j = \sum_{i=1}^{S} \gamma_{ij} (p_i - \mu_j)(p_i - \mu_j)^\top \tag{18}$$

Finally, the transformation matrix $\mathbf{T}^* = (\mathbf{R}, \mathbf{t})$ is computed by block M_T, which tries to minimize the KL-divergence between the transformed GMM parameters Θ of the source and the GMMs $\widehat{\Theta}$ of the target:

$$\mathbf{T}^* = \arg\min_T KL(T(\widehat{\Theta}) |\ \Theta) = \arg\min_T \sum_{j=1}^{J} \frac{\hat{\pi}_j}{\sigma_j^2} \|T(\hat{\mu}_j) - \mu_j\|^2 \tag{19}$$

where $\Sigma_j = diag([\sigma_j^2, \sigma_j^2, \sigma_j^2])$ due to the fact that the covariances are chosen to be isotropic. This computes the alignment of the components' centroids instead of the alignment of the point clouds themselves.

The loss function of the DeepGMR module is back-propagated to the KPConv module in order to fine-tune its parameters with respect to the segmentation into the desired J components.

Loss function: The training objective of the loss function is to minimize the registration error. Many previous methods try to minimize the actual distance between the corresponding points in source and target point clouds [41,60], but in the case of cross-time registration this may not be ideal. We employ the directed Hausdorff distance, which has been proposed before [51] as a suitable metric for erosion. Given the ground truth transformation $\mathbf{T_{ideal}} = (\mathbf{R}, \mathbf{t})_{ideal}$ that aligns the source $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ to the target $\widehat{\mathbf{P}} = \{\widehat{p_j} \in \mathbb{R}^3 |\ j = 1, 2, \ldots, M\}$ and the predicted transformation $\mathbf{T} = (\mathbf{R}, \mathbf{t})$ that CrossTimeReg estimates, the loss function that we aim to minimize is:

$$L = \sqrt{D_H + D_{MH}} \tag{20}$$

where D_H is the standard Hausdorff distance calculated as the maximum of the directed Hausdorff distances D_h,where $D_h =$

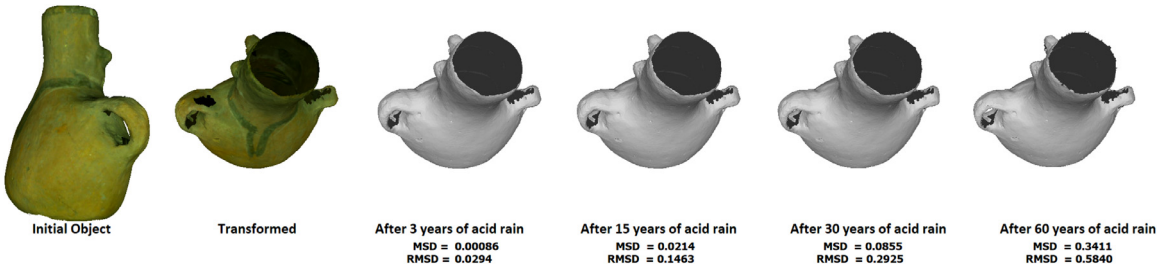

Fig. 2. The steps of the ECHO dataset creation for one object. The object is initially transformed and then the erosion simulator runs for 20 epochs of 3 years each. In this example, the initial model is shown degraded due to the effect of acid rain after 3, 15, 30 and 60 years. Below each step the point-wise MSD (Eq. (26)) and RMSD from the transformed model are given.

$\max_i(\min_j \|p_i - \widehat{p}_j\|)$:

$$D_H = \max\left(D_h(\mathbf{P}, \widehat{\mathbf{P}}),\ D_h(\widehat{\mathbf{P}}, \mathbf{P})\right) \tag{21}$$

and D_{MH} is the average directed Hausdorff distance:

$$D_{MH} = \frac{1}{N}\sum_{i=1}^{N} \min_j(\|p_i - \widehat{p}_j\|) \tag{22}$$

The average directed Hausdorff distance denotes the mean value of the minimum Euclidean distances $\|p_i - \widehat{p}_j\|$ between the initial source point cloud and the eroded target point cloud.

5. ECHO: a dataset of Eroded Cultural Heritage Objects

To the best of our knowledge, there is no publicly available dataset with ground truth for cross-time 3D registration. In order to benchmark and train cross-time 3D registration algorithms, we propose the ECHO dataset. Starting from a publicly available dataset of CH objects (see Section 5.1) we first applied a random rotation and translation $(\mathbf{R}, \mathbf{t})$ to the objects; then we used an artificial erosion process to erode the transformed objects. Since erosion is done in situ and the $(\mathbf{R}, \mathbf{t})$ parameters are known, we have the ground truth for benchmarking cross-time registration algorithms. The process is outlined in Fig. 2.

The ECHO dataset consists of three main parts, the original dataset, the transformed objects and the transformed-eroded objects. All three parts of the dataset along with the steps performed are explained thoroughly in the following subsections.

ECHO will be made publicly available with this paper.

5.1. Initial CH dataset

As a cornerstone, we selected the freely available *SHREC 2021: Retrieval of Cultural Heritage Objects* dataset [15] hereafter called SHREC2021. SHREC2021 dataset consists of 1575 3D scans of CH objects from pre-Columbian cultures captured in the Josefina Ramos de Cox museum in Lima, Peru. The SHREC2021 dataset is separated into two sub-datasets, considering two aspects, the shape and the culture. Each of the datasets is also divided into a collection set (70% of the dataset) and a query set (30% of the dataset) that can be used for training and testing respectively. The dataset regarding shape (referred as *datasetShape*) consists of 938 objects, 661 objects for training and 277 for testing. The other dataset is related to the retrieval-by-culture challenge of the SHREC competition, thus we will refer to it as *datasetCulture*. This dataset consists of 637 objects, 448 objects for training and 189 for testing. The objects of both sub-datasets are 3D meshes in.OBJ format, each consists of nearly 40,000 triangles, and they have been preprocessed so as to be centered in the origin of 3D space and with the up direction along the Y-axis. Figs. 3 and 4 show examples from SHREC2021.

Fig. 3. Original CH objects from SHREC2021 datasetShape.

Fig. 4. Original CH objects from SHREC2021 datasetCulture.

5.2. Building the ECHO dataset

Random transformation As a first step, we generated a variation of the initial dataset by applying a randomly calculated rigid 3D transformation; each object of the SHREC2021 dataset has been randomly rotated and translated. The rotation parameters were unrestricted while the translation vector was restricted to a maximum limit of 30 cm. The latter was decided based on the size of the objects. Fig. 5 shows examples of the initial objects along with their transformed instances. This dataset can of course be used as is for evaluating regular 3D registration algorithms. However, we extended it as per the next Section, in order to assess cross-time registration algorithms specifically.

