Master thesis and internship[BR]- Master's thesis : Registration of sets of points obtained by x-ray tomography with respect to CAD models[BR]- Integration internship

Auteur : Fransolet, Maxime
Promoteur(s) : Béchet, Eric
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil en aérospatiale, à finalité spécialisée en "aerospace engineering"
Année académique : 2020-2021
URI/URL : http://hdl.handle.net/2268.2/11553

University of Liège - School of Engineering and Computer Science

Registration of sets of points obtained by x-ray tomography with respect to CAD models

Fransolet Maxime

Master's thesis carried out to obtain the degree of Master of Science in Aerospace & aeronautics Engineering

Academic year 2020-2021

Registration of sets of points obtained by x-ray tomography with respect to CAD models

Author: Maxime Fransolet
Promoter: Mr. Eric Béchet
Engineering, computing & technology, Aerospace & aeronautics engineering
Academic year 2020-2021

Abstract: The aim of this master thesis is to determine the defaults of an object by comparing a set of points obtained by its x-ray tomography (source) and a set of points obtained from its CAD model (target). The starting algorithm is made of three principal steps. First, the sets of points are pre-processed using a CAD software. Then, two registration steps are applied to the sets of points: namely, the covariance descriptor-based (CDB) algorithm and the improved iterative closest points (ICP) algorithm, with a novel estimation method for registration error proposed by [19]. Finally, the error of each point of the "source" point cloud with respect to the "target" point cloud is displayed. Different improvements of the starting algorithm will then be tested in order to improve its performances and its robustness. The different algorithms are applied to artificial and real sets of points in order to determine which improvements to keep for the final algorithm. The criteria are a trade off between computation cost and robustness of the algorithm. The final result is a robust algorithm that automatically registers two corresponding sets of points, removes the non corresponding points between the two sets and finds deformations of the source point cloud. Other improvements that have not been tested in this study are then discussed in order to give ideas for the future improvement of the algorithm.

Résumé: L'objectif de ce mémoire est de déterminer les défauts d'un objet en comparant un ensemble de points obtenus à partir de sa tomographie à rayons X (source) et un ensemble de points obtenus à partir de son modèle CAO (target). L'algorithme de départ est composé de trois étapes principales. Tout d'abord, les ensembles de points sont prétraités à l'aide d'un logiciel de CAO. Ensuite, deux étapes d'enregistrement sont appliquées aux ensembles de points, à savoir l'algorithme "covariance descriptor-based" (CDB) et l'algorithme "improved iterative closest points" (ICP) avec une nouvelle méthode d'estimation de l'erreur d'enregistrement proposée par [19]. Enfin, l'erreur de chaque point du nuage de points "source" par rapport à sa projection sur le nuage de points "target" est affichée. Différentes améliorations de l'algorithme de départ sont ensuite testées afin d'améliorer ses performances et sa robustesse. Les différents algorithmes sont testés sur des ensembles de points artificiels et réels afin de déterminer quelles améliorations sont à retenir pour l'algorithme final. Les critères utilisés sont un compromis entre le temps de calcul et la robustesse de l'algorithme. Le résultat final est un algorithme robuste qui enregistre automatiquement deux ensembles de points correspondants, supprime ensuite les points sans correspondances entre les deux ensembles et trouve les déformations du nuage de points source.D'autres améliorations qui n'ont pas été testées dans cette étude sont ensuite discutées afin de donner des idées pour l'amélioration future de l'algorithme.

Contents

1 Introduction **1**

2 Starting algorithm **2**
2.1 Coarse registration: Covariance descriptors-based algorithm (CDB) 3
2.2 Fine registration: Iterative Closest Point algorithm (ICP) 4
2.3 Results 6
2.3.1 Sample 1 6
2.3.2 Sample 2 9

3 Improvements of the ICP **10**
3.1 Results 11
3.2 Limitations 13
3.2.1 CDB limitations 13
3.2.2 ICP limitations 15

4 Improvements of the Coarse registration **17**
4.1 Improvements of the coarse registration using geometrical features 17
4.1.1 Algorithm 1 18
4.1.2 Algorithm 2 19
4.2 Improvement of the CBD using multiple ICP 22
4.2.1 Results 22

5 Determination of the defects and deformations **22**
5.1 NX models 25
5.2 Dental prosthesis models 27
5.3 Symmetrical mechanical models 28
5.4 Non symmetrical mechanical model 35
5.5 Spherical model 37

6 Determination of the parameters **39**

7 Future improvements **41**

8 Conclusion **43**

1 Introduction

Recently, industrial products have to meet increasingly narrow requirements concerning its performances and appearance. Obviously, there are always manufacturing defects in a workpiece in comparison to its CAD model ([1]-[2]), and the functionality of the product may be impacted. Therefore, evaluating those defects in a non-destructive way is an important steps as it will give a good indication of the quality of a product without putting it to waste. The industrial registration technology is a method used to analyse the defects of the workpiece according to its CAD model. It is used to check if the workpiece meets the requirements or not. The use of computed tomography (CT) has increased in the industry to perform non-destructive testing of products ([3]-[4]). The comparison analysis is mostly performed between the point cloud obtained from CT and a point cloud obtained from the CAD of the part. Point clouds are sets of points represented in a three dimensional coordinate system and are often used to represent the external surface of an object. Comparing the model and the part using point clouds is time efficient and convenient as the data structures are simple and general.

In order to compare the point clouds, a first step called registration is performed as the point clouds do not share the same origin axes. In practice, different approaches have been used to perform the registration CT images to CAD models. The most common method consist in manually detecting the corresponding feature points of the image and the model. The feature points are points located on high curvature zones such as corners or edges. In 1992, a semi-automatic approach was proposed by L. Tong and X. Ying [6]. This method was used to improve the robustness of the manual method. The method is based on curvature features and texture features of the point clouds to perform the registration. The user still has to manually select a minimum of three pairs of corresponding points and then apply an ICP algorithm to refine the results. Those methods ([5]-[6]-[7]) unfortunately depend too much on human error and thus can not be used in the industry. In order to reduce human intervention, Pinter and al. [8] compared three methods based on anisotropic Gaussian filtering, Hessian matrix calculation, and structure tensor calculation, to describe the correct fibre orientations of fibre reinforced composite materials. However, the procedure used is complex and thus the method might fail to perform the registration without errors due to the analysis.

Recently, Rae [9] constructed a "Non-rigid point cloud registration based lung motion estimation using tangent-plane distance". In 2010, Bernardini [10] performed initial registration by using surface curvature and SURF. Kim [11] proposed a "fully automated registration of 3D data to a 3D CAD model for project progress monitoring". In 2008, Sdika [13] performed the registration by applying "A fast nonrigid image registration with constraints on the Jacobian using large scale constrained optimization". Ran applied the genetic algorithm to the Powell algorithm [14]. It was used to get the registration parameters as close to the global optimum as possible with a certain precision. All those researches ([13]-[14]-[15]) made it possible to perform registration with a guaranteed accuracy. However, they are hard to implement because of their complexity or impracticability.

In 2007, Akca [16] worked on "co-registration of surfaces by 3D least square matching". In 2014, Theiler and his partners [17] proposed a "Keypoint-based 4-Points Congruent Sets - Automated marker-less registration of laser scans". In 2015, Yang [12] proposed an optimization method by combining L-BFGS-B with cat swarm optimization for medical image registration. They are currently working on the robustness of the method. In 2016, Chen H and his workmates [18] presented a 3D registration based on perception in augmented reality environment, it is a coplanar 4 points set registration algorithm used to align the source point cloud and target point cloud.

These strategies make the automatic registration feasible, unfortunately the robustness is still low.

In 2019, Gongjie Yao, Yongning Zou, Jue Wang, Haosong Yu and Taoyan Chen [19] proposed a Fully automated registration of 3D CT data to CAD model for surface deviation measurement. This method is more robust and does not involve human interaction. Moreover, it uses point to plane projections to identify the error properly.

This master thesis mainly follows the methodology proposed in [19]. The core of the algorithm is a pre-process followed by a coarse matching using a coarse driven based (CDB) algorithm. An iterative closest point (ICP) algorithm is then used to perform a fine registration. The deformation of the workpiece is directly given by the error obtained from the ICP. Those algorithms are tested and their limitations are then highlighted. Therefore, modifications are tested in order to improve the robustness of the algorithm as well as its precision. The limitations of the ICP algorithm come from non corresponding point as well as the approximation of the error. The limitations of the CDB comes from the impossibility to detect the correct mirror transformation when large portion of the CT scan are missing. Firstly, the precision of the ICP is improved by removing non corresponding point between the 2 sets of points. Different norm are then tested and discussed. An approach to improve the determination of the error of the ICP will also be discussed but not tested. Then, different methodologies are tested to improve the robustness of the CDB in order to match the 2 sets when the correct mirror can not be determined. Two algorithms based on the features of the data sets as well as an algorithm based on multiple ICP are tested. Unfortunately, the improvements of the CDB based on features must be rejected due to a poor robustness of those algorithms.

