

# Registration of Deformable Objects using a Depth Camera

Rubén de Celis  
Vicomtech-IK4  
Foundation  
Mikeletegi 57,  
SPAIN, 20009,  
Donostia

rdcelis@vicomtech.org

Nagore Barrena  
Vicomtech-IK4  
Foundation  
Mikeletegi 57,  
SPAIN, 20009,  
Donostia

nbarrena@vicomtech.org

Jairo R. Sanchez  
Vicomtech-IK4  
Foundation  
Mikeletegi 57,  
SPAIN, 20009,  
Donostia

jrsanchez@vicomtech.org

Ramón J. Ugarte  
Vicomtech-IK4  
Foundation  
Mikeletegi 57,  
SPAIN, 20009,  
Donostia

rugarte@vicomtech.org

## ABSTRACT

This paper describes a method for registration and tracking of deformable objects from points clouds taken from depth cameras. Our method uses a reference model of the object in order to detect rigid and deformed regions in the input cloud. It is based on the fact that deformed objects normally have areas that are not affected by the deformations. These parts are found iteratively allowing to register the object using a chain of rigid transformations. Deformed regions are detected as those that do not satisfy rigidity constraints. Results show that correspondences of points belonging to both rigid and deformed regions can be accurately established with the reference model even in cluttered scenes.

## Keywords

3D Tracking, Deformable Object Recognition, Depth Camera

## 1 INTRODUCTION

The research presented in this paper is motivated by the need to detect deformations in elastic volumes for augmented reality applications and mechanical simulations. The field of computer vision already presents important advances in the tracking of non-rigid surfaces using monocular images [?]. These methods are usually based on geometric constraints applied to the analytical models of the objects to be detected. These models define the deformable objects as surfaces, and seek for results that are visually attractive. However, their physical behaviour involves properties such as elasticity, which affect their mechanical behaviour that cannot be well modelled with the cited techniques.

In the last years a great research effort has been done in the field of 3D reconstruction and object tracking thanks to the emergence of commodity depth cameras. They can give depth information for image pixels, usually using infrared technology. These kind of devices are particularly appropriate for the problem stated in this work as they allow to obtain a point cloud representation of the scene easily in real-time.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

The basis of 3D object recognition involves finding a set of correspondences between a known reference object and the reconstructed scene. This problem is usually solved under rigidity assumptions that allows registering the scene using an euclidean transformation. However, in the case of deformed objects the complexity of the problem increases as there is no a unique transformation that registers all the scene points with the reference.

In this work we pose the problem of deformable object registration as a recursive rigid registration problem. In our approach non deformed parts of the model are iteratively registered, representing the scene as a chain of rigid transformations. As a result points corresponding to deformed regions can be precisely detected and matched with the reference object in a straightforward manner. The main contribution of this method mainly relies on its simplicity which enables fast and robust implementations.

The paper is organized in four main sections: the "Related Work" section introduces the reader in the state of the art and develop tools, the "Method Overview" presents the execution pipeline of the method, the "Results" section is where are performed and discussed the data obtained, and "Conclusion and Future Work" shows the conclusion reached and open the following step in the investigation line.

## 2 RELATED WORK

The registration of 3D scenes is a well known problem that can be defined as the alignment of two different

point clouds representing the same scene. The emergence of depth cameras, such as Kinect, has spawned new interests in this line during the last years. One of the most representative works in this area was developed in [?] describing the techniques that are nowadays most used to register 3D objects. The main contribution of this research was a representation of objects as point clouds based on 3D feature histograms [?]. This representation describes the local geometry of object points relative to their neighbourhood that can be used to match point correspondences between different reconstructions. The work resulted in an open source library called Point Cloud Library (PCL) [?].

Most of the research works, like [?] and [?], assume rigid conditions for the objects to be scanned. Under this assumption point clouds can be aligned using a single euclidean transform. The most common registration techniques rely on the use of 3D keypoints detectors and descriptors in order to get correspondences that allow to find the transformation.

Some 3D detectors have been developed inspired by 2D image detectors like SIFT [?]. For now, a small set of detectors has been proposed specifically for 3D point clouds and range maps being ISS [?] and NARF [?] the most representatives.