Introduction of erosion Erosion due to atmospheric factors can affect the physiology of the object, resulting in alteration of its small-scale features that can challenge the registration process. We extended the aforementioned dataset by providing an eroded

Fig. 5. Original CH objects from SHREC2021 datasetShape (left), along with their transformed instances (middle). On the (right), a combination of the original and the transformed object is shown in order to demonstrate the translation value.

dataset of the transformed objects. The eroded set represents the erosion/degradation phenomenon that an object faces when exposed to the outdoor environment. Without loss of generality, we focus on chemical weathering of carbonate stone, i.e. the process that carbonate stone objects undergo when exposed to weather and especially to common atmospheric chemicals, such as carbon dioxide (CO_2) and nitrogen dioxide (NO_2).

The exact physico-chemical composition of the original objects of our source CH dataset is not known. Since our aim is the training and benchmarking of cross-time registration algorithms, rather than the simulation of realistic erosion for the objects' specific material, we have assumed that they consist of carbonate stone (a common material for CH objects, e.g. around the Mediterranean region) and applied weathering models that were available to us for this material (see below). Note that weathering models for other materials are not commonly available and, to the best of our knowledge, no other large publicly available dataset of eroding CH objects exists, for any type of material, that is suitable for training and testing deep networks (i.e. contains ground truth).

To this end, we adapted the simulation algorithm for the erosion of carbonate stone and marble presented in [51]. The simulator estimates the degradation of homogeneous marble or carbonate stone objects after their uniform exposure (spatially and temporally) to environmental conditions of polluted areas. We used the cases of chemical weathering in polluted atmosphere regions and the interaction of sulfur dioxide (SO_2), nitrogen dioxide (NO_2) and carbon dioxide (CO_2) with the material of the object. Specifically, we considered the effects of dry deposition of crust due to pollution and the recession by acid rain, which can result in gain or loss of material on the surface of an object. Dry deposition indicates the reaction of the material with SO_2 and NO_2 and manifests itself by the creation of crust upon the object's surface due to the

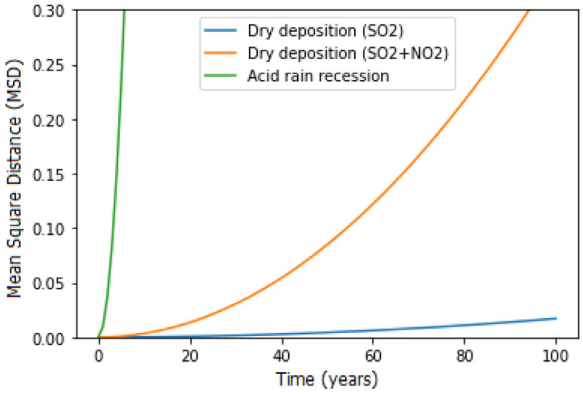

Fig. 6. Point-wise MSD between the initial and eroded carbonate stone objects over a period of 0–100 years, for different weathering cases.

transfer of chemical compounds from polluted industrial environments [61]. Recession by acid rain is the effect of surface loss of the object mainly due to its reaction with water and SO_2, NO_2 and CO_2 [62].

These weathering processes describe the change of the surface geometry and can be formulated as follows. Assuming that the initial object is modelled as a set of 3D points $\mathbf{P} = \{p_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ and $\mathbf{n} = \{n_i \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ are the normals per 3D point, the deposition/recession process relies on a computational and chemical model. The model can be formulated as a uniform offsetting procedure based on the diffusion equation:

$$\frac{\partial \mathbf{P}}{\partial t} = \mu \nabla^2 \mathbf{P} = \delta\ \mathbf{n} \tag{23}$$

so the target eroded surface $\widehat{\mathbf{P}} = \{\widehat{p_j} \in \mathbb{R}^3 |\ j = 1, 2, \ldots, N\}$ is then calculated as

$$\widehat{\mathbf{P}} = \mathbf{P} + \delta\ \mathbf{n}\ dt \tag{24}$$

which becomes:

$$\widehat{p_j} = p_i + \delta_i\ n_i\ dt \tag{25}$$

where n_i is the normal vector of point p_i, δ_i is the surface alteration at this point as predicted by the erosion model and dt is the time interval that the change is occurred. The above computation can be repeated for a number of epochs. Each epoch consists of time intervals of dt, where the environmental conditions are simulated. At the end of each epoch a new eroded surface is produced. The final surface produced after the total number of epochs reflects the changes that the initial surface faced when exposed to weathering conditions. If $\delta_i > 0$, the process simulates the surface recession due to dry deposition and when $\delta_i < 0$ it simulates the recession due to acid rain at a specific point i. The surface alteration offset δ derives from modeling the chemical processes according to the weathering models, described in [51,61–64].

In order to quantify the degradation that the CH objects experience under the above chemical models, we computed the Mean Square Distance (MSD) between each initial object and its eroded counterpart over a period of 0–100 years. Let $\mathbf{P_0} = \{p_{0i} \in \mathbb{R}^3 |\ i = 1, 2, \ldots, N\}$ be the original transformed point cloud and $\widehat{\mathbf{P_e}} = \{\widehat{p_{ej}} \in \mathbb{R}^3 |\ j = 1, 2, \ldots, M\}$ be the eroded point cloud after e years of erosion ($\widehat{\mathbf{P_e}}$ has level of erosion = e); then the MSD at level e is calculated as:

$$\text{MSD} = \frac{1}{N} \sum_{i=1}^{N} \| p_{0i} - \widehat{p_{ej}} \|^2 \tag{26}$$

where $\widehat{p_{ej}}$ is the nearest neighbor of p_{0i}.

Fig. 6 shows how considerably high is the degradation due to acid rain, compared to the respective degradation due to crust.

Fig. 7. Original CH objects from SHREC2021 datasetShape (left), along with their transformed instances (middle). The transformed object after application of acid rain erosion simulation is shown on the (right).

The erosion simulation is performed on each transformed object, for the time interval of 60 years, divided into 20 epochs (of 3 years each). During the simulation, the object's rigid parameters do not change, so we can argue that the ground truth random transformation matrix still holds.

Fig. 7 shows examples of the three main steps of the creation of ECHO dataset, the original object, the transformed one and the eroded instance.

6. Experiments

Our experiments are divided into five parts. First, we evaluate the proposed registration algorithm against the relevant state-of-the-art methods using the proposed ECHO dataset for the challenging problem of cross-time registration. Second, we compare the methods across multiple levels of erosion on ECHO. Third, we evaluate them on the task of traditional registration on two datasets; the ECHO dataset (containing only random rigid transformations, no erosion) and the SHREC2016 dataset [65]. Fourth, we evaluate CrossTimeReg on real erosion data, by performing cross-time registration on data from the PRESIOUS project [16]; these data are derived from erosion accelerators that simulate acid rain, salt and freeze-thaw effects on marble and soapstone slabs. Fifth, we attempt to measure the contribution of the proposed curvature downsampling on the cross-time registration task.