Even if the principal objective of this work is to determine the defect of a workpiece in a non-destructive way, the registration and comparison of point clouds can be used in other domain. This type of algorithm can also be used to compare CAD models. In fact, some big companies store a huge data base of CAD models. When the data were digitized, the 3 views models had to be converted to CAD models. It so happens that different CAD models of the same pieces were done. As those CAD models were not build in the same way, it is impossible to directly delete the copy using a CAD software. Thus, this algorithm could be used to recognize the multiple model of the same pieces by comparing point clouds obtained from those CAD in order to reduce the data base of those companies.

2 Starting algorithm

The algorithm used as a starting point is mainly inspired by the algorithm proposed in [19]. The first step of the algorithm is to pre-process the two point clouds. The pre-process is used to ensure that the two point clouds share the same point density. This step is performed manually using a CAD software. Automatic methods can be used [19] but was not tested in this work. Unfortunately, the origin axes of the two point clouds are different. Thus, the source point cloud must be translated and rotated in space in order to match the target point cloud. This step is called registration and is divided in two parts. Firstly, the point clouds are roughly matched using the covariance descriptors-based (CDB) algorithm, which is presented in section 2.1. This algorithm matches the centroid and aligns the principal axes of the point clouds. This step is important as it is an efficient way to match the points as it does not involve iteration. However, the matching of the CDB is not precise due to the randomness of the point clouds. Therefore, an iterative closest point (ICP) algorithm is applied to the rough matching. The ICP algoritm is presented in section 2.2. This algorithm gives better precision as it is based on corresponding points least

square distances. An improvement of the classical ICP is made by projecting the source points to local planes build from the target point cloud to determine the corresponding pair of points. This algorithm must always be preceded by a rough matching as it has a high computation cost due to the need of iterations. Moreover, if the initial alignment is bad, the ICP may converge to local minimums. The point to plane error obtained after the convergence of the ICP gives a good approximation of the deformation at that location. The methodology is tested on different test samples that are either man-made or coming from the industry and the results are then presented in section 2.3.

2.1 Coarse registration: Covariance descriptors-based algorithm (CDB)

The first registration step to apply to the two pre-processed point clouds is the covariance descriptor-based algorithm. This algorithm matches the centroid and the principal axes of the two point clouds with the help of the singular value decomposition (SVD). The first step of the method is to determine the centroid of both point cloud. The point clouds are called the source point cloud ($\mathbf{S}$) and the target point cloud ($\mathbf{T}$). $\mathbf{S}$ is translated and rotated while $\mathbf{T}$ is considered the reference and thus is fixed. The centroid is easily determined by equation (1). The centroid of $\mathbf{S}$ and the centroid of $\mathbf{T}$ are respectively called $\bar{\mathbf{p}}^{(\mathbf{1})}$ and $\bar{\mathbf{p}}^{(\mathbf{2})}$. $\mathbf{p_i^{(k)}}$ are the coordinates of the i^{th} point of a point cloud k and N_k is its number of points, k=1 if $\mathbf{p_i^{(k)}} \in \mathbf{S}$ and 2 if $\mathbf{p_i^{(k)}} \in \mathbf{T}$.

$$\bar{\mathbf{p}}^{(\mathbf{k})} = \frac{1}{N_k} \sum_{i=0}^{N_k-1} \mathbf{p_i^{(k)}}. \tag{1}$$

The next step is to determine the covariance matrix of both $\mathbf{S}$ and $\mathbf{T}$. The covariance matrix of a point cloud k is defined as (2),

$$\mathbf{Cov_k} = \frac{1}{N_k} \sum_{i=0}^{N_k-1} (\mathbf{p_i^{(k)}} - \bar{\mathbf{p}}^{(\mathbf{k})})(\mathbf{p_i^{(k)}} - \bar{\mathbf{p}}^{(\mathbf{k})})^T. \tag{2}$$

After the covariance matrix of both point cloud is determined, the singular value decomposition of the covariance matrices is performed. The SVD writes as in equation (3). As the covariance matrices are symmetric, the matrices $\mathbf{U}$ and $\mathbf{V}$ are equal. The matrices $\mathbf{U}$ and $\mathbf{D}$ contain respectively the eigenvectors and the eigenvalues of the covariance matrix, which correspond to the principal axes of the point cloud.

$$\mathbf{Cov_k} = \mathbf{U_k D_k V_k}^T. \tag{3}$$

The translation matrix, $\mathbf{Tr}$, and the rotation matrix, $\mathbf{R}$, can now easily be obtained from the matrices $\mathbf{U_k}$ as shown in equation (4) and (5).

$$\mathbf{R} = \mathbf{U_2 U_1}^{-1}, \tag{4}$$

$$\mathbf{Tr} = \bar{\mathbf{p}}^{(\mathbf{2})} - \mathbf{R}\bar{\mathbf{p}}^{(\mathbf{1})}. \tag{5}$$

Each point of the source point cloud is now translated and rotated as shown in equation (6),

$$\mathbf{p_i^{(1)}} = \mathbf{R p_i^{(1)}} + \mathbf{Tr}. \tag{6}$$

Typically, the eigenvectors are determined without a preference of direction. Thus, even if each eigenvectors are paired with respect to their eigenvalues, the source point cloud might suffer from all the mirror symmetries about the plane defined by its principal axes. Therefore, the next step of the algorithm is to determine which mirror is the correct one. If no other filters are applied, eight mirrors are possible. Four of them are physical and correspond to the 180-degrees rotation about one of the principal axes but the other four correspond to a symmetry, thus are not rigid body motions. An example of a piece and the piece obtained after its mirror transformation is shown in figure (1). Certain methods may be used to remove the latter *a priori* but such methods have not been implemented in this study. Thus, keeping the wrong mirror might lead to unusable data, as presented in section (3.2). In order to be as robust as possible, all the mirror transformations are verified and only the transformation that minimizes the error is kept. The way the error is determined is discussed in section (2.2).

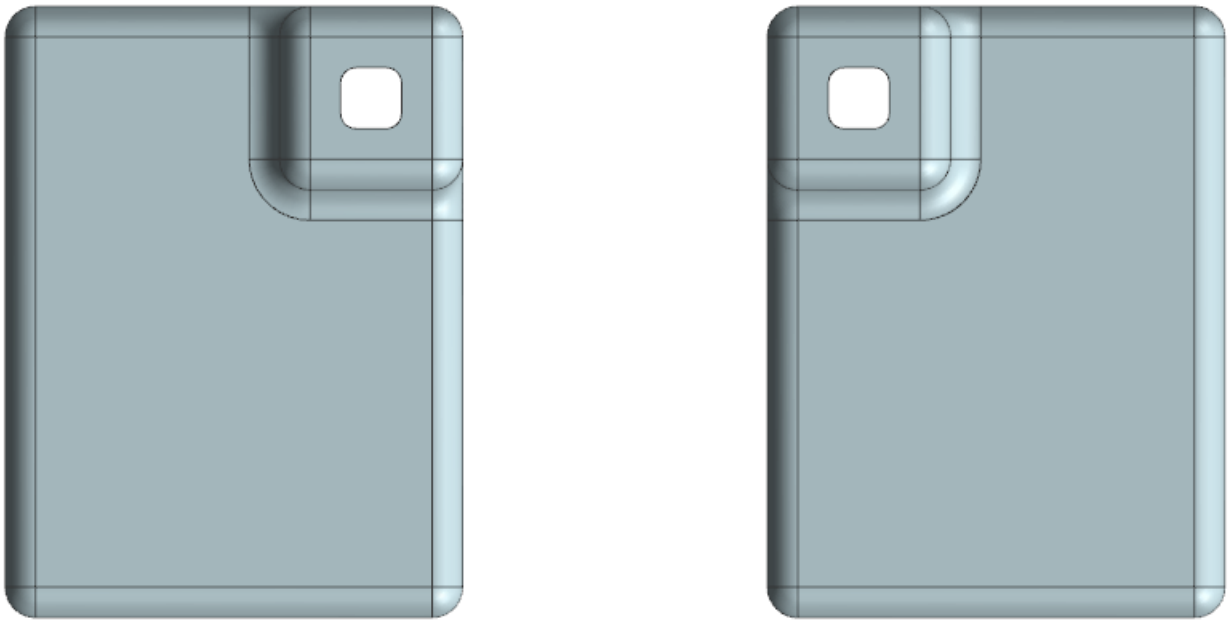

Figure 1: Example of a mirror transformation

Unfortunately, the results obtained from the CDB algorithm are not very precise when the clouds differ significantly. The matching is only perfect if the 2 point clouds are perfectly identical. This is illustrated in figure (2). This is why a fine registration is necessary to converge to a better registration of the point clouds. Moreover, the CDB algorithm only provides a matching of the principal axes, and does not give any idea about deformed regions of the workpiece. Nevertheless, the CDB is a good starting point as the matching is quick and robust.