Concerning descriptors, a commonly accepted taxonomy divides the descriptors in local, such as 3DSC [?], FPFH [?] or SHOT [?]; and global, as CVFH [?], ESF [?] or VFH [?]. Local descriptors are calculated for individual points being suitable for handling cluttered scenes and partially occluded objects. Global descriptors encode the object geometry, having higher invariance and being more descriptive. They are very suitable for the retrieval and classification of objects with poor geometric structure.

There are also very relevant advances in the registration of 3D scenes containing deformable elements. Works like [?], [?] or [?] illustrate cited procedure based on iterative minimization techniques. The input data is aligned with the reference model by minimizing an energy function that depends on various geometrical constraints. The main problem of these kind of methods is the existence of local minima in the objective function that cannot be always avoided. Method described in [?], simultaneously solves correspondences between points on source and reference clouds using an energy function that penalizes huge deformations and favours rigidity and consistency. The method implements a graph of nodes, whose nodes are chosen by uniform sampling, and each node have influence over the deformation of the nearby nodes. The computational cost is exponential with respect to the nodes and depends on the resolution used to generate the graph. The approach presented in [?], implements non-rigid reconstruction pipeline on the GPU and his approach include a custom

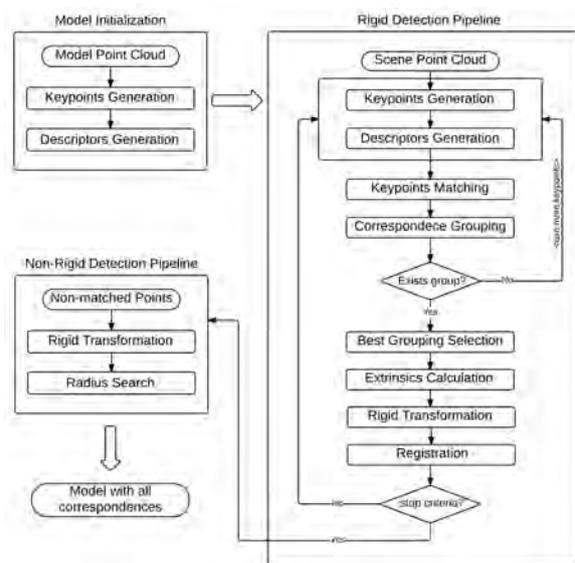


Figure 1: Detection Flow Diagram has 3 phases (Model Initialization, Rigid Detection Pipeline and Non-Rigid Pipeline).

RGB-D camera. The deformations between two scans are given by ARAP framework [?] that measures deformations existing between a pair of meshes. This type of registration is not useful when the goal is to detect deformations, because they perform deformations in the corresponding representation during the input data aggregation process.

Our method, unlike [?] or [?], is not based on iterative minimization frameworks. Instead we rely on simpler point correspondences that, besides simplicity, allow to avoid local minima as we can directly obtain the involved transformations once point correspondences are found.

### 3 METHOD OVERVIEW

The method proposed is designed to detect the deformations on the surface of the object. Given a reference model and the objects found in the scene, the method detects correspondences between the reference model and the model found in the scene, including undeformed and deformed regions. Figure 1 illustrates, the detection flow, divided in three phases. In the first phase, the reference model is initialized; in the second phase, the rigid regions considered as undeformed regions are detected; in the third phase, the non rigid regions considered as deformed regions are detected. The internal representation of the real world is based on point clouds, without edges.

In the initialization phase, the 3D keypoints of the model and their descriptors are computed from its point cloud representation. With the reference model initialized, the tracking is performed using a sequence of 3D

scans of the state of the scene as input. In the rigid detection phase, following the same procedure as in the initialization, the keypoints and descriptors are computed from the point cloud of the scene. Once the 3D keypoints and descriptors are computed, the keypoints are matched between the reference model and the scene, based on their descriptors information. Then all the correspondences are grouped in order to cluster the set of correspondences into instances that are present in the scene. An instance is defined as a subset of keypoints of the reference model matched with scene keypoints that satisfy a geometric consistency with the reference model. The best instance is used to calculate a rigid transformation to seek points corresponding to undeformed regions. With all the undeformed regions, the non-rigid detection phase is started with the non-matched points from the previous phase as input. Points of deformed regions are transformed with the best rigid transformation obtained in the previous phase. After applying the transformation, a radius search is executed to find for each non-matched point of the reference model the corresponding point in the scene.

The following subsections explain the phases in more detail.