We compare against both geometry-based and deep-learning, local and global, as well as correspondence-based and non-correspondence-based algorithms. Regarding geometry-based methods, we compare against ICP [19], RANSAC [21] and Fast Global Registration (FGR) [22]. For RANSAC and FGR, we evaluated several variants with different feature extraction methods: FPFH [27], Spin Images (SI) [28], SISI, LD-SIFT [30], and RICI [29]. We tested each feature descriptor, as a pre-process step for both RANSAC and FGR algorithms. However, we kept and present the combinations that gave the best result in the Recall_{α} metric. Regarding deep-learning methods, we evaluated PRNet [48], PointNetLK [39], PCRNet [41], RPM-Net [43], DCP [42] and DeepGMR [45]. For ICP, RANSAC, FGR and FPFH we used the python implementations from the Open3D library [66], while for SISI, LD-SIFT and RICI we adapted the code bases released by the authors, which were implemented in MATLAB and C++. For the deep-learning methods, we used the pre-trained models provided by the open-source library Learning3D [67]. To ensure a fair comparison, all deep learning methods (including the proposed one) have been trained on the ModelNet dataset [68]. We have trained the complete CrossTimeReg pipeline using the ModelNet dataset with the annotated data provided in [45]. The first 20 classes of ModelNet have been used, as only those have been annotated by the authors (the rest were used for testing). Random translations and rotations are generated on the fly during the training/validation process for each annotated input point cloud of the ModelNet dataset. Based on the ablation study [45] regarding the ideal number J of Gaussian distributions, we use $J = 16$ for all experiments.

All tests were run on a PC with an i7-7700K CPU at 4.20 GHz, NVIDIA GeForce GTX 1080 Ti GPU and 32 GB of RAM.

6.1. Evaluation metrics

The rotation and translation errors are the absolute errors in Euler angles and translation vectors with respect to the ground truth. Ideally, both should be zero:

If $\mathbf{T}_{GT} = (\mathbf{R}_{GT}, \mathbf{t}_{GT})$ and $\mathbf{T}_{pred} = (\mathbf{R}_{pred}, \mathbf{t}_{pred})$ are the ground truth and predicted transformations respectively, the rotation and translation errors are measured as:

$$Error(R) = \|\mathbf{I} - \mathbf{R}_{GT}^{-1}\ \mathbf{R}_{pred}\| \tag{27}$$

$$Error(t) = \|\mathbf{t}_{GT} - \mathbf{t}_{pred}\| \tag{28}$$

where $\mathbf{I}$ is the identity matrix.

We next measure the *root mean square error (RMSE)* in Euclidean space against the ground truth solution. For the case of cross time registration, it is not sufficient to consider the registration error between the transformed source and the target, since there may not exist exact correspondence between them. Further, the commonly eroded surfaces of the objects may erroneously be measured as registration error, even though they represent the actual degradation of material. We thus measure the effect of the predicted transformation $\mathbf{T}_{pred}$ against the ground truth transformation $\mathbf{T}_{GT}$ on the source object based on [69]:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|(\mathbf{T}_{pred}\, p_i - \mathbf{T}_{GT}\, p_i)\|^2} \tag{29}$$

where N is the total points of the source object.

Moreover, following [12,70], we use the *root mean square distance (RMSD)* metric as a distance function employing Euclidean distance. It measures the similarity across the post-registration point cloud and the target point cloud (ground truth). This metric often appears in the literature as RMSE, but we decided to differentiate it from the aforementioned RMSE of Eq. (29) in order to highlight the difference of measuring the distance between the target and transformed point clouds from the error based only on the ground truth transformation. This results from the observation that the source and the target are not the same or parts of the same point cloud. The target object is eroded, which means that even if we perform the ground truth transformation on the source object, the result will not coincide with the target object. Thus, the RMSD which measures the distance between the point clouds, will not present the real registration success or error. We estimate RMSD as:

$$\text{RMSD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|(\mathbf{T}_{pred}\, p_i - \widehat{p}_i)\|^2} \tag{30}$$

Table 1
Registration results on the ECHO dataset when only random rotations, translations and 60 years of erosion are performed on the initial objects. The metrics evaluated are rotation error, Error(R), translation error, Error(t), root mean square error, RMSE, root mean square distance, RMSD and Recall with threshold 0.2. Bold and *dark gray* denote best and second best performing methods for each measure respectively. For fairness reasons, we have not included in bold, cases where CrossTimeReg performs best when trained on the training partition of ECHO; instead such cases are in bold italics.

	Method	Registration		Error(R)	Error(t)	RMSE	RMSD	Recall$_\alpha$(%)	Mean Exec. Time (sec)
		Local	Global						
Geometry-based	ICP [19]	✓		1.6992	42.5667	38.6065	42.583	0	34
	FPFH-RANSAC [21,27]		✓	1.8314	29.2151	29.3316	29.2326	0	32
	SI-FGR [22,28]		✓	1.8202	0.0629	1.1298	1.1344	21.91	32
	SISI-RANSAC [21,30]		✓	0.9984	0.1044	0.6870	0.6877	96.88	67
	LD-SIFT -RANSAC [21,30]		✓	**0.3496**	0.0793	**0.2789**	**0.2878**	98.79	68
	RICI-FGR [22,29]		✓	1.1396	*0.0495*	1.1832	1.1396	20.77	38
Deep Learning	PRNet [48]	✓		1.7514	1.0184	1.4723	1.4858	43.12	14
	PointNetLK [39]	✓		1.7413	29.2389	29.2514	29.2561	0	11
	PCRNet[41]		✓	1.8095	49.3442	49.3603	49.3600	0	10
	RPM-Net [43]		✓	1.6993	29.2594	29.2784	29.2755	0	15
	DCP [42]		✓	1.6881	38.6109	38.6542	38.6133	0	15
	DeepGMR [45]		✓	1.0065	0.0673	0.9454	*0.6746*	*99.31*	4
	CrossTimeReg		✓	*0.9942*	**0.0448**	*0.6764*	0.6812	**99.55**	6
	CrossTimeReg (trained on ECHO)		✓	***0.1397***	0.0714	***0.2606***	0.6928	***99.98***	6

Since exact point correspondences do not exist in cross-time registration, we approximate the computation using the nearest neighbor $\widehat{p}_i$ of the respective point.

Finally, we compute the success rate across the dataset $recall_\alpha$, i.e. the percentage of tests for which the RMSE is below a certain threshold α:

$$recall_\alpha = \frac{|S_\alpha|}{|S|} \times 100\% \tag{31}$$

where $|S|$ is the total number of tests performed and $|S_\alpha|$ is the number of tests that achieve RMSE less that the threshold α.

In previous literature, more metrics have been proposed and used to evaluate registration techniques, such as Chamfer distance or Earth Mover's Distance [71]. However, these metrics are less robust and have the same problem as RMSD in the case of erosion, i.e. they do not take into consideration the common erosion that may have occurred on all points of the surface. They have thus not been considered further.

6.2. Experimental results and analysis

Synthetic data - ECHO dataset In Table 1 we summarize the quantitative registration results on the challenging ECHO dataset for cross-time 3D registration; Fig. 8 illustrates some qualitative results. CrossTimeReg generally outperforms the state-of-the-art under most performance metrics.

Since cross-time registration involves point clouds with non exact point-level correspondences, methods like FPFH and DCP fail to converge in every run of this experiment. In addition, the initial poses of corresponding objects are generally far apart, both in terms of translation and rotation, and thus local methods like ICP, PointNetLK and PRNet fail to converge for many objects.