2.2 Fine registration: Iterative Closest Point algorithm (ICP)

After the CDB is performed and the correct mirror was determined, an Iterative Closest Point algorithm can be performed to do a finer registration. The ICP algorithm matches point clouds by minimizing a cost function based on corresponding pairs of points. The first step of the algorithm is to determine, for each point of the source point cloud (**S**), its corresponding point on the target point cloud (**T**). The classical ICP simply takes the closest point contained in the target point

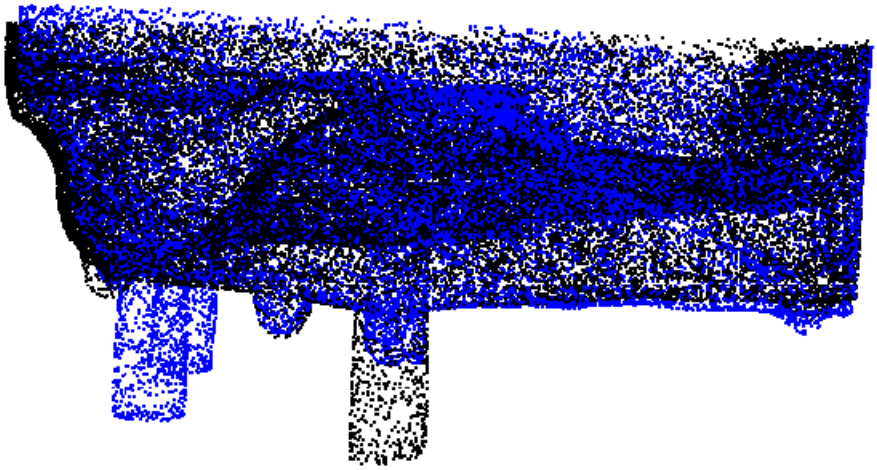

Figure 2: Limit coarse registration

cloud. Unfortunately, this approach leads to additional error due to the random aspect of point clouds, which is not ideal to determine defects in the workpiece. The corresponding point is therefore a projection of $\mathbf{p_i^{(1)}}$ on a local plane build with its three closest points $\mathbf{p_{l,m,n}^{(2)}}$. If the three points are close to be aligned, the plane can not be properly defined. In that case, the next closest point is used to replace $\mathbf{p_n^{(2)}}$ until the plane can be defined in a robust way. The distance from $\mathbf{p_i^{(1)}}$ to its projection is thus determined by equation (7). For the sake of simplicity, $\mathbf{p_i^{(1)}}$ is called $\mathbf{p_i}$ from now on.

$$d(\mathbf{p_i}, \overrightarrow{\mathbf{n_i}}, D) = \frac{A_i x_i + B_i y_i + C_i z_i + D_i}{\sqrt{A_i^2 + B_i^2 + C_i^2}}, \tag{7}$$

where x_i, y_i, z_i are the coordinates of $\mathbf{p_i}$ and $\overrightarrow{\mathbf{n_i}} = (A_i, B_i, C_i)$ is the normal vector representing the local plane of interest. The projection of $\mathbf{p_i}$ on its corresponding local plane is called $\mathbf{q_i}$. This step makes it possible to determine pair of corresponding points. The second step of the algorithm is to find the translation and rotation matrices that minimize a cost function. The cost function used is the mean error, i.e. the mean value of the local error. In this study, the local error is defined as the square distance between $\mathbf{p_i}$ and $\mathbf{q_i}$. The cost function is shown in equation (8),

$$f = \frac{1}{N} \sum_{i=0}^{N-1} dist(\mathbf{q_i}, \mathbf{p_i})^2. \tag{8}$$

The minimization can thus be performed by the least square fitting of the cost function f. The cost function obtained after the least square fitting is performed and shown in equation (9).

$$\begin{cases} min \sum_{j=0}^{N-1} d_j(\mathbf{p_i}, \overrightarrow{\mathbf{n_i}}, D), \\ f = \frac{1}{N} \sum_{i=0}^{N-1} ||\mathbf{q_i} - \mathbf{R}\mathbf{p_i} - \mathbf{Tr}||^2. \end{cases} \tag{9}$$

The least square fitting of 2 corresponding point clouds is based on the combined covariance matrix of those point clouds. This covariance matrix ($\mathbf{H}$) is computed using equation (10). The rotation and translation matrices can now be determined from the SVD of $\mathbf{H}$ as shown in equation (11).

$$\mathbf{H} = \sum_{i=0}^{N-1} (\mathbf{q_i} - \bar{\mathbf{q}})(\mathbf{p_i} - \bar{\mathbf{p}})^T, \tag{10}$$

where $\bar{\mathbf{q}}$ and $\bar{\mathbf{p}}$ are respectively the centroid of the source point cloud and the projected point cloud computed as shown in equation (1).

$$\begin{cases} \mathbf{R} = \mathbf{V}\mathbf{U}^T, \\ \mathbf{Tr} = \bar{\mathbf{q}} - \mathbf{R}\bar{\mathbf{p}}. \end{cases} \tag{11}$$

Finally, the convergence of the registration is verified. If the variation of the mean error between 2 iterations is not below a certain threshold t, the algorithm iterates until the error threshold is satisfied. If the number of iteration is above a predefined maximum number of iteration, the algorithm stops and tells it is unable to converge. The chosen error threshold is a percentage of the relative error between the second to last and the last iteration. Inequation (12) must be satisfied to check convergence of the ICP where k is the current iteration and t is the threshold, where e is the relative erro between 2 iterations. The scheme of the ICP algorithm is shown in figure (3).

$$e = \frac{f^{k+1} - f^k}{f^{k+1}} < t. \tag{12}$$

2.3 Results

The results of this procedure is highlighted by two data sets. One of them is artificial and the other one comes from a real x-ray tomography.

2.3.1 Sample 1

The first sample is a simple model build using NX. It is called the NX models. The 2 point clouds represent the exact same geometry. The points are however randomly spread out along the surface of the model to simulate a real CT scan. The target and the source point clouds are shown in figure (4) and (5).

A random rotation as well as a random translation are applied to the source point cloud. The CDB is first applied to the source point cloud to match the target point cloud. The matching of the 2 point clouds after this first step is shown in figure (6). Then, the ICP algorithm is applied to the 2 resulting point clouds. The iterative closest point threshold t is set to 0.05 % and the maximum