### 3.1 Model Initialization

The reference model is represented as a point cloud. It can be loaded from a CAD or captured from a 3D scanner, provided that it is undeformed. The initialization process consists in the selection and computation of a set of keypoints and their descriptors from the cloud.

In order to obtain a good representation which enables a stable tracking, it is important to perform a proper selection of keypoints. There are well known detectors such as ISS and NARF which use the gradient of the surface around the vicinity to detect representative keypoints. Although there are good candidates to perform the matching of rigid surfaces, they are not appropriate to deformable models because the surface gradient is not invariant.

For this reason, a uniform downsampling is used in order to obtain the keypoints. Although this approach is not the best choice for rigid models, it works well with deformable models since it ensures a good distribution of keypoints along the surface of the object.

Once keypoints are selected SHOT descriptors are used to define each keypoint. SHOT descriptor shows a good balance between recognition accuracy and time complexity [?]. The SHOT descriptor encodes information on the topology of the surface in an area that stores information about the neighbourhood of a point. The area is divided into 32 bins, with 8 divisions along the azimuth, 22 along the elevation and 2 along the radius.

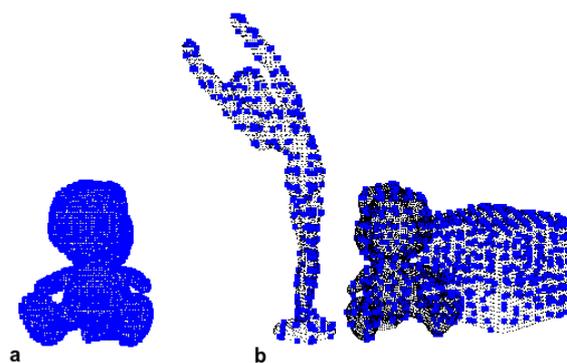


Figure 2: **a:** Reference model with all points as keypoints **b:** Scene that contains the reference model sampled with uniform downsampling to choose the keypoints. The keypoints are colored in blue.

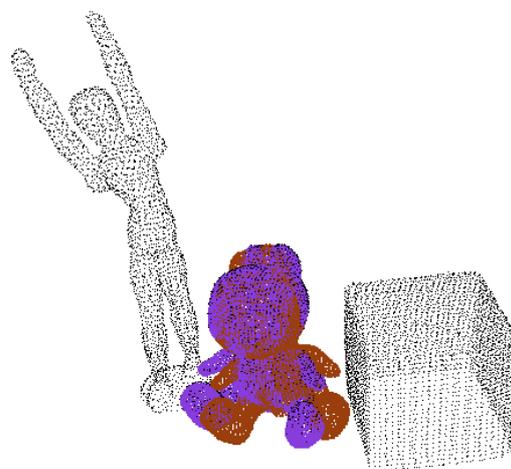


Figure 3: Two possible transformations corresponding to two instances of Fig.2(a). The purple transformation fit better than the brown transformation.

### 3.2 Rigid Detection Pipeline

The rigid detection pipeline starts with extraction of the keypoints and descriptors of the scene. This process is done using the same method as in the model initialization, i.e. using a uniform downsampling. But, in this case the frequency of the sampling is lower because of performance reasons (Figure 2).

Once the 3D keypoints and descriptors are computed, the descriptors are used in order to match the keypoints of the current scene and the keypoints of the reference model. All the correspondences obtained are grouped into subsets or instances. These instances are built enforcing geometric constraints between pairs of correspondences [?]. If there are not enough matches to allow a correspondence grouping, the scene is down-sampled again iteratively increasing the sampling frequency.

After obtaining the set of instances, a rigid transformation is obtained from the instance with the higher number of correspondences (see Figure 3). The rigid transformation is computed as in [?]. It can be computed with a minimum of three points to obtain the position and orientation with 6 DOF.

This result is used to partially register the scene with the reference. However, points not belonging to the selected instance may still not be aligned if the scene has deformations. In order to detect this situation an inlier test is performed using the distance between the points of the partially registered scene and their correspondences in the reference model. For points classified as outliers the registration process is executed again iteratively. This approximation is very effective to represent those deformations which can be expressed as chain of rigid transformations. The iterative process stops when a maximum number of iterations is reached, or a fixed percentage of correct matches is obtained. These thresholds are configured depending on the particular problem domain, the number of deformations and the result of the deformation.