The performance of geometry-based global registration methods RANSAC and FGR rely on feature matching or keypoint detection from hand-crafted descriptors. Such descriptors face unusual challenges in the case of eroded objects. When SI is used as the local descriptor, its instability in the presence of noise and non-uniform sampling of the object's surface, result in many failed registrations. SISI and RICI, being derivatives of the SI, face similar challenges. RICI fails to properly identify the keypoints across the source and the target because it counts the intersections of homocentric circles with the surface. A target object which is evenly eroded produces different intersections to the corresponding source object. LD-SIFT, being both rotation and scale invariant, performs considerably better than the rest of the state-of-the-art; since erosion may affect the surface of the object evenly, the scale invariant features result in better recovery of the correct transformation. However, in terms of translation, the errors are larger and this is reflected in the Recall$_\alpha$ metric which is not as good as that of CrossTimeReg. A significant disadvantage of LD-SIFT is the large computation time and memory requirements which precludes its use in real time applications and on large scale datasets.

Interestingly, most deep learning methods perform significantly worse on the cross-time registration problem than geometry-based methods. This can be due to the fact that the networks have been trained on a different dataset and task than the related test ones. As mentioned before, to ensure a fair comparison, all deep learning methods have been trained on the ModelNet dataset for the traditional 3D registration problem. Thus, methods like PointNetLK that are trained on feature detection for specific object categories, fail to recognize useful features in different objects categories, like the pots and figurines of cultural heritage datasets. The generalization to unseen data, unrestricted rotation and significant translation results in poor performance for many deep learning methods. However, PRNet, DeepGMR and CrossTimeReg seem to overcome this obstacle and produce accurate registration results. The fact that PRNet was designed to perform partial-to-partial registration, can explain why the method converges on the cross-time registration problem. Cross-time registration shares a lot of common with the partial-to-partial registration, since the source and the target may have different surfaces but share common parts of their geometry. Still, PRNet is a local method and does not converge under large transformations.

Both DeepGMR and CrossTimeReg learn latent correspondences between the point clouds and GMM components, which are pose-invariant. Thus, the registration result is invariant to the magnitude of transformation or the density of the input geometries. However, DeepGMR estimates the correspondence between all points and all components in the latent GMM, meaning that its performance degrades when the point clouds partially overlap or if the points of the source and the target point clouds have been shuffled and randomly sampled. CrossTimeReg overcomes this drawback with the addition of the curvature based sub-sampling step. Moreover, with the addition of the KPConv network, CrossTimeReg learns local shifts effectively, implying that it learns the erosion part of the cross-time registration. The CrossTimeReg model has been trained in the same dataset as the rest of the deep

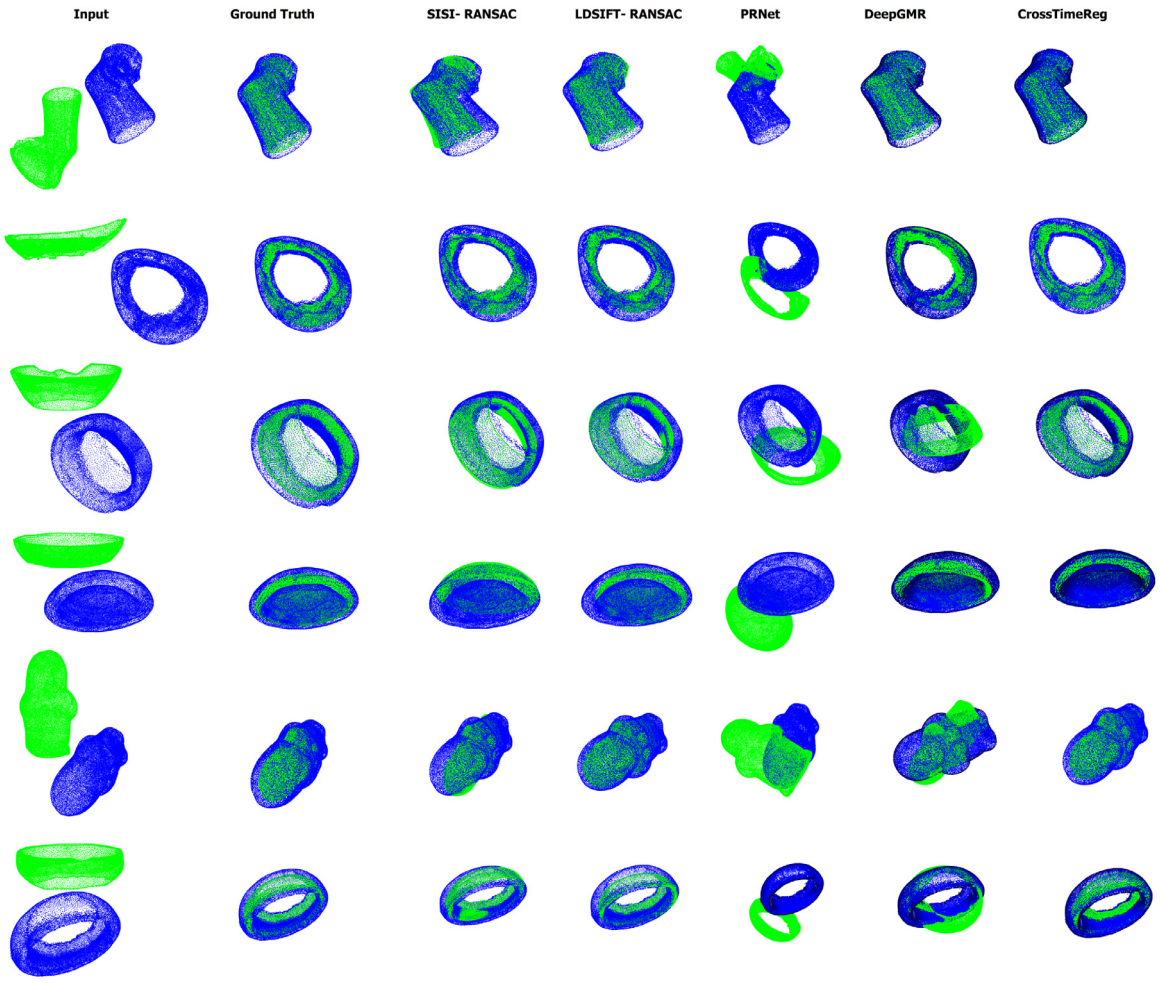

Fig. 8. Comparison between different registration methods on examples from the ECHO dataset for cross-time registration. Methods with the highest recall rates (Recall_{α} > 40%) are included.

networks (ModelNet), so that it is fairly comparable against the state-of-the-art. In order to investigate the effect of using eroded models in training, we have also fine-tuned the CrossTimeReg model on the training partition of the ECHO dataset. As can be seen from the last line of Table 1, the performance increases spectacularly.