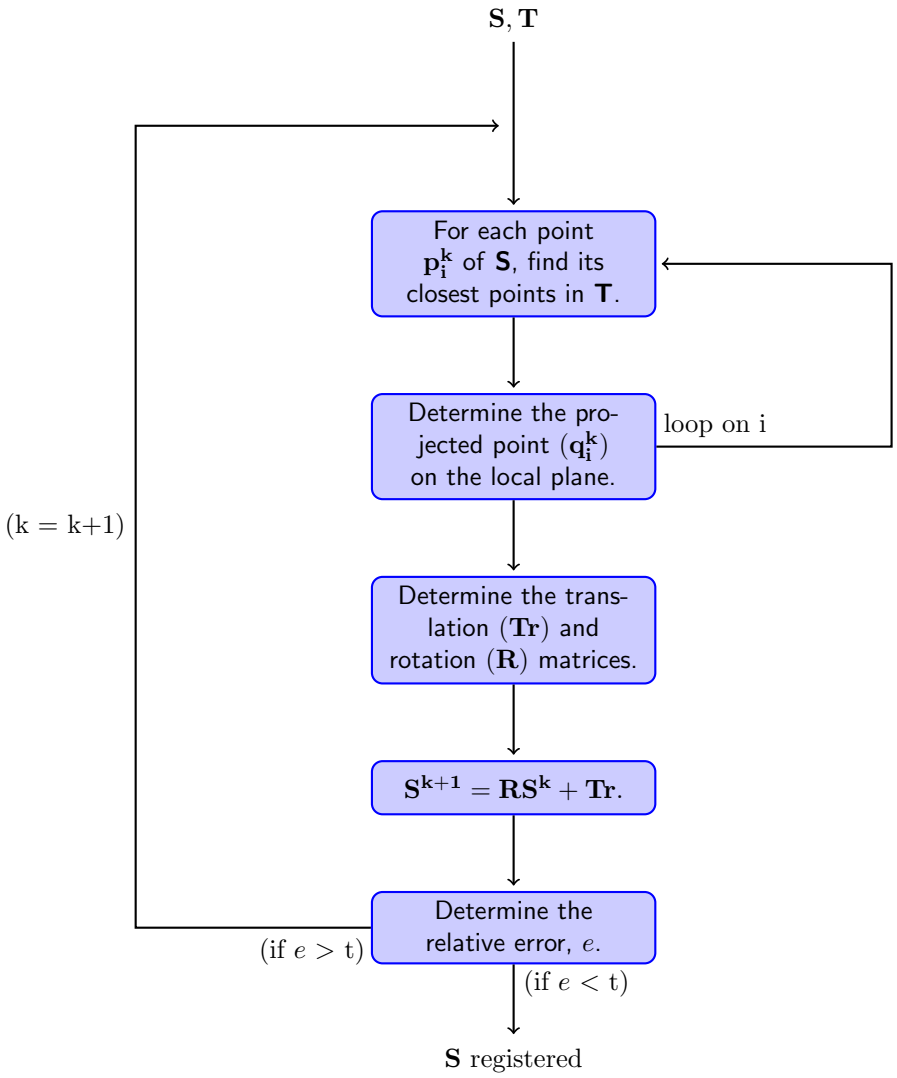

Figure 3: Scheme of the ICP algorithm

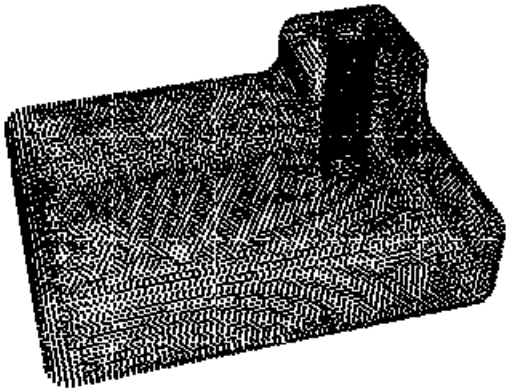

Figure 4: Sample 1 : Target point cloud

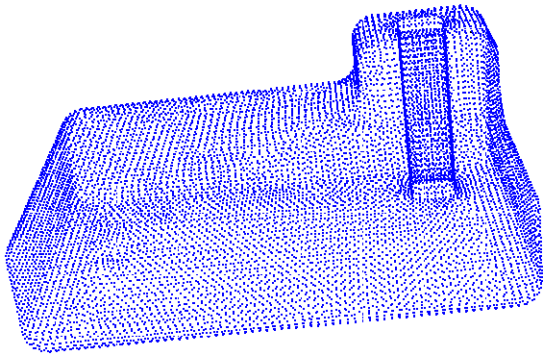

Figure 5: Sample 1 : Source point cloud

number of iteration is set to 20. The matching is shown in figure (7) and the convergence of the

error is plotted in figure (8).

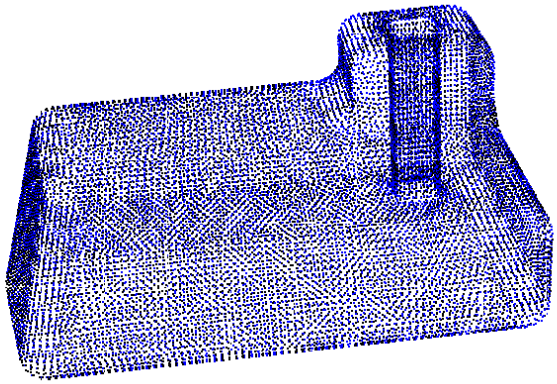

Figure 6: Sample 1 : CDB matching

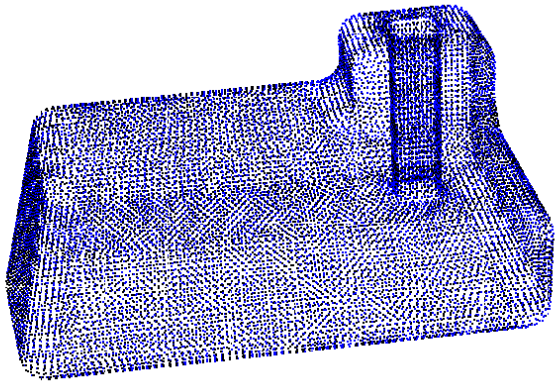

Figure 7: Sample 1 : ICP matching

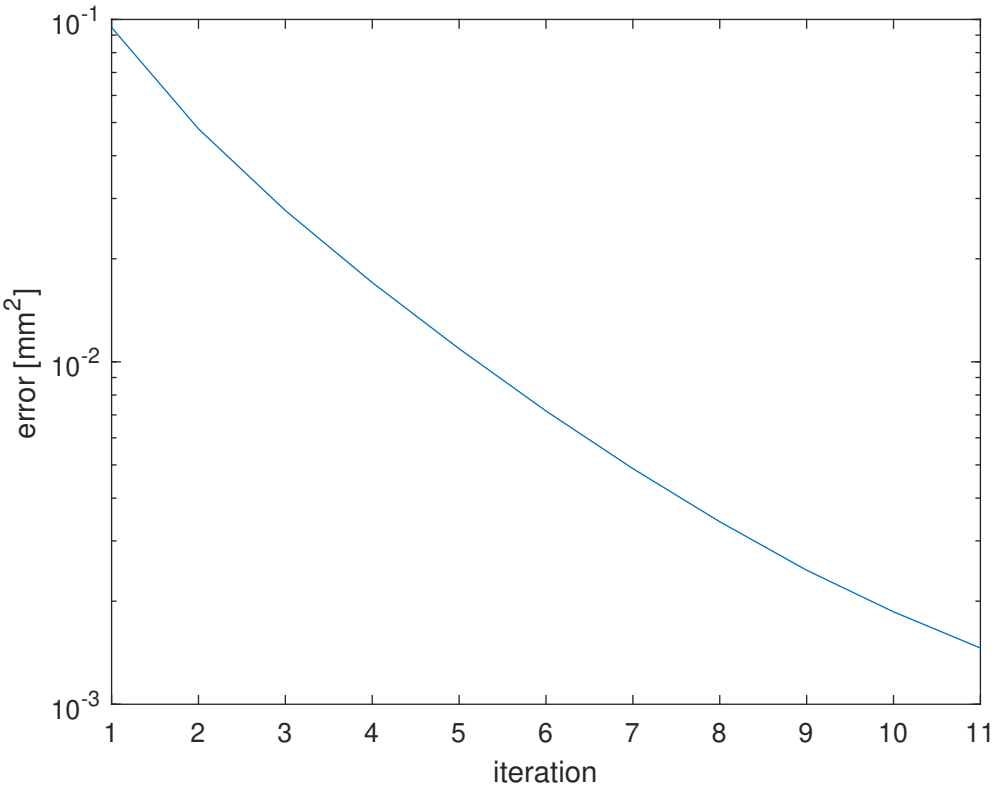

Figure 8: Sample 1 : Convergence of the ICP

It can be seen that the matching made by the CDB already gives a good matching visually as the **S** and **T** share the exact same geometry and the repartition of their points are roughly equivalent. However, it can be seen from the convergence analysis that the ICP reduced the error by a factor more than 10. In fact, the error in this case must converge to 0 as the original geometries are the same. A residual error may however be seen due to the approximation of the target point cloud by local plane as those planes do not match the exact geometry of the model.

2.3.2 Sample 2

The second sample uses 2 different scans of a dental prosthesis. The geometry of the 2 scans is not exactly the same due to the measurement requirements, some part where added in order to fix the workpiece. It leads to non corresponding part between the two models. The shape of both scans can be seen in figure (9) and (10). The 2 scans are respectively named empreinte plâtre 1 and empreinte plâtre 2.