### 3.3 Non-Rigid Detection Pipeline

When the stop criteria is reached, the non-rigid detection phase begins with the non-matched points from the rigid detection phase as input. In this phase, the transformation of the best set of correspondence grouping is used to register deformed points near from corresponding points in the scene. So, this transformation is a first approximation to the place where finally the deformed points could be localized in the scene.

With the first approximation performed, a radius search is executed for each non-matched point. The search is based on a threshold used as max distance between each reference model point and its corresponding scene point. The point of the scene closest to each searched point of the reference model, is taken as correspondence of the point. In addition, only non-matched points of the reference model and scene are used for the phase of non-rigid detection. The rest of the points are not taking into account for this phase. It improves the point search time and reduces the possible false positive in the matching process.

## 4 RESULTS

In this section we present a set of three experiments that show the results obtained using the proposed method. The experiments are divided into two groups: synthetic experiments that measure the accuracy and performance of the proposed method under controlled conditions using cad models, and not synthetic experiments that are focused to evaluate the method using models and scenes obtained with depth sensors. The solution used to obtain the models and scenes to the last group

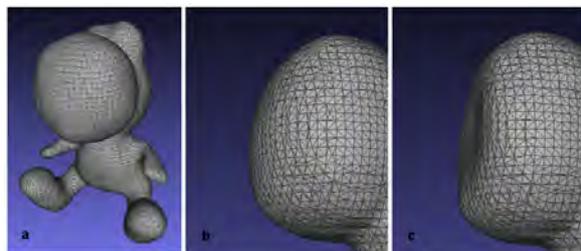


Figure 4: **a:** Reference model without deformations **b:** Detail of the candidate region to be deformed corresponding to the model **c:** Detail of the deformed region corresponding to the model

is Structure Sensor for mobile devices [?] with a simple 3D scanner. For all the experiments, only the point cloud corresponding to the vertices of the models and scenes are used in the method.

The experiments have been performed in a computer with Intel Core i5 3.2GHz, 8 GB of RAM DDR3 665MHz and Windows 8 64 bits operative system.

### 4.1 Synthetic Experiments

In the first experiment, the performance is evaluated using the model in Figure 4 (a). The model has 11798 points. The same model with a translation in one axis and two rotation in different axes is defined as model to be detected, so that the two models are misaligned. The aim is measuring the used time in the different detection tasks. The five main task involved in the process of detection are normal computation, sampling, descriptors computation, correspondences computation and correspondence grouping. Figure 5 shows the times for the different tasks against the sampling factor. The sampling factor is steadily reduced by 20% in each test, thereby increasing the number of keypoints used for the detection process.

The main execution time corresponds to the correspondences computation task. The normals computation task and the sampling task are constant with very low cost in terms of time. Moreover the descriptors computation, correspondences computation and correspondence grouping tasks increase proportionally to the number of points and therefore inversely to sampling factor.

For the second experiment, the model with the deformation in Figure 4 is inserted in a cluttered scene (see Figure 6). The scene is translated in one axis and is rotated twice in different axes to produce a misalignment with the model. Different level of noise is applied to each dimension of the 19836 points (see Fig6(a,b)). The noise has a uniform distribution between -1 and 1 that is multiplied by a maximum displacement for each intensity level of noise.



Figure 5: Time of five main task involved in the process of detection

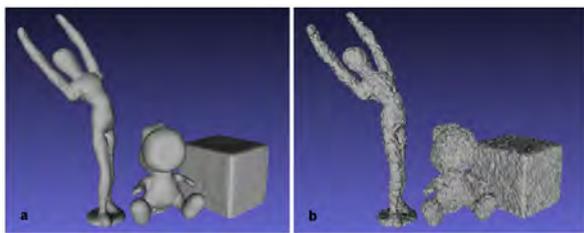


Figure 6: **a:** Original Scene (without noise) **b:** Scene with random noise in the vertices

Figure 7 shows the characterization of the different points in rigid or non-rigid region. The 0 column is the reference case with 11579 points corresponding to the rigid region and 189 point corresponding to the non-rigid region. The classification errors measure the number of points incorrectly classified respect to the reference case. The max displacement of each level of noise introduced to the scene is a percentage of the unit world (average distance of all points to its closest point) fixed in 0.0060702.

In most cases, the characterization of the point in rigid or non-rigid is correct and hence the matches are correct. Only the case with the 18% of the max displacement of the noise presents high classification errors, however the obtained matches are correct. Thereby the characterization of points which are wrong classified is incorrect but the matches obtained are correct.

The experiments with random noise demonstrate how robust the method is. While the models preserve the surface, it is possible to determine the deformation between the reference model and the scene with a reasonable error due to the noise.

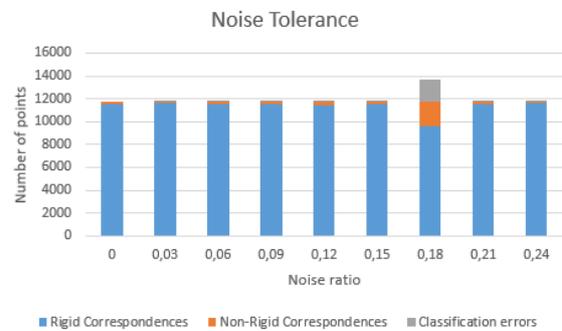


Figure 7: Characterization of the scene (Fig.6(a)) points in rigid or non-rigid with different percentage of word unit (0.0060702) used as max distance in each level of noise

## 4.2 Not Synthetic Experiments

The next set of experiments are performed to identify deformations using Structure Sensor to capture the reference model and the scene. The reconstructions obtained present noise but preserve the topological information of the object upon which the detection will be run. The following set of experiment is performed under the cited assumption.

As a general rule for the figures in the section, the green model represents the reference in the experiments. The blue lines, show a representative subset of the correspondences detected. A representative number of correspondences, and not all, are drawn for better visualization of the correspondences in the experiments.

Figure 8(a, b) shows the acquired pillow model used as reference. In this case a soft pillow is used. On the other hand, Figure 8(c,d) shows the second acquired pillow model with a deformed region produced by a force applied in the centre of the pillow.

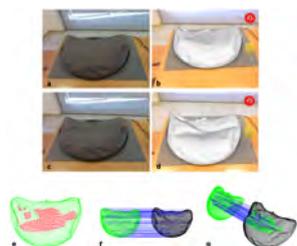


Figure 8: **a,b**: Reference model without deformations **c,d**: Model with deformed regions **e**: Reference model with the deformed region in red color and undeformed region in green **f**: Rigid correspondences **g**: Non-rigid correspondences

The method detects the deformed and undeformed regions (see Fig.8(e)), using as input the pillow reference model from Figure 8(b) and the second model from Figure 8(d). For each point of undeformed region (see Fig.8(f)) and for each point of deformed region (see Fig.8(g)), the correspondences between reference and the model are calculated. The green points of the pillow in Figure 8(g) represent the points of the scene detected as corresponding points of the deformed region. The execution time is about 4.55 second using 2067 keypoints of 10336 points of the reference model and 2654 keypoints of 10380 points of the second model.

Using the same reference model as in the previous experiment, the experiment is performed in cluttered scene (see Fig.9(a,b,c,d)). Figure 9(a,b) shows the state of the scene before the pillow deformation, and Figure 9(c,d) shows the state of the scene after the pillow deformation performed in the centre of it. The reference model has been segmented from the reconstruction obtained in Figure 9(b).

The distinction between deformed and undeformed is displayed in Figure 9(e). The point matching for undeformed regions are shown in Figure 9(f) and the deformed regions in Figure 9(g). The execution takes about 5.48 seconds using 2425 keypoints of 6015 points of the reference model and 2926 keypoints of 14478 points of the scene.

### 4.3 Discussion

The sampling for the model and the scene directly influences in the time execution and in conjunction with the threshold used in the rigid detection pipeline are main sensible parameters. Both determine the goodness of the result and depend on the resolution and on the characteristic topology of the reference model. Bad parametrization of the values produces bad characterization of the some points like deformed points, but nevertheless the match between the reference model and the scene is good.

The proposed method does not work with full deformation or greatly exaggerated deformations and fails if it is

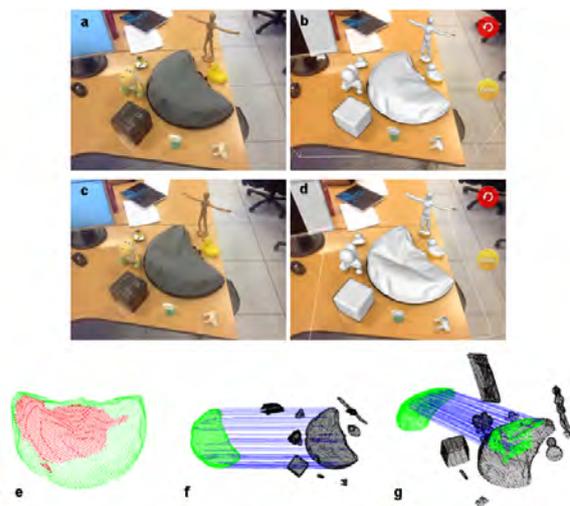


Figure 9: **a,b**: Reference model without deformations in a cluttered scene **c,d**: Model with deformed regions in a cluttered scene **e**: Reference model with the deformed region in red color and undeformed region in green **f**: Rigid correspondences **g**: Non-rigid correspondences

folded upon itself. It is necessary a region undeformed, large enough compared to the reference for searching the possible deformed regions. Also fails with models that are not topologically characterizable or without enough surface characterizable. An pragmatic example of this case is a sphere. It is impossible to know which points have been exactly deformed because any section of the surface is identical to any other section of the sphere.

When the objects present joints also can be approached as a chain of transformations (see Fig.10). In Figure 10, the reference model has two deformation produced by two rotations in two different parts of the humanoid, one in the waist and other one in the left elbow of the humanoid (see Fig.10(a,b)). In Figure 10(c), the two deformations respect to the reference model are detected and their corresponding points are matched in Figure 10(d).

In general the execution time is less than 1 second when the sampling factor is not too small and it increases the number of keypoints in the detection process. Thereby it might be possible the real-time execution, selecting the suitable values for the sampling factor and rigid detection threshold.

## 5 CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a 3D registration framework for deformable object tracking that can easily detect rigid and deformed regions. We propose the use of correspondence grouping that allows to

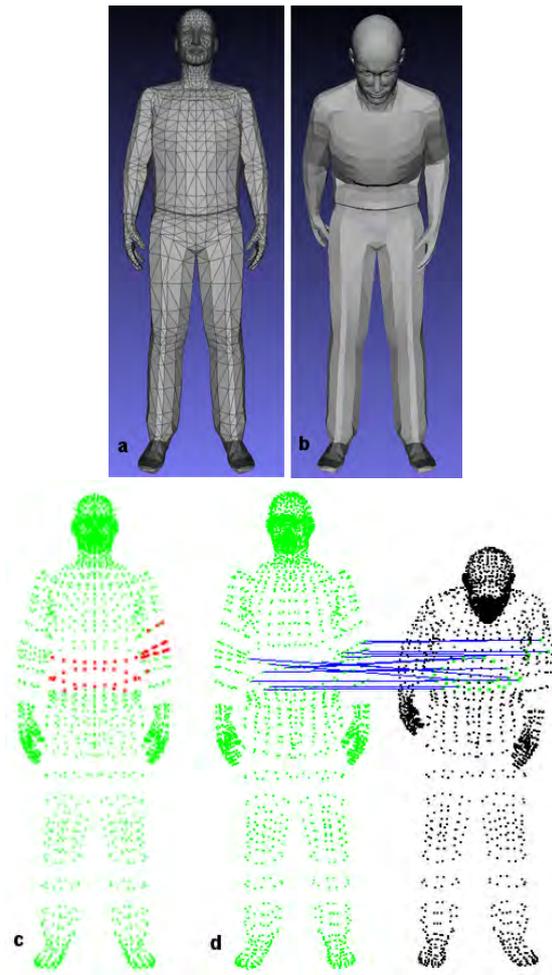


Figure 10: **a:** Reference model without deformations **b:** Model with deformed regions **c:** Reference model with the deformed region in red color and undeformed region in green **d:** Non-rigid correspondences

obtain a chain of rigid transformations for undeformed regions. This solution allows detecting deformed regions in a straightforward manner using a simple radius search. By this way, each point of the deformed region is matched with the closest point in the reference scene.

Unlike the solutions found in the state of the art, our approach relies in a simple 3D point correspondence strategy that allows converging fast and at the same time avoiding local minima.

Experiments have shown that the method behaves properly in cluttered scenes and it is particularly suitable for point clouds captured using commodity depth cameras. Moreover, the set of experiments performed in an uncontrolled environment proves the validity of the method. The method make it possible to isolate the undeformed regions and search for the deformed regions. Additionally, the set of experiments concerning to the chain of deformations shows that it can be suit-

able to obtain a chain of rigid transformations wherever needed.

As future work it is planned to extend the method integrating a frame-to-frame tracking strategy. This would allow to get a more stable and faster convergence, avoiding to compute in each frame all the point correspondence grouping. Moreover it would also make it easier to filter the input cloud detecting more outliers.

Finally, the work exposed is the first step in the development of a system for modelling elastic objects using physical mass-spring simulation techniques. Once completed, the registration could be further improved introducing mechanical constraints to the proposed instance grouping algorithm.

## 6 REFERENCES

- [Ale12] Luis A Alexandre. 3d descriptors for object and category recognition: a comparative evaluation. In Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, volume 1. Citeseer, 2012.
- [AMT+12] Aitor Aldoma, Zoltan-Csaba Marton, Federico Tombari, Walter Wohlking, Christian Potthast, Bernhard Zeisl, Radu Bogdan Rusu, Suat Gedikli, and Markus Vincze. Point cloud library. IEEE Robotics and Automation Magazine, 1070(9932/12), 2012.
- [AVB+11] Aitor Aldoma, Markus Vincze, Nico Blodow, David Gossow, Suat Gedikli, Radu Bogdan Rusu, and Gary Bradski. Cad-model recognition and 6dof pose estimation using 3d cues. In Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, pages 585-592. IEEE, 2011.
- [CB07] Hui Chen and Bir Bhanu. 3d free-form object recognition in range images using local surface patches. Pattern Recognition Letters, 28(10):1252-1262, 2007.
- [FHK+04] Andrea Frome, Daniel Huber, Ravi Kolluri, Thomas BÅ¼low, and Jitendra Malik. Recognizing objects in range data using regional point descriptors. In Computer Vision-ECCV 2004, pages 224-237. Springer, 2004.
- [IKH+11] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In Proceedings of the 24th annual ACM symposium on User interface software and technology, pages 559-568. ACM, 2011.

- [LAGP09] Hao Li, Bart Adams, Leonidas J Guibas, and Mark Pauly. Robust single-view geometry and motion reconstruction. In *ACM Transactions on Graphics (TOG)*, volume 28, page 175. ACM, 2009.
- [Low04] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91-110, 2004.
- [LSP08] Hao Li, Robert W Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer graphics forum*, volume 27, pages 1421-1430. Wiley Online Library, 2008.
- [NIH+11] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinect-fusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR)*, 2011 10th IEEE international symposium on, pages 127-136. IEEE, 2011.
- [Occ15] Structure Occipital. Sensor for mobile devices: <http://structure.io>, February 2015.
- [RBB09] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212-3217. IEEE, 2009.
- [RBTH10] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2155-2162. IEEE, 2010.
- [RMBB08] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, and Michael Beetz. Persistent point feature histograms for 3d point clouds. In *Proc 10th Int Conf Intel Autonomous Syst (IAS-10)*, Baden-Baden, Germany, pages 119-128, 2008.
- [Rus10] Radu Bogdan Rusu. Semantic 3d object maps for everyday manipulation in human living environments. *KI-KÄ $\frac{1}{4}$ nstliche Intelligenz*, 24(4):345-348, 2010.
- [Rus15] Radu Bogdan Rusu. Point cloud library (pcl): <http://pointclouds.org>, February 2015.
- [SA07] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, 2007.
- [SF10] Mathieu Salzmann and Pascal Fua. Deformable surface 3d reconstruction from monocular images. *Synthesis Lectures on Computer Vision*, 2(1):1-113, 2010.
- [SRKB10] Bastian Steder, Radu Bogdan Rusu, Kurt Konolige, and Wolfram Burgard. Narf: 3d range image features for object recognition. In *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, volume 44, 2010.
- [TSDS10] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *Computer Vision-ECCV 2010*, pages 356-369. Springer, 2010.
- [WV11] Walter Wohlkinger and Markus Vincze. Ensemble of shape functions for 3d object classification. In *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*, pages 2987-2992. IEEE, 2011.
- [Zho09] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 689-696. IEEE, 2009.
- [ZNI+14] Michael Zollhofer, Matthias Niebner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics, TOG*, 4, 2014.