Feature-based automatic 3D registration for cultural heritage applications

Federico Tombari
Computer Vision Lab
University of Bologna
Bologna, Italy
federico.tombari@unibo.it

Fabio Remondino,
3D Optical Metrology (3DOM) unit
Bruno Kessler Foundation (FBK)
Trento, Italy
remondino@fbk.eu

Abstract—This paper presents a review of the state-of-the-art techniques in the field of 3D invariant features for the automatic registration of point clouds and 3D meshes. The paper proposes also a multi-stage 3D registration pipeline implemented using the PCL libraries. Experiments are carried out on datasets related to heritage scenarios and addressing large-scale outdoor data acquisitions as well as small objects.

Keywords—Features, Matching, 3D, Point Cloud, PCL

I. INTRODUCTION

Many surveying and 3D modeling applications rely on features extraction as primary input for data alignment, segmentation and analysis. Feature points (or keypoints) are generally used for images registration, 3D reconstruction, motion tracking, robot navigation, object detection and recognition, point cloud alignment, etc. In the last years the detection of repeatable and distinctive image keypoints was the goal of many research activities [1]-[3]. Afterwards, due to the large diffusion of (low-cost) 3D sensors and the growing importance of applications based on 3D data (e.g. shape retrieval, object recognition, point cloud registration, etc.), the focus has shifted towards 3D shapes [4]-[6]. The aim is the detection of 3D features and similarities as well as the establishment of correspondences between surfaces or point clouds in order to align them in an automated and accurate way. But despite several methods have been recently proposed for 3D point clouds and surfaces registration (see Table 1 and [6][8] for reviews and comparisons), most of the algorithms employed for cultural heritage applications still rely on an interactive approach, where the user is required to guide the initial coarse alignment process by manually providing correspondences before running the final fine registration, generally based on an ICP-like algorithm [9]. Thus the goal of the paper is to review the use of automated techniques - based on 3D feature correspondences - for the automatic alignment / registration and reconstruction of 3D data (clouds or meshes) in the field of cultural heritage. The use of such approaches has the advantage of making the registration process more automatic, thus requiring less human supervision, at the same time without relying blindly on random ICP initializations. The use of feature correspondences indeed is able to provide a single, coarse initialization for a successive ICP-like refinement.

The following sections review the state-of-the-art in the field of 3D keypoint detection, together with 3D descriptors and feature matching. Moreover a multi-stage registration pipeline which can be used for registering point clouds and 3D meshes of various sizes is presented, showing that each step of the pipeline can be interchanged with alternative algorithms available in the literature. Experimental results from heritage objects are reported, showing examples of both pairwise and multi-view alignments.

Notably, several methods implementing these automated 3D data alignments are available in the Point Cloud Library (PCL, www.pointclouds.org), currently the reference open source library for 3D Robotic Perception and 3D Computer Vision applications and will be presented here.

II. REGISTRATION METHODS

The goal of a 3D data registration procedure is to determine the rigid body transformation parameters (3 rotations and 3 translations, thus assuming equal scale) between two (pairwise) or more (multi-view) partially overlapping clouds or 3D meshes. Normally the registration procedure consists of an initial coarse alignment and a successive refinement to achieve an accurate solution (Figure 1). The coarse registration starts with some correspondences between the data in order to compute an initial estimation of the motion between the 3D views. These correspondences can be extracted in various ways, automatically or manually. Then the fine registration minimizes a certain distance function being normally the distance between the correspondences or between a point in the

---

TABLE 1: Commonly used methods for coarse and fine registration of 3D data.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>COARSE REGISTRATION</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>FINE REGISTRATION</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Several registration methods (Table 1) were developed in the last years [6][8] and the most powerful approach which enables fully automated registrations with reasonable computational times is the feature-based one coupled with an ICP-like method (Figure 1). Automated coarse registration, recognition and segmentation techniques are mainly based on feature-based methods which normally consist of feature detection, feature description and correspondence estimation/rejection. Detectors are operators that search for interest points ("keypoints") which are distinctive, repeatable and geometrically stable under different transformations. On the other hand, (local) descriptors analyze the local neighborhood of a keypoint providing an invariant representation. This information can then be used to classify the extracted keypoints among specific categories or for determining point-to-point correspondences between two point clouds.

The most common approach for the fine alignment is represented by the Iterative Closest Point (ICP) algorithm and its successive variants [17]-[23]. The accurate parameters of the rigid body transformation are generally estimated by the use of closed-form solutions, mainly singular value decomposition (SVD) [26] and quaternion methods [27]. The closed-form solutions cannot fully consider the statistical point error models although methods that can model the anisotropic point errors were proposed [28][29]. ICP-like approaches, while able to provide a fine geometric alignment, have the problem of local minima, since they are based on greedy optimization. This means that, before running the ICP, the two datasets have to be coarsely aligned by providing an initial guess to the system. Conversely to most commercial software datasets have to be coarsely aligned by providing an initial user initialization (3 points), automatic 3D registration methods aims at substituting this step by a coarse registration determined through a feature matching stage, so to make the whole process completely automatic.

Fig. 1: Feature-based pipeline for the pairwise (coarse+fine) registration algorithm of point clouds or surfaces. The lower boxes describe the automated steps of the procedure.

### III. PAIRWISE REGISTRATION

For pairwise registrations, 3D features (or descriptors) are first extracted from both datasets (e.g. point clouds), then matched together, thus yielding point-to-point correspondences. Correspondences are successively pruned within a correspondences rejection stage in order to eliminate outliers. Those matches that have not been discarded are then used for the initial registration [30] that coarsely aligns one cloud to the other, thus allowing a faster convergence to the successive fine registration stage.

The next sections report the several stages composing the coarse registration process (and the implemented pipeline).

#### A. Keypoint Detection

The detection of salient interest points in 3D data is a recent research topic which is gaining a great momentum following the important achievements reached within image-based applications [2]. The goal of a 3D keypoint detector is to extract a subset of points out of a mesh or a point cloud which is distinctive and repeatable [16]. While distinctiveness is a property which is usually measured globally on the whole cloud or structure, repeatability is usually locally defined and brings in resilience to noise and disturbance factors, which is an important characteristic of the extracted keypoints. Although this stage is not mandatory in most 3D registration scenarios – i.e. it can be substituted by uniform/random sampling – it often helps increasing the reliability of the correspondences during the correspondence estimation stage.

Different 3D keypoint detectors were proposed in the research literature [16] and can be divided in fixed-scale (Figure 2) and adaptive-scale detectors (Figure 3). The latter explore a scale-space and associate, to each extracted keypoint, a characteristic scale which will be used as the support for the successive description stage. On the other hand fixed-scale descriptors detect salient keypoints only at a specific and fixed scale, which is an input parameter for the algorithm. In Figure 2b a qualitative comparison among several state-of-the-art fixed-scale 3D detectors over two range-based 3D models is presented. The evaluated detectors are: Harris3D [31] and its variant using “Tomasi” saliency as provided by the pcl_keypoints module within PCL, the Local Surface Patches (LSP) detector [32], the Intrinsic Shape Signatures (ISS) detector [5] and the fixed-scale version of the detector proposed in [33], named as KPQ in [16]. All detectors are run at the same scale, i.e. set to 6 times the mesh resolution. The evaluated detector have a remarkably different behavior: while certain approaches such as LSP and ISS extract keypoints more uniformly along the mesh, Harris3D (in both variants) and KPQ tend to focus more on depth discontinuities, avoiding to detect keypoints over ample locally planar surface regions.

#### B. 3D Local Descriptors

Following the stage of keypoint detection, each extracted point is associated with a compact representation of the 3D local neighborhood (or support) usually defined over a spherical or cylindrical volume centered in the keypoint [39][40][41]. If a keypoint extractor is not available, usually descriptors are computed over a set of randomly or uniformly sampled points of the mesh/cloud. 3D local descriptors should not be confused with 3D global descriptors [42]. Indeed, the former aim at obtaining a representation of the 3D local neighborhood of a keypoint, while the latter provide a description of the whole 3D shape/structure.
Fig. 2: The fixed-scale detector process (pruning keypoint selection + Non-Maxima Suppression (NMS) procedure based upon a saliency measure) to extract 3D keypoints from point clouds or meshes (a). Qualitative comparison among different 3D keypoint detectors extracted on two 3D models (b).

Fig. 3: The adaptive-scale detector process and methods to extract 3D keypoints from point clouds and meshes [16].

3D local descriptors are thus inherently more robust with respect to clutter and occlusions, and are in general more suitable to be deployed in the context of 3D point clouds registration with respect to global descriptors due to the presence of only partially overlapping surfaces.

A deep review with experimental evaluation among local descriptors is presented in [43]. It is worth pointing out that several implementations of the state of the art in terms of 3D local descriptors are available in PCL.

C. Correspondences Estimation and Rejection

The correspondences estimation stage takes two sets of descriptors - each computed on one of the two clouds being processed - and aims at determining point-to-point correspondences by matching the two sets. Matching is typically solved as a Nearest-Neighbor Search (NNS) problem based on the Euclidean distance, although different measures can be more suitable for specific types of descriptors. The NNS problem is carried out by means of efficient indexing schemes such as Kd-trees [44]; given the high dimensionality of the search (typically of the order of a few hundreds), we will employ approximate Kd-tree-based schemes such as Kd-tree Forests in the publicly available implementation within the FLANN library [45].

For each element $d_1$ of the set of descriptors $D_1 = \{d_{11}, d_{12}, \ldots, d_{1n}\}$, we thus find its NN in the set $D_2 = \{d_{21}, d_{22}, \ldots, d_{2m}\}$, i.e. $d_2$. The pair $(d_1, d_2)$ yields a correspondence between the associated points on the clouds. Hence, a total of n correspondences can be potentially found out of this process. This is usually not convenient, since due to data noise and only partially overlapping surfaces many descriptors on $D_1$ will not have (or yield a correct) NN on $D_2$. To deal with this, effective correspondence rejection schemes are employed. The simplest approach is probably represented by thresholding the Euclidean distance in order to discard correspondences yielding a high distance in the descriptor space. Alternatively the threshold is applied directly to the Euclidean distance on the ratio between...
the 1-NN and the 2-NN [46]. These correspondence rejection schemes work in the descriptor space only. More advanced schemes exist that take into account also the 3D domain of point coordinates to perform outlier rejection in a more geometrical sense. One approach is represented by RANSAC [47], where the generative model is the 6-Degree-of-Freedom (6DOF) transformation between the two clouds. In order to estimate this transformation, a minimum of 3 correspondences are needed. RANSAC iteratively samples 3 correspondences, estimates the associated transformation and starts building a consensus over it. The consensus size (i.e. how well a correspondence needs to fit the model in order to be included in the consensus set) can be specified either as a metric parameter or as a function of the mesh resolution.

The developed pipeline contains the thresholding of the Euclidean distance followed by RANSAC. Other useful rejection schemes available in literature can be found in the pcl_registration module within PCL.

D. Coarse and Fine Registration

Once an outlier-free subset of correspondences has been determined as described in the previous stages, the 6 Degrees of Freedom (DOF) transformation between two clouds (or meshes) can be estimated. Among several options available in literature [48], the method relying on unit quaternion [30] is normally selected. The transformation brought in by this last step represents the coarse alignment between the two datasets. Since it has been estimated only based on a small subset of points, in most practical scenarios this transformation needs to be refined. Normally an ICP-like method is used taking into account all points of one cloud having a close nearest neighbor from the other clouds. If the coarse step was able to provide a good initial estimate for the refining stage, the ICP-like algorithm will quickly converge.

IV. MULTI-VIEW REGISTRATION

Multi-view registration aims at aligning together several views of the same object/scene into a unique 3D model. Conversely to other algorithms such as those employed within the Simultaneous Localization And Mapping (SLAM), in this case the views are handed in without temporal relationships with respect to one another. A standard approach for dealing with this problem is to rely on pairwise alignment (such as that described in the previous sections) between all possible view pairs and then to select appropriately the best subset of pairs to merge together in the final reconstruction. This selection is carried out with the goal of determining a sequence of pairwise alignments between as many views as possible, in order to find a univocal way to merge together all views into the same point cloud. This can be interpreted as transforming the loopy graph - where each vertex is represented by one view - into a directed graph [49]. A typical score employed to select the best pairwise registrations is the area of overlap between the two [49]. Optionally, once the views have been merged together into the same representation, a further refinement stage can be run globally on all views in order to further refine the global alignment between the selected views [51]. A sketch of this pipeline is shown in Figure 4.

As anticipated, during the pairwise alignment stage, it is important to associate to each view pair a score representing how well the two clouds have aligned. In our approach we propose to use, as a score \( s(i,j) \) between views \( i \) and \( j \), the normalized number of outlier-free correspondences used to compute the pairwise transformation

\[
s(i, j) = \frac{\text{max# corrs} - \text{#corrs}(i, j)}{\text{max# corrs}}
\]

where \( \text{#corrs}(i,j) \) is the number of outlier-free correspondences between view \( i \) and \( j \) after Correspondence Rejection and \( \text{max# corrs} \) is the maximum number of correspondences yielded among all pairwise transformations: this normalization allows bounding the weight to 1.

A useful step needed to eliminate pairwise alignments that didn’t reach a correct registration or between pairs that do not show any overlap is to threshold this score in order to avoid fetching these pairs to the following global selection stage. Once a reliable set of overlapping view pairs has been determined, a global selection stage selects the correct configuration of pairs that allows merging all views into the same one. The developed approach consists of the method presented in [49] based on the Minimum Spanning Tree (MST) of a graph where the vertices are represented by the views and the edges are represented by the previously defined scores associated to the respective pairwise alignments. Other recent methods exist in literature [50][51] which employ different global refinement and alignment schemes.

![Fig. 4: The typical pipeline of the multi-view registration algorithm, which includes the pairwise steps (orange blocks) applied over all overlapping view pairs (OPi).](image)

V. EXPERIMENTAL RESULTS

The section presents some experimental results on datasets related to the heritage field and deploying pairwise and multi-view feature-based alignment tools in PCL. Given the remarkably different nature of the processed data as well as the use of different 3D sensors, we avoided relying on a specific feature detection stage and in all reported results the proposed pipeline deploys random sampling in the feature detection stage and the SHOT descriptor [40] in the feature description stage.
A. Experiment #1

The used data consists of 5 different viewpoints of a Metopa artifact. Firstly, a pairwise registration was tested. Through descriptor matching (Figure 5), a series of point-to-point correspondences is found and a qualitative coarse alignment is achieved. Notably, the number of visually correct correspondences is high (see also Table 2) and the alignment brought in by the coarse step is already satisfactory. Table 2 reports quantitative results concerning the registration of all view pairs that could be aligned via descriptor matching (i.e., yielding a RANSAC consensus set above 5). In particular, Table 2 shows the number of correspondences left after RANSAC and the Root Mean Squared Error (RMSE) after the coarse registration step and after the successive ICP refinement step based on 15 iterations. The RMSE has been computed between all point pairs belonging to the two clouds being closer than 5 times the average mesh resolution computed over all views. From Table 2 it is also notable how the descriptor matching step is able to yield in all cases a good initialization for the ICP refinement, obtaining a registration error (“RMSE\_coarse”) which is close to that reported after the ICP (“RMSE\_ICP”).

![Table 2: Selected overlapping pairs with extracted correspondences and RMSE [mm] after the coarse and fine registration. The * symbol denotes the views selected by the MST algorithm for the multi-view alignment.

<table>
<thead>
<tr>
<th>View Pair</th>
<th># Corresp.</th>
<th>RMSE_coarse</th>
<th>RMSE_ICP</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0,1) *</td>
<td>94</td>
<td>0.0203</td>
<td>0.0176</td>
</tr>
<tr>
<td>(0,3) *</td>
<td>75</td>
<td>0.0174</td>
<td>0.0155</td>
</tr>
<tr>
<td>(1,2) *</td>
<td>80</td>
<td>0.0170</td>
<td>0.0170</td>
</tr>
<tr>
<td>(1,3)</td>
<td>69</td>
<td>0.0181</td>
<td>0.0181</td>
</tr>
<tr>
<td>(1,4) *</td>
<td>29</td>
<td>0.0171</td>
<td>0.0160</td>
</tr>
<tr>
<td>(2,3)</td>
<td>56</td>
<td>0.0176</td>
<td>0.0165</td>
</tr>
</tbody>
</table>

As anticipated, the multi-view registration approach is based on the Minimum Spanning Tree (MST) algorithm [49], which is able to select a subset of overlapping views yielding a direct graph (the pairs with the “*” symbol in Table 2). The graph and the selected edges building up the MST are shown in Fig. 6-b). Additionally, the final global reconstruction including all 5 views is shown in Figure 6-a).

B. Experiment #2

The experiment tried to automatically register with a pairwise approach a dataset with a very small overlap among the views (<15%). Beside the small overlap, the range maps are also denoted by poor geometric characteristics. The descriptor-based pipeline was nevertheless able to align a subset of cloud pairs, as it is demonstrated by the results of the registration shown using separate colors for each range map (Figure 7). Although in such difficult cases it is not expectable that automatic registration methods can obtain reliable global reconstruction in all cases, the results of this test motivate enforcing automated registration even with challenging data, since the use of automatic registration techniques can speed up notably the workload associated to human supervision by providing a good percentage of correct pairwise alignments without the need of human intervention.

C. Experiment #3

In this last experiment, a dataset consisting of 3 point clouds acquired with a TOF laser scanner is employed. The pairwise registration pipeline is applied over the two pairs, keeping the point cloud in common with both alignments as a reference. Differently from the other experiments, the automated registration is performed working only on the point cloud representation, without generating the 3D meshes. Figure 8 shows the three input point clouds as well as the results of the automatic alignment, respectively reporting the registration obtained by the coarse registration based on descriptor matching (shown in white, green red, each color associated to a different cloud) and the final alignment obtained by applying also the refinement stage based on the ICP method.

VI. CONCLUSIONS

In this paper we review the state-of-the-art in automated alignment / registration of point clouds and meshes. In particular, the feature-based approach was deeply considered and tested with three different datasets and experiments. This was possible using the implementations available in the PCL Library, an open source library for 3D robotics and computer vision applications.

Automated registration of 3D data is nowadays feasible and it can avoid long and tedious procedures for coarse alignment. Being based on a blind search for correspondences, like for the automated orientation of images, a reasonable overlap between the datasets and enough geometric features must be present otherwise accuracy and reliability of the approach are low.

The reported automated registration procedures (pairwise or multi-view) can be significantly useful in case of large amount of range maps or point clouds in order to speed up the data processing stage.

REFERENCES

Fig. 5: Examples of pairwise registrations for the 5 range data of experiment #1. In row (a) and (b), from left to right, the found point-to-point correspondences used for the coarse alignment are shown (pairs 1-2 and 1-3), together with the alignment results shown using a color for each view and the texture mode.

Fig. 6: The results of the multi-view alignment for experiment #1 (a). Loopy graph obtained via pairwise alignment of the 5 views, where each edge reports the associated weight based on outlier-free correspondences (b). The red edges are those selected by [49] to yield a directed graph for obtaining a global merge of all views.


Figure 7: Example of automated pairwise registration of range maps with small overlap and poor geometric features.

Figure 8: Pairwise (1-2 and 2-3) registration results for experiment #3.