ECHO Dataset - Multiple levels of erosion: In order to detect how the registration methods perform on different levels of degradation, in this section we evaluate the registration methods on the ECHO dataset against different levels of erosion. We have performed experiments for 20 different levels of erosion; from 1 year up to 60 years. Fig. 9 shows the RMSE metric for the most accurate methods. For clarity of illustration, we have excluded the methods which had average RMSE greater than 20, for every erosion level. It can be deduced that most methods tend to perform worse as the level of erosion increases. This is because when there is no or small amount of degradation, the geometry of the objects remains basically the same, so the identified keypoints and subsequent registration are accurate enough. However, as degradation increases, the target shape differentiates more and more from the source shape and most traditional registration methods tend to lose accuracy. Across all levels of erosion, CrossTimeReg appears to have stable performance, which even increases slightly at the highest levels.

In Table 2 we summarize the quantitative registration results on the ECHO dataset when only random rotations and translations are performed (no erosion).

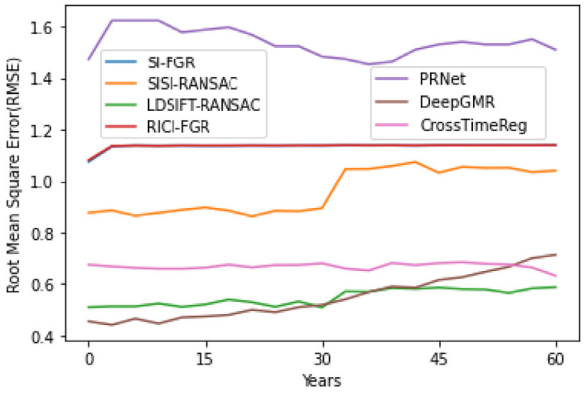

Fig. 9. Comparison between different registration methods on examples from the ECHO dataset for cross-time registration on different levels of erosion.

By comparing Tables 1 and 2 we can see that, relative to other methods, CrossTimeReg performs better when erosion is involved.

SHREC2016 dataset - traditional registration: In this section we evaluate the methods on traditional registration using the SHREC2016 dataset [65]. We chose this dataset, consisting of 383 models, as it is related to the cultural heritage domain. The 3D objects are pottery models originating from the Virtual Hampson Museum collection [72]. Again, we performed random rotations and

Table 2
Registration results on the ECHO dataset when only random rotations and translations are performed on the initial objects (no erosion). Bold and *dark gray* denote best and second best performing methods for each measure respectively.

	Method	Error(R)	Error(t)	RMSE	Recall$_a$(%)
Geometry-based	ICP [19]	1.6453	40.3423	30.0234	0
	FPFH-RANSAC [21,27]	1.8315	29.2325	29.4201	0.07
	SI-FGR [22,28]	1.7274	**0.0247**	1.0814	92.52
	SISI-RANSAC [21,30]	1.2945	0.2363	0.8774	85.89
	LD-SIFT -RANSAC [21,30]	**0.7021**	0.1661	**0.5102**	*99.01*
	RICI-FGR [22,29]	1.7392	*0.0272*	1.0814	93.73
Deep Learning	PRNet [48]	1.7368	0.9868	1.4728	49.35
	PointNetLK [39]	1.7346	29.2016	29.2192	0
	PCRNet[41]	1.8054	49.4641	49.4701	0
	RPM-Net [43]	1.6779	29.1860	29.2018	0
	DCP [42]	1.7219	39.7070	39.7200	0
	DeepGMR [45]	0.9578	1.9203	*0.5192*	98.34
	CrossTimeReg	*0.9456*	1.0821	0.6751	**99.43**

Table 3
Registration results on SHREC2016 dataset when only random rotations and translations are performed (no erosion). Bold and *dark gray* denote best and second best performing methods for each measure respectively.

	Method	Error(R)	Error(t)	RMSE	Recall$_a$(%)
Geometry-based	ICP [19]	1.2593	20.8497	91.60	9.14
	FPFH-RANSAC [21,27]	1.2295	18.2610	89.56	9.39
	SI-FGR [22,28]	*0.0062*	**0.0014**	**0.002**	*99.67*
	SISI-RANSAC [21,30]	0.0118	1.6920	3.40	99.47
	LD-SIFT -RANSAC [21,30]	**0.0021**	*0.0418*	**0.002**	**99.74**
	RICI-FGR [22,29]	0.0088	1.6819	2.66	98.47
Deep Learning	PRNet [48]	1.4985	71.6138	119.12	1.30
	PointNetLK [39]	1.4044	22.0351	100.32	7.57
	PCRNet[41]	1.3341	35.9334	98.40	2.61
	RPM-Net [43]	1.3670	98.0969	143.25	0.26
	DCP [42]	1.5797	123.5982	160.85	0
	DeepGMR [45]	0.7650	0.2756	5.47	98.56
	CrossTimeReg	0.0341	0.2328	*2.64*	98.13

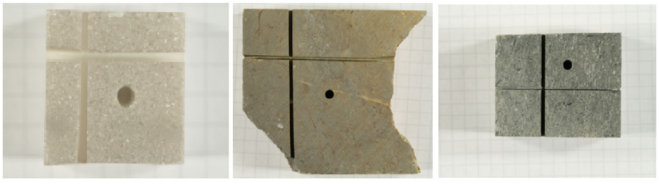

Fig. 10. Some of the stone slabs used in PRESIOUS accelerated erosion experiments [16]. The stones named Elefsis consist of pentelic marble, while stones names Nidaros consist of grytdal soapstone, representing the material of two monuments that were considered in the PRESIOUS project, the Demeter Sanctuary in Elefsis (Greece) and the Nidaros Cathedral in Trondheim (Norway) respectively.

translations to the objects (no erosion is present) and the results are shown in Table 3. The behaviour of the methods is similar to that on the ECHO dataset, a positive indication for the quality of ECHO; CrossTimeReg demonstrates high accuracy without achieving the top results. Interestingly, in both Tables 2 and 3 where no erosion is involved, CrossTimeReg is one of the top 2 deep learning methods, while non-learning methods perform extremely well.

PRESIOUS dataset - real erosion data from accelerated erosion experiments: To demonstrate how CrossTimeReg performs in the case of real erosion data, we employed data from the PRESIOUS EU project [16]. These data consist of three accelerated erosion experiments on two different types of stone slabs; pentelic marble and grytdal soapstone, see Fig. 10.

The erosion effects that were studied in the accelerated erosion experiments were acid rain weathering, salt and freeze-and-thaw effects. Table 4 gives details on the 3D scanned slabs across the erosion experiments. We tested CrossTimeReg and LD-SIFT with RANSAC, see Table 5. For each stone slab, we register the initial scan with the scan after the first period of erosion (Round 1 - Round 2), the scan after the first period with the final scan (Round 2 - Round 3) and the initial scan with the final scan after 2 periods of experiments (Round 1 - Round 3). Both methods have been run on the same hardware. CrossTimeReg's execution time increased with the number of object points, but this was only due to the curvature downsampling component; the execution of the rest of the modules of CrossTimeReg remained constant, irrespective of the number of object points. LD-SIFT had to be interrupted after 4 hours on the same data. To overcome this, we uniformly down-sampled the 3D models, so that the down-sampled meshes would contain 50K points. For this reason, we omitted the execution time of LD-SIFT in the above Table. Since, the dataset had no ground truth of the transformations performed on the objects, we measured the RMSD based on eq. (30), which measures also the distance due to the degradation of the objects. Table 5 and Fig. 11 show that CrossTimeReg behaves favourably compared to LD-SIFT on real data.

Evaluating curvature downsampling (CDS) for cross-time registration: In order to get a better intuition of the contribution of the proposed downsampling method on final performance, we carried out an ablation study on the ECHO dataset. The erosion level was varied from no erosion (only random rotations and translations) up to 60 years of erosion due to deposition of crust and acid rain. We compare against the case where uniform sampling, based on

Table 4
Details of the slabs used in the PRESIOUS dataset. Round 1 contains the initial scanned object, Round 2 after the first period of erosion and Round 3 after the second period of erosion.

Experiment	Material	Stone Slab Label	Round	# points
Freeze and Thaw	Pentelic Marble	Elefsis Large 01	1	998621
			2	790553
			3	847791
	Grytdal Soapstone	Nidaros Bad Large 01	1	1904088
			2	2671989
			3	2778924
Salt	Pentelic Marble	Elefsis Large 02	1	1236236
			2	1050038
			3	1336365
	Grytdal Soapstone	Nidaros Bad Large 02	1	2023069
			2	3978584
			3	4250544
Acid Rain	Pentelic Marble	Elefsis Large 03	1	983698
			2	976587
			3	612447
	Grytdal Soapstone	Nidaros Good Large 03	1	3009981
			2	2858613
			3	3130228

Table 5
Registration results on the PRESIOUS dataset of real erosion data from accelerated experiments. The best performance between LD-SIFT and CrossTimeReg is in bold. CrossTimeReg was run on the original datasets; LD-SIFT had not completed execution after 4h on the original datasets and was run on subsampled versions.

Exp.	Stone Slab Label	Rounds	LD-SIFT	Cross Time Reg	
			RANSAC [21,30] RMSD	RMSD	Exec.Time (sec)
Freeze and Thaw	Elefsis Large 01	1 - 2	0.03854	**0.03817**	250
		2 - 3	0.05358	**0.03858**	196
		1 - 3	0.05916	**0.03867**	264
	Nidaros Bad Large 01	1 - 2	0.05087	**0.03617**	450
		2 - 3	0.03605	**0.03132**	511
		1 - 3	0.03840	**0.03838**	461
Salt	Elefsis Large 02	1 - 2	0.04682	**0.03888**	252
		2 - 3	**0.03667**	0.03719	250
		1 - 3	0.04079	**0.03676**	239
	Nidaros Bad Large 02	1 - 2	0.03831	**0.03564**	663
		2 - 3	0.04702	**0.03793**	900
		1 - 3	0.03537	**0.03033**	699
Acid Rain	Elefsis Large 03	1 - 2	**0.03552**	0.03799	174
		2 - 3	0.04253	**0.03544**	118
		1 - 3	0.05479	**0.03667**	150
	Nidaros Good Large 03	1 - 2	**0.03896**	0.03931	331
		2 - 3	**0.03460**	0.03819	343
		1 - 3	0.03794	**0.03253**	406

voxel size, is used; in our experiments we considered voxel size = 0.05 (5 cm in metric scale) which gives the best results. We calculated the RMSE based on Eq. (29). As can be seen from Fig. 12, the proposed downsampling scheme behaves stably across levels of erosion. On the contrary, uniform downsampling has better RMSE results on small erosion values but increases with the level of erosion.

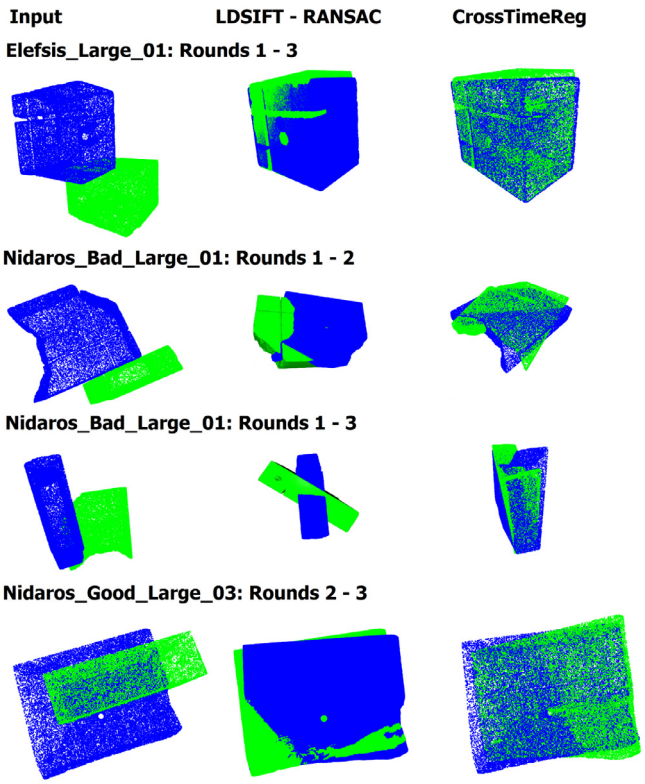

Fig. 11. Qualitative comparison between CrossTimeReg and LD-SIFT on examples from the PRESIOUS dataset of real eroded data.

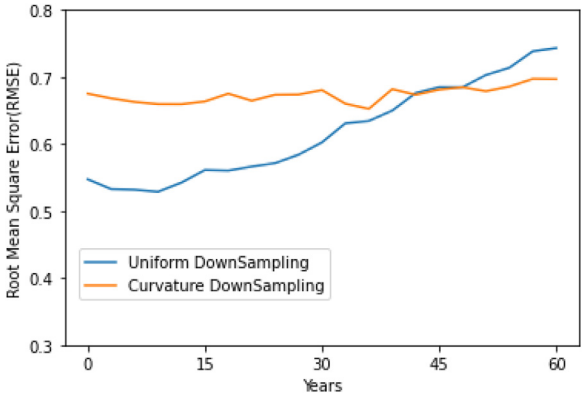

Fig. 12. Ablation study of downsampling methodologies on different levels of erosion on the ECHO dataset.

7. Conclusions and future work

The challenging problem of cross-time 3D registration has been defined and CrossTimeReg, a deep learning method for cross-time 3D registration, has been proposed. CrossTimeReg achieves state-of-art accuracy and robustness to large initial transformations while being computationally efficient. Indeed computational efficiency is a main advantage of the proposed method against previous very accurate geometry based methods. The proposed method is also very stable as the level of erosion increases. Its computational efficiency can be further optimized, especially the subsampling step. A new dataset, ECHO, has been created to facilitate the evaluation of techniques on cross-time registration with high quality models and ground truth. We anticipate that the public availability of ECHO will facilitate future experiments for cross-time related tasks (registration, retrieval, recognition). In ad-

dition, an implementation-based comparison of both deep learning and geometry-based object registration algorithms has been made, with some interesting observations.

As a future step, we consider to replace the hand-crafted features with features learned specifically for cross-time registration by the network. We plan to integrate our network into larger systems, for tasks such as monitoring and segmentation of changes that CH objects undergo and extend the framework to register and fuse multiple modalities, in addition to 3D point clouds. Moreover, we intend to study partiality in conjunction with degradation on CH objects.

If extreme erosion values are applied on surface meshes, then mesh folding can occur. This can also arise on boundary and fragile edges (e.g. Fig. 2). Mesh folding is a challenging but worthwhile problem to address (see for example [73,74]).

Another potential avenue of future work is to apply a non-uniform, more realistic erosion model in the simulation process, which could take into consideration the surface's orientation, shape and texture.

Credit Author Statement

Evdokia Saiti: Conceptualization; Data curation; Formal analysis; Investigation; Methodology; Validation; Visualization; Software; Writing - original draft. **Antonios Danelakis:** Conceptualization; Methodology; Validation; Visualization; Software; Writing review & editing. **Theoharis Theoharis:** Conceptualization; Funding acquisition; Project administration; Methodology; Resources; Supervision; Validation; Visualization; Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has received funding from the European Union's Horizon 2020 research and innovation program CHANGE under the Marie Skłodowska-Curie grant agreement No. 813789.

References

[1] Fonseca MJ, Ferreira A, Jorge JA. Towards 3D modeling using sketches and retrieval. Proceedings of the Eurographics workshop on sketch-based interfaces and modeling 2004. Citeseer; 2004. p. 127.
[2] Gojcic Z, Zhou C, Wegner JD, Guibas LJ, Birdal T. Learning multiview 3D point cloud registration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2020. p. 1759–69.
[3] Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. ACM Trans Graph (tog) 2019;38(5):1–12.
[4] Wu W, Qi Z, Fuxin L. Pointconv: deep convolutional networks on 3D point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2019. p. 9621–30.
[5] Kim P, Chen J, Cho YK. SLAM-driven robotic mapping and registration of 3D point clouds. Autom Constr 2018;89:38–48.
[6] Weinmann M, Leitloff J, Hoegner L, Jutzi B, Stilla U, Hinz S. Thermal 3D mapping for object detection in dynamic scenes. ISPRS Ann Photogramm Remote Sens Spat Inf Sci 2014;2(1):53.
[7] Kerl C, Sturm J, Cremers D. Dense visual SLAM for RGB-D cameras. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2013. p. 2100–6.
[8] Mellado N, Dellepiane M, Scopigno R. Relative scale estimation and 3D registration of multi-modal geometry using growing least squares. IEEE Trans Vis Comput Graph 2015;22(9):2160–73.
[9] Papaioannou G, Schreck T, Andreadis A, Mavridis P, Gregor R, Sipiran I, Vardis K. From reassembly to object completion: a complete systems pipeline. J Comput Cult Herit (JOCCH) 2017;10(2):1–22.
[10] Chang W, Zwicker M. Global registration of dynamic range scans for articulated model reconstruction. ACM Trans Graph (TOG) 2011;30(3):1–15.
[11] Zollhöfer M, Stotko P, Görlitz A, Theobalt C, Nießner M, Klein R, Kolb A. State of the art on 3D reconstruction with RGB-D cameras. In: Computer graphics forum, vol. 37. Wiley Online Library; 2018. p. 625–52.
[12] Saiti E, Theoharis T. An application independent review of multimodal 3D registration methods. Comput Graph 2020;91:153–78.
[13] Pintus R, Pal K, Yang Y, Weyrich T, Gobbetti E, Rushmeier HE. Geometric analysis in cultural heritage.. In: Proceedings of the GCH; 2014. p. 117–33.
[14] Saha S, Foryś P, Martusewicz J, Sitnik R. Approach to analysis the surface geometry change in cultural heritage objects. In: Proceedings of the international conference on image and signal processing. Springer; 2020. p. 3–13.
[15] SHREC 2021: retrieval of cultural heritage objects. http://www.ivan-sipiran.com/shrec2021.html; Accessed on March2021.
[16] PRESIOUS fp7-600533 eu project -finalevaluationreport. http://presious.eu/file_downloads/PRESIOUS-D5.8-FinalEvaluationReport.pdf, Accessed on May2021.
[17] Tam GK, Cheng Z-Q, Lai Y-K, Langbein FC, Liu Y, Marshall D, Martin RR, Sun X-F, Rosin PL. Registration of 3D point clouds and meshes: a survey from rigid to nonrigid. IEEE Trans Vis Comput Graph 2012;19(7):1199–1217.
[18] Tombari F, Remondino F. Feature-based automatic 3D registration for cultural heritage applications. In: Proceedings of the digital heritage international congress (DigitalHeritage), vol. 1. IEEE; 2013. p. 55–62.
[19] Besl PJ, McKay ND. Method for registration of 3D shapes. In: Sensor fusion IV: control paradigms and data structures, vol. 1611. International Society for Optics and Photonics; 1992. p. 586–606.
[20] Rusinkiewicz S, Levoy M. Efficient variants of the ICP algorithm. In: Proceedings of the third international conference on 3-D digital imaging and modeling. IEEE; 2001. p. 145–52.
[21] Fischler MA, Bolles RC. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 1981;24(6):381–95.
[22] Zhou Q-Y, Park J, Koltun V. Fast global registration. In: Proceedings of the European conference on computer vision. Springer; 2016. p. 766–82.
[23] Zhang Z, Dai Y, Sun J. Deep learning based point cloud registration: an overview. Virtual Real Intell Hardw 2020;2(3):222–46.
[24] Mavridis P, Andreadis A, Papaioannou G. Efficient sparse ICP. Comput Aided Geom Des 2015a;35:16–26.
[25] Han X.-F., Sun S.-J., Song X.-Y., Xiao G.-Q.. 3D Point cloud descriptors in hand-crafted and deep learning age: state-of-the-art. arXiv e-prints2018;:arXiv–1802.
[26] Diez Y, Roure F, Lladó X, Salvi J. A qualitative review on 3D coarse registration methods. ACM Comput Surv (CSUR) 2015;47(3):1–36.
[27] Rusu RB, Blodow N, Beetz M. Fast point feature histograms (FPFH) for 3D registration. In: Proceedings of the IEEE international conference on robotics and automation. IEEE; 2009. p. 3212–17.
[28] Johnson AE, Hebert M. Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Trans Pattern Anal Mach Intell 1999;21(5):433–49.
[29] van Blokland BI, Theoharis T. Radial intersection count image: a clutter resistant 3D shape descriptor. Comput Graph 2020;91:118–28.
[30] Darom T, Keller Y. Scale-invariant features for 3D mesh models. IEEE Trans Image Process 2012;21(5):2758–69.
[31] Lowe DG. Distinctive image features from scale-invariant keypoints. Int J Comput Vis 2004;60(2):91–110.
[32] Yang J, Li H, Campbell D, Jia Y. Go-ICP: a globally optimal solution to 3D ICP point-set registration. IEEE Trans Pattern Anal Mach Intell 2015;38(11):2241–54.
[33] Li H, Hartley R. The 3D-3D registration problem revisited. In: Proceedings of the IEEE 11th international conference on computer vision. IEEE; 2007. p. 1–8.
[34] Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B (Methodol) 1977;39(1):1–22.
[35] Granger S, Pennec X. Multi-scale EM-ICP: a fast and robust approach for surface registration. In: Proceedings of the European conference on computer vision. Springer; 2002. p. 418–32.
[36] Jian B, Vemuri BC. Robust point set registration using gaussian mixture models. IEEE Trans Pattern Anal Mach Intell 2010;33(8):1633–45.
[37] Eckart B, Kim K, Kautz J. HGMR: hierarchical gaussian mixtures for adaptive 3D registration. In: Proceedings of the European conference on computer vision (ECCV); 2018. p. 705–21.
[38] Qi CR, Su H, Mo K, Guibas LJ. Pointnet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 652–60.
[39] Aoki Y, Goforth H, Srivatsan RA, Lucey S. Pointnetlk: robust & efficient point cloud registration using pointnet. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2019. p. 7163–72.
[40] Lucas BD, Kanade T, et al. An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th international joint conference on artificial intelligence, vol. 2. Vancouver, British Columbia; 1981. p. 674–9.
[41] Sarode V., Li X., Goforth H., Aoki Y., Srivatsan R.A., Lucey S., Choset H.. Pcrnet: point cloud registration network using pointnet encoding. arXiv preprint arXiv:1908079062019;.
[42] Wang Y, Solomon JM. Deep closest point: Learning representations for point cloud registration. In: Proceedings of the IEEE/CVF international conference on computer vision; 2019a. p. 3523–32.
[43] Yew ZJ, Lee GH. Rpm-net: Robust point matching using learned features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2020. p. 11824–33.
[44] Gold S, Rangarajan A, Lu C-P, Pappu S, Mjolsness E. New algorithms for 2D and 3D point matching: pose estimation and correspondence. Pattern Recognit 1998;31(8):1019–31.

[45] Yuan W, Eckart B, Kim K, Jampani V, Fox D, Kautz J. Deepgmr: Learning latent gaussian mixture models for registration. In: Proceedings of the European conference on computer vision. Springer; 2020. p. 733–750.
[46] Mellado N, Aiger D, Mitra NJ. Super 4PCS fast global pointcloud registration via smart indexing. In: Computer graphics forum, vol. 33. Wiley Online Library; 2014. p. 205–15.
[47] Chen Y-W, Mimori A, Lin C-L. Hybrid particle swarm optimization for 3-d image registration. In: Proceedings of the 16th IEEE international conference on image processing (ICIP). IEEE; 2009. p. 1753–6.
[48] Wang Y., Solomon J.M.. Prnet: self-supervised learning for partial-to-partial registration. arXiv preprint arXiv:1910122402019b;.
[49] Mavridis P, Andreadis A, Papaioannou G. Fractured object reassembly via robust surface registration.. In: Proceedings of the Eurographics (Short Papers); 2015b. p. 21–4.
[50] Bromblet P, Vallet J, Verges-Belmin V. Illustrated glossary on stone deterioration patterns, 3. Monuments and sites; 2008.
[51] Perakis P, Schellewald C, Kebremariam KF, Theoharis T. Simulating erosion on cultural heritage monuments. Proceedings of the 20th international conference on cultural heritage and new technologies (CHNT20); 2015.
[52] Barbosa IB, Gebremariam KF, Perakis P, Schellewald C, Theoharis T. Establishing parameter values for the stone erosion process. In: Proceedings of the CAA2015; 2015. p. 347.
[53] Hameiri E, Shimshoni I. Estimating the principal curvatures and the Darboux frame from real 3-D range data. IEEE Trans Syst Man Cybern Part B (Cybern) 2003;33(4):626–37.
[54] Agapaki E, Nahangi M. Chapter 3 - scene understanding and model generation. In: Brilakis I, Haas C, editors. Infrastructure computer vision. Butterworth-Heinemann; 2020. p. 65–167. ISBN 978-0-12-815503-5, 10.1016/B978-0-12-815503-5.00003–6
[55] Yang Y-L, Lai Y-K, Hu S-M, Pottmann H, et al. Robust principal curvatures on multiple scales. In: Proceedings of the symposium on geometry processing; 2006. p. 223–6.
[56] Patrikalakis NM, Maekawa T. Shape interrogation for computer aided design and manufacturing. Springer Science & Business Media; 2009.
[57] Chen C, Li G, Xu R, Chen T, Wang M, Lin L. Clusternet: deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2019. p. 4994–5002.
[58] Thomas H, Qi CR, Deschaud J-E, Marcotegui B, Goulette F, Guibas LJ. Kpconv: flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision; 2019. p. 6411–20.
[59] Thomas H.. Rotation-invariant point convolution with multiple equivariant alignments. arXiv preprint arXiv:2012040482020;.
[60] Khazari AE, Que Y, Sung TL, Lee HJ. Deep global features for point cloud alignment. Sensors 2020;20(14):4032.
[61] Yerrapragada SS, Jaynes JH, Chirra SR, Gauri K. Rate of weathering of marble due to dry deposition of ambient sulfur and nitrogen dioxides. Anal Chem 1994;66(5):655–9.
[62] Baedecker PA, Reddy MM. The erosion of carbonate stone by acid rain: laboratory and field investigations. J Chem Educ 1993;70(2):104.
[63] Gauri KL, Bandyopadhyay JK. Carbonate stone: chemical behavior, durability and conservation; 1999.
[64] Yerrapragada SS, Chirra SR, Jaynes JH, Li S, Bandyopadhyay JK, Gauri K. Weathering rates of marble in laboratory and outdoor conditions. J Environ Eng 1996;122(9):856–63.
[65] Pratikakis I, Savelonas M, Arnaoutoglou F, Ioannakis G, Koutsoudis A, Theoharis T, Tran M, Nguyen V, Pham V, Nguyen H, et al. SHREC'16 Track: partial shape queries for 3D object retrieval Proceedings of the 3DOR, 1; 2016.
[66] Zhou Q.-Y., Park J., Koltun V.. Open3D: a modern library for 3D data processing. arXiv:1801098472018;.
[67] Learning3d: Learning3D: a modern library for deep learning on 3D point clouds data.https://github.com/vinits5/learning3d, Accessed on March2021.
[68] Wu Z, Song S, Khosla A, Yu F, Zhang L, Tang X, Xiao J. 3D shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 1912–20.
[69] Choi S, Zhou Q-Y, Koltun V. Robust reconstruction of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 5556–65.
[70] Yang J, Cao Z, Zhang Q. A fast and robust local descriptor for 3D point cloud registration. Inf Sci 2016;346:163–79.
[71] Fan H, Su H, Guibas LJ. A point set generation network for 3D object reconstruction from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 605–13.
[72] Virtual Hampson Museum collection. https://hampson.cast.uark.edu/, Accessed on May2021.
[73] Kim S-J, Lee D-Y, Yang M-Y. Offset triangular mesh using the multiple normal vectors of a vertex. Comput Aided Des Appl 2004;1(1–4):285–91.
[74] Chen Y, Wang CC. Uniform offsetting of polygonal model based on layered depth-normal images. Comput-Aided Des 2011;43(1):31–46.