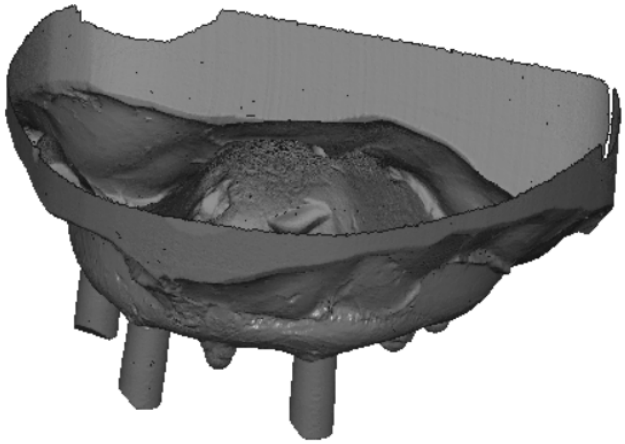

Figure 9: Sample 2 : Empreinte plâtre 1

Figure 10: Sample 2 : Empreinte plâtre 2

As empreinte plâtre 2 contains more data points, it is used as the target point cloud. The matching of the ICP is shown in figure (11) and (12). The convergence of the ICP error is shown in figure (13).

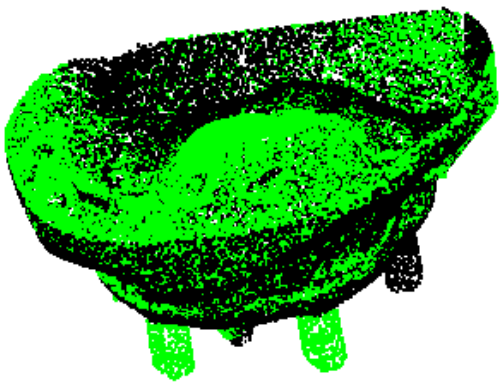

Figure 11: Sample 2 : ICP matching 1

Figure 12: Sample 2 : ICP matching 2

As many points in the source point cloud do not have correspondence in the target point cloud, the matching of the 2 point clouds is not accurate. The bad matching can be seen visually. The non corresponding points should be removed from the error estimation to give proper results. Such an improvement is discussed in section (3).

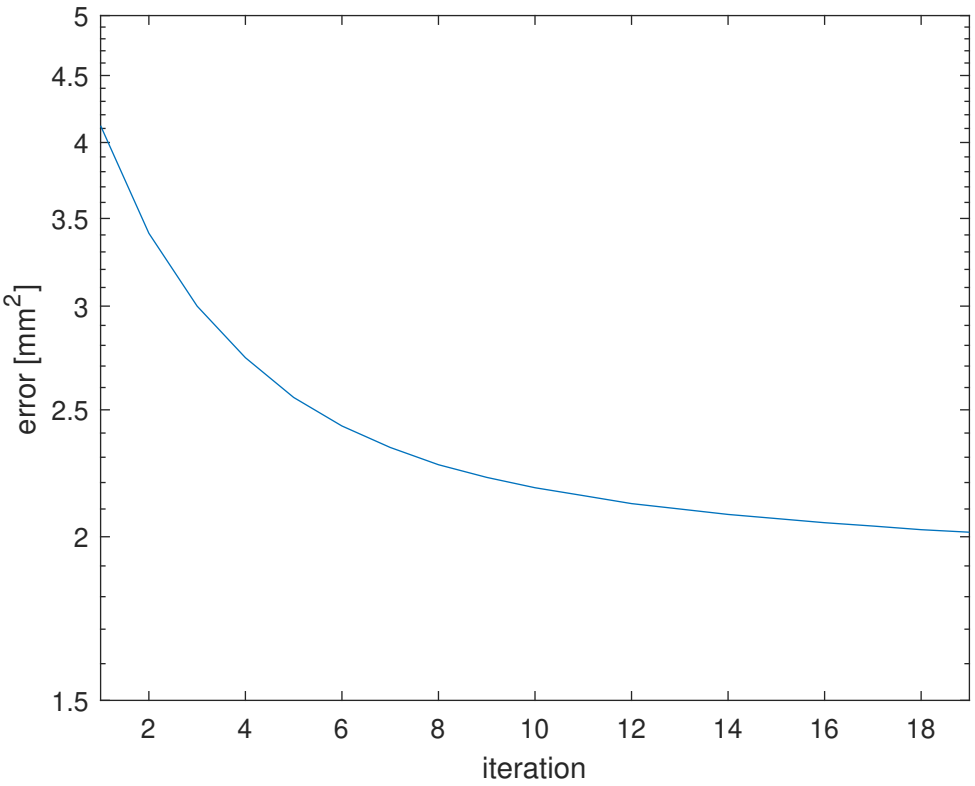

Figure 13: Sample 2 : Convergence of the ICP

3 Improvements of the ICP

In order to reduce the additional error caused by non corresponding points, a semi automatic approach is performed. All the points from S with a local error (f_i^k) at a given iteration k over a certain threshold (t_{rem}) are not taken into account to compute the translation and the rotation matrices. The local error is the square distance from a point to its projection on local plane of T. These points are also removed from the computation of the error f^k. The threshold used is based on the mean value of the error at a given iteration. The criteria is shown in equation (13).

$$\begin{cases} if \quad (f_i^k < t_{rem} * f^k), \ \mathbf{p_i} \ and \ \mathbf{q_i} \ are \ kept, \\ else \qquad\qquad\qquad\quad , \ \mathbf{p_i} \ and \ \mathbf{q_i} \ are \ rejected. \end{cases} \tag{13}$$

The number of removed points thus evolves with each iteration of the ICP. Unfortunately, the ideal threshold t_{rem} varies in parallel to the amount of non corresponding points as it is based on the mean error. Thus, this selection has to be performed manually. The removed points are highlighted by the graphical representation. The updated scheme of the ICP is shown in figure (14). From now on, points above the threshold are always rejected when the ICP is performed even if it is not specified.

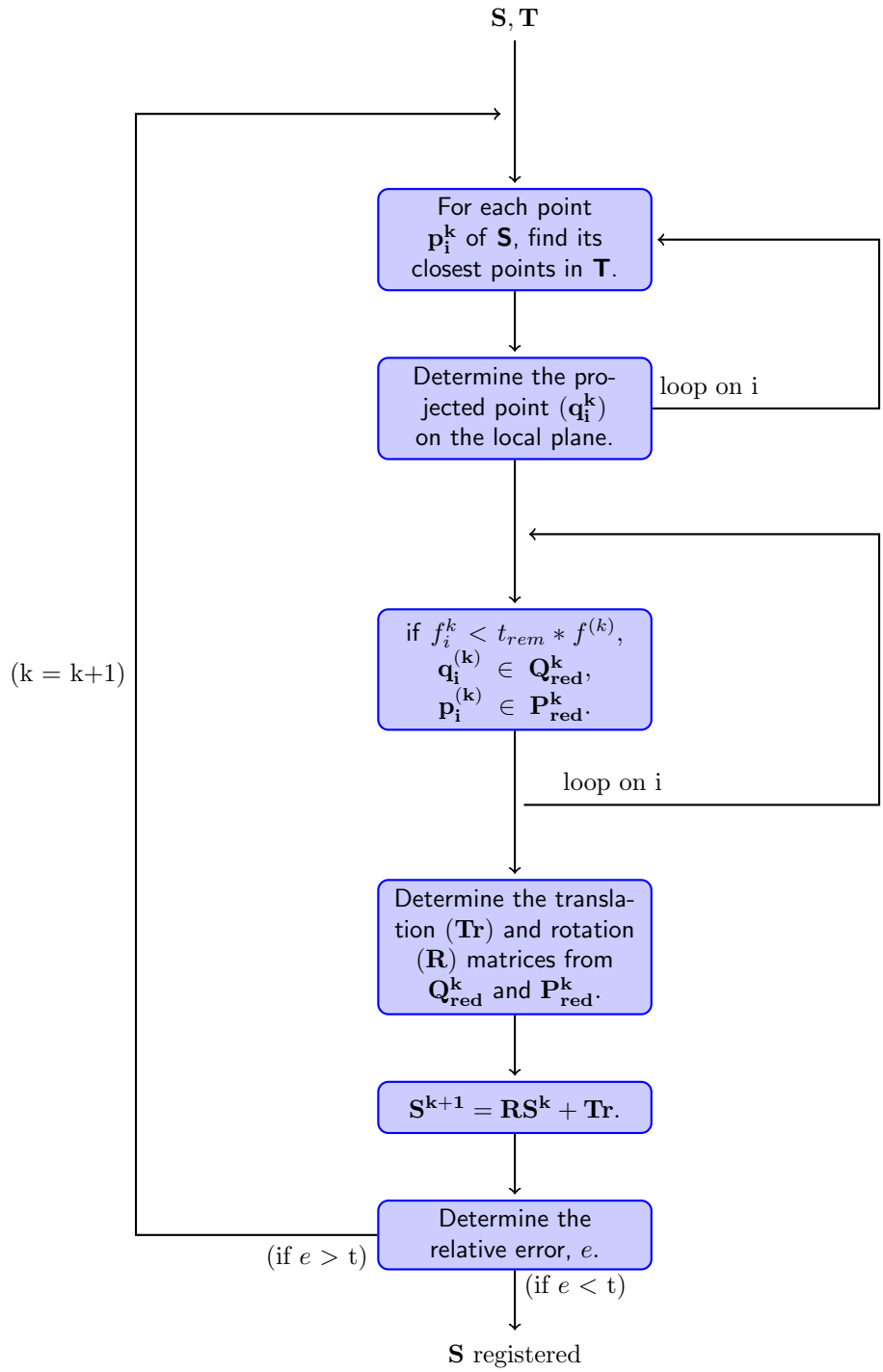

Figure 14: Updated scheme of the ICP algorithm

3.1 Results

The dental prosthesis used in the previous section is used as a reference. The first comparison is done between the previous results, i.e. no points are removed, and using $t_{rem} = 10$. The error comparison is shown in figure (15).

The case 2 is the case where no points are rejected and the case 1 is the one where the $t_{rem} = 10$.

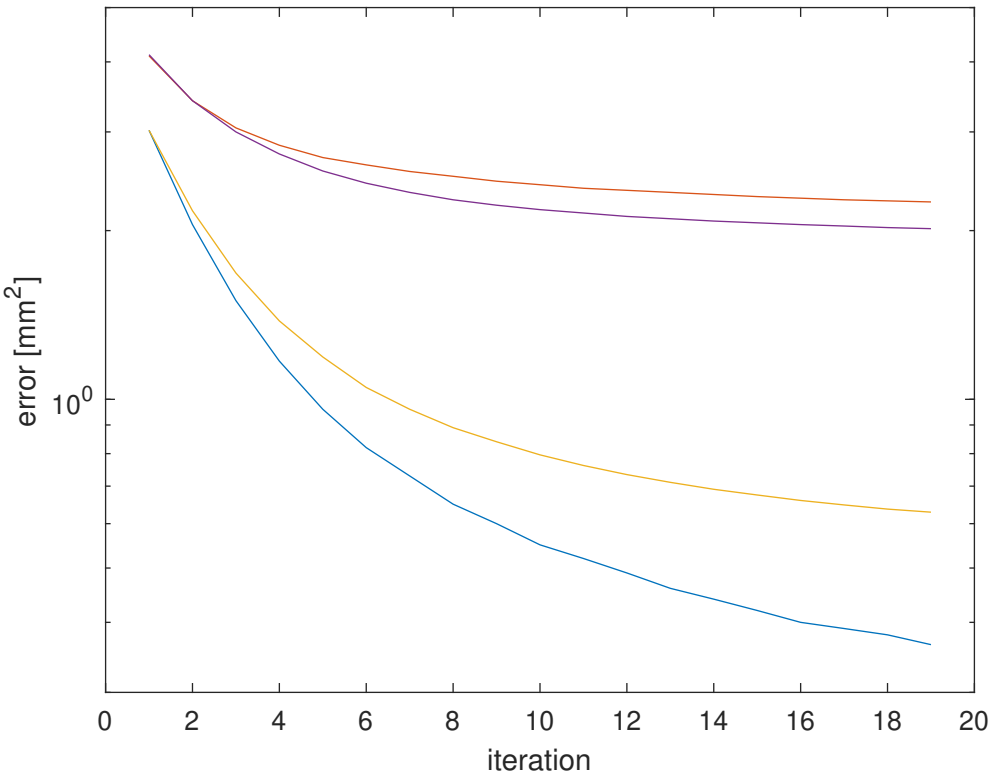

Figure 15: Error comparison

The total error is the error taking into account all the points from S and the reduced error is the error obtained if the points above the threshold are rejected. In order to have a good comparison, the reduced error of case 2 is determined with the exact same point as the reduced error of the case 1 as no point are rejected in case 2. From this plot, it can be deduced that, even though the total error obtained by removing points is higher, the reduced error is lower. It can thus be deduced that the ICP minimizes the error on the kept points.

Now that we have shown that the error on the kept points is minimized using the ICP, two different thresholds are to be compared. The first one is $t_{rem} = 10$ and the second one is $t_{rem} = 1.5$. The matching of the first one is shown in figure (16) and (18). The matching of the second one is shown in figure (17) and (19).

From these figures, it can be deduced that the threshold of 1.5 gives better a matching than the threshold of 10. It can be seen that many non corresponding points are still taken into account when the threshold is 10. The threshold must be modified until the added regions visually seems to be removed. This step obviously relies on human decision as it will decide whether or not the threshold gives satisfying results. Poor decisions might lead to removing either too many or too little points from the matrices and error estimation. Finding a way to perform this step automatically might be an interesting improvement of the algorithm but it has not been done in this study.

Figure 16: $t_{rem} = 10$: side view

Figure 17: $t_{rem} = 1.5$: side view

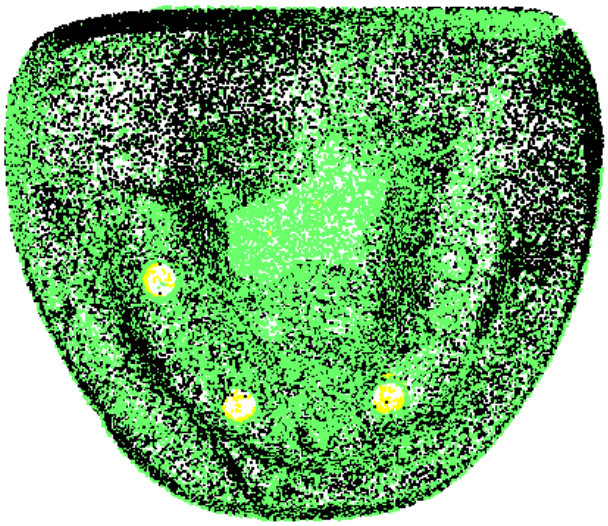

Figure 18: Threshold = 10 : top view

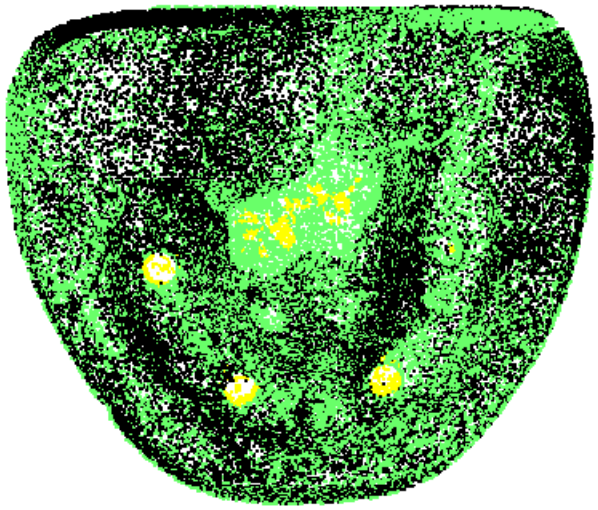

Figure 19: Threshold = 1.5 : top view

3.2 Limitations

In order to determine the limits of the algorithm, a test case is done by cutting into empreinte plâtre 1. The two compared models are shown in figure (20) and (21). The idea is to represent a CT image with a lot of information missing due to limitations of the scans for example. For this purpose, the cutted piece, which is called empreinte plâtre 1 coupée, is the source point cloud. The target point cloud is empreinte plâtre 2.

3.2.1 CDB limitations

When the coarse registration is applied to the 2 point clouds , the alignment is only based on the centroid and on the principal axes. In the registration of empreinte plâtre 1, even though the 2 point clouds did not represent exactly the same geometry, the centroid and the principal axes were close to one another. In this case, as one side of the source geometry is missing, this has a huge impact on the location of the centroid and the orientation of the principal axes. In this case, the detection of the correct mirror is made impossible. In fact, the error of the CDB is minimized by a non-physical mirror transformation of the original source as the rough matching is not accurate enough. This kind of issue is shown in figure (22).

Figure 20: Target : Empreinte plâtre 2

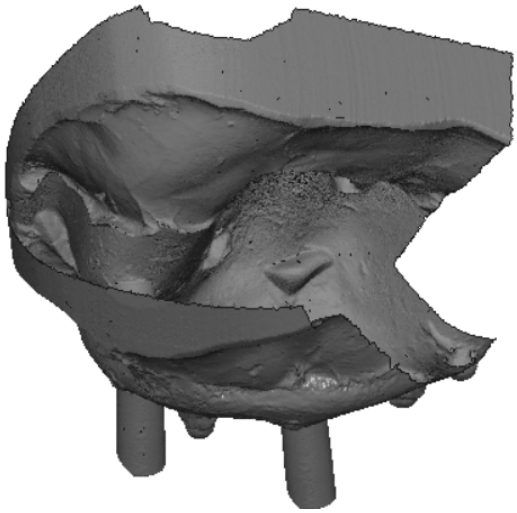

Figure 21: Source : empreinte plâtre 1 coupée

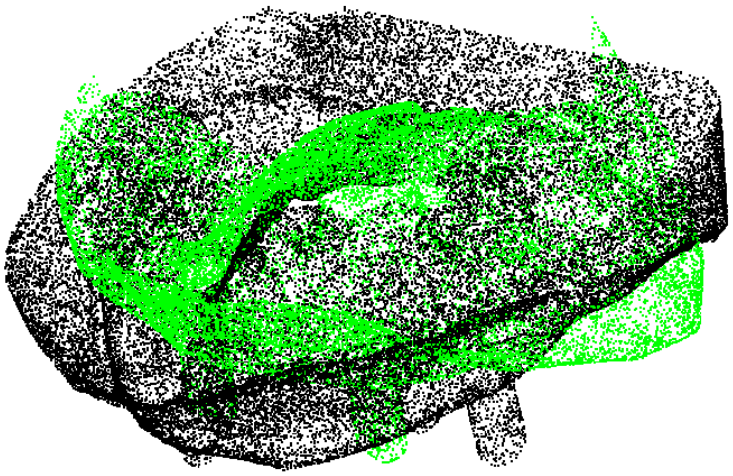

Figure 22: Initial matching from the CDB

Therefore, if the ICP is applied to the resulting point clouds, it might converge to a local minimum which does not correspond to a physical match. Moreover, as this example is almost symmetric, one may think the matching is correct as it is shown in figure (23) and (24). The expected matching is performed by manually overlaying the 2 point clouds. The ICP is then performed.

The convergence of the ICP is shown in figure (25). The number of iterations was set to 200 to better highlight the local minimum. Here, it can clearly be seen that the residual error is above the expected error. Unfortunately, the expected error is not known *a priori*. Meaning that it can be difficult to distinguish local minimum and real minimum in some cases. The process might become even harder with pieces containing even more symmetries. Once again, the verification analysis relies on human decisions. Certain algorithms may be developed to take care of the verification but it has not been done in this study.

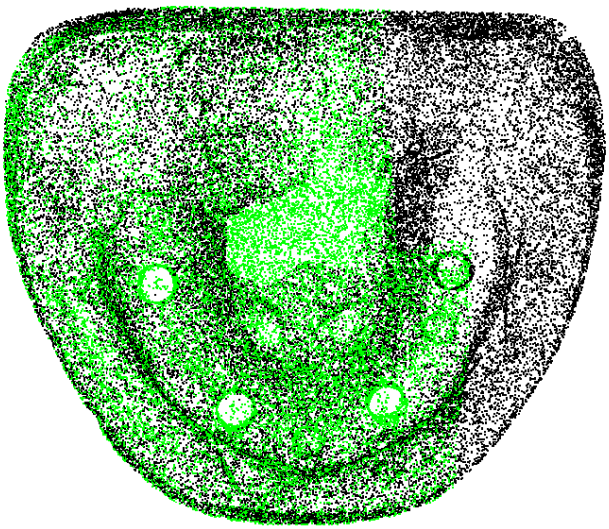

Figure 23: Expected matching

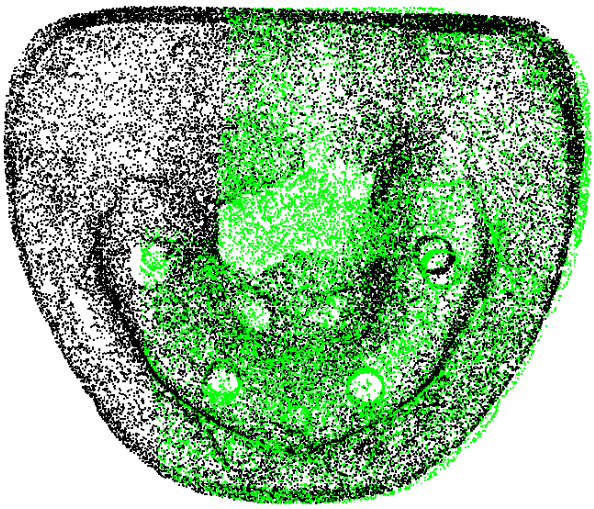

Figure 24: Matching after CDB followed by ICP

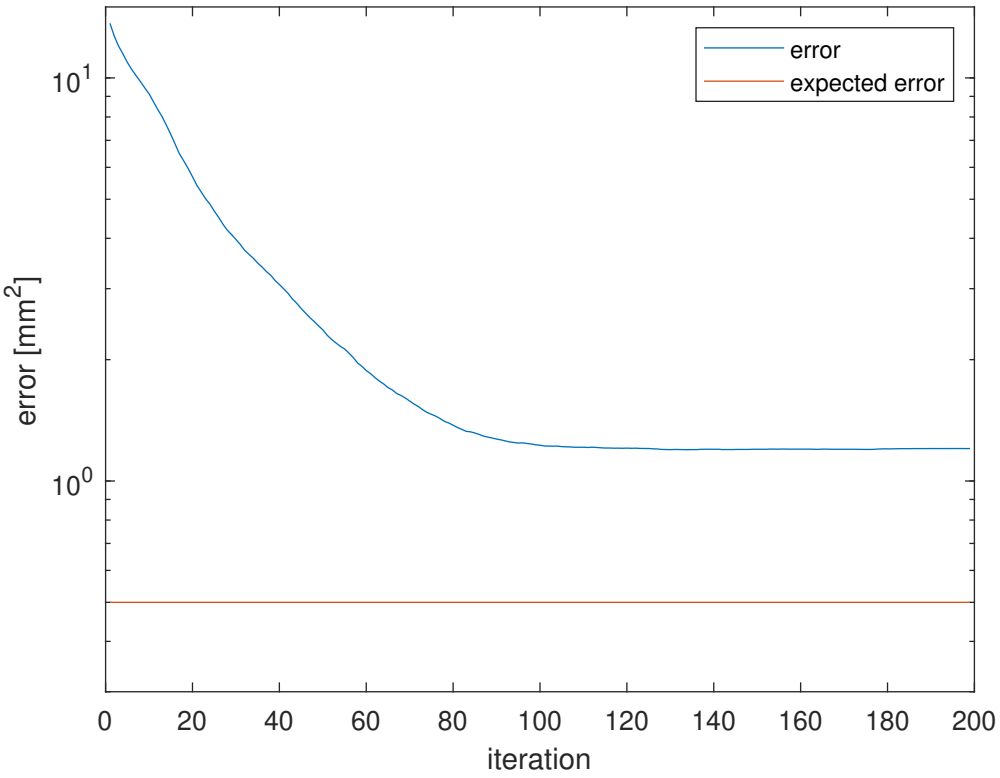

Figure 25: CDB limitations : convergence of the ICP

3.2.2 ICP limitations

The first test of the ICP is to apply the algorithm without any rough matching. In this case, as the correspondence in between points is poorly done, the algorithm needs a huge number of iterations to converge to a minimum. Moreover, this minimum is most likely a local minimum that corresponds to a poor matching of the clouds. The matching obtained is shown in figure (26). The

convergence is shown in figure (27). The results obtained depend vastly on the starting orientation of both points clouds. If the ICP is applied without any sort of pre-matching, the registration will randomly pick up the closest local minimum to the starting point. This leads to unpredictable and unusable results.

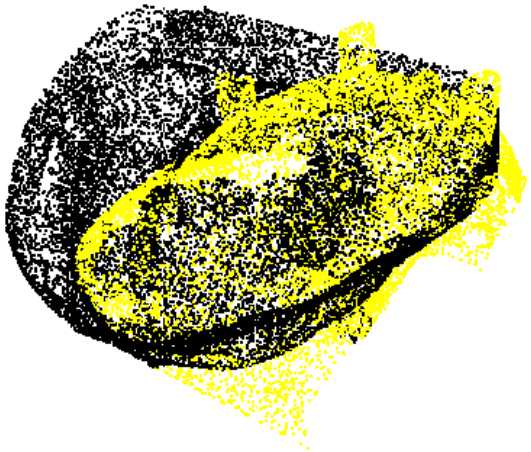

Figure 26: ICP limitation test 1 : matching

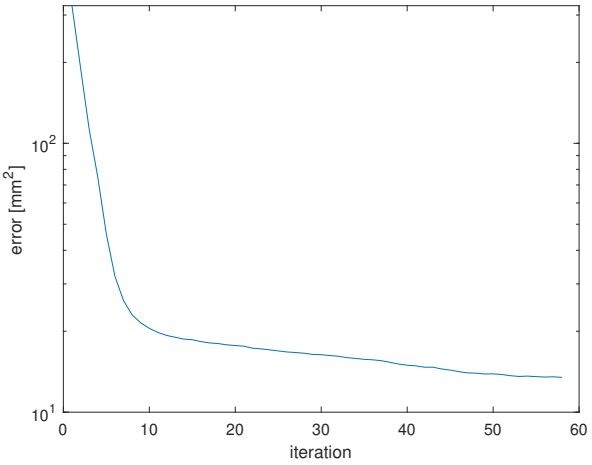

Figure 27: ICP limitation test 1 : convergence

The second test is to roughly overlap the 2 point clouds manually to check if the ICP can converge to the local minimum when the data sets have a lot of differences. In order to show the robustness of the ICP, a rotation and a translation are applied from this manual matching. The starting point and the matching are shown respectively in figure (28) and (29). As the starting point is far from the minimum, 250 iterations were performed. The convergence is shown in figures (30).

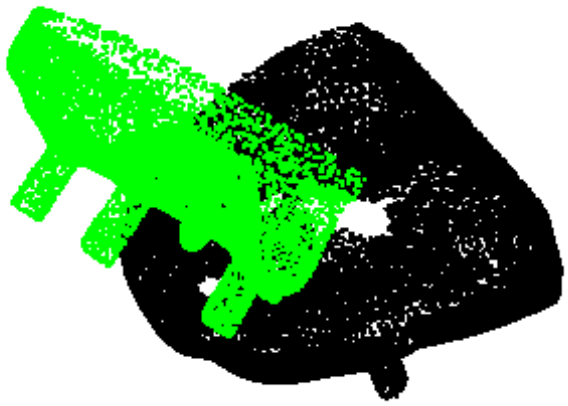

Figure 28: ICP limitation test 2: Starting point

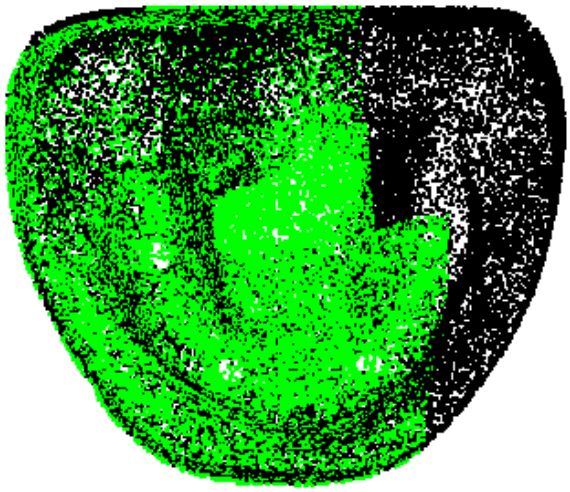

Figure 29: ICP limitation test 2 : Matching

It can be seen that even though the starting point is far from being ideal, the ICP can converge to the correct minimum after a large number of iteration. Obviously, crossing a local minimum or not depends on the geometry of the pieces but also on the starting point. The closest the

starting point is to the real minimum, the more likely it will converge to it. Moreover, it will lead to a smaller computational cost as it requires less iterations. These results show that a manual matching might be enough to get good results out of an ICP algorithm. It also shows that the ICP can converge if there are many missing parts on the source point cloud as long as the rough matching is done correctly.

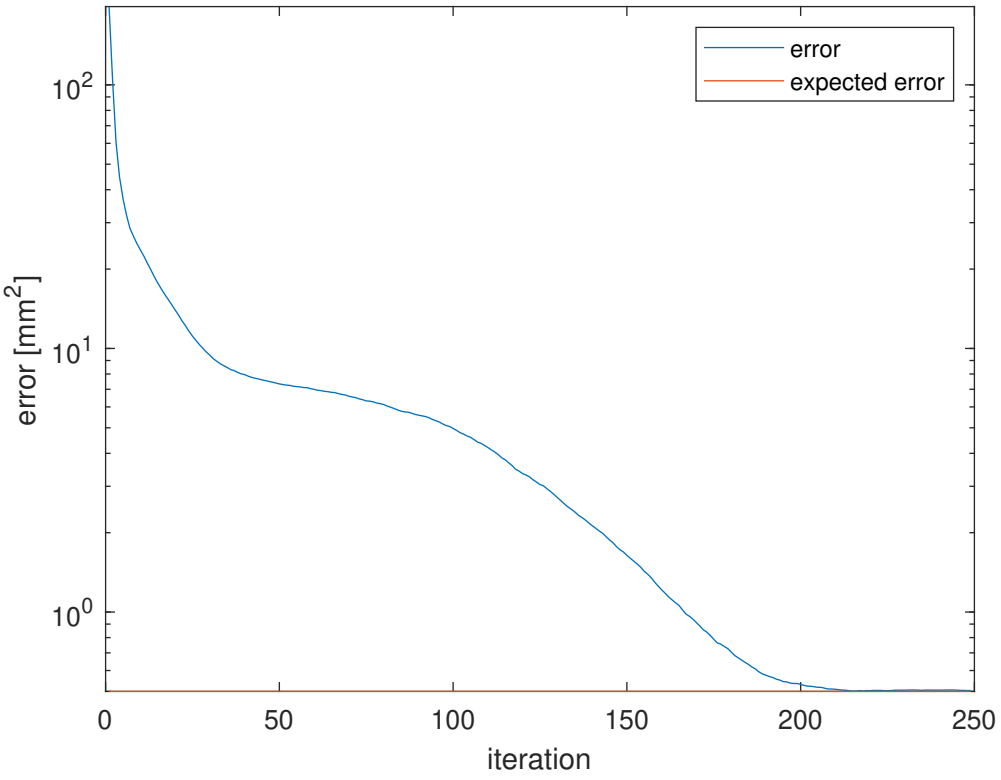

Figure 30: ICP limitation test 2: Convergence

4 Improvements of the Coarse registration

From the previous sections, it can be deduced that as long as the CDB gives a good coarse registration, the ICP will be able to converge to the correct minimum. In fact, the ICP was robust enough to handle slight body modifications and slight misalignments. In this section, different methodologies are tested to improve the robustness of the coarse registration.

4.1 Improvements of the coarse registration using geometrical features

The first tried methods rely on geometrical features of the point clouds. In fact, it is possible to determine geometrical properties at a given point $\mathbf{p_i}$ by looking at its neighboring point in its point cloud. These properties can then be used to determine pairs of corresponding points. In this section, the 10-30 closest points to $\mathbf{p_i}$ are used to determine the properties.

4.1.1 Algorithm 1

In this section, the source point cloud and the target point cloud are respectively called $\mathbf{Q^{(1)}}$ and $\mathbf{Q^{(2)}}$.The presented algorithm comes from [20]. It is based on 2 geometrical parameters, the curvature parameter and the angle parameter. The scheme of the algorithm is presented in figure (31). The first step is to determine the curvature and the angle parameter of each points contained in both point cloud $\mathbf{Q^{(k)}}$. For a query point $\mathbf{q_i^{(k)}} \in \mathbf{Q^{(k)}}$, its m closest points, $\mathbf{g_{i,l}^{(k)}}$, are defined in equation (14).

$$\mathbf{g_{i,l}^{(k)}} = \min_{\mathrm{m}} dist(\mathbf{g_{i,l}^{(k)}}, \mathbf{q_i^{(k)}}). \tag{14}$$

The curvature parameter can now be determined from equation (15),

$$\tau_i^{(k)} = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}, \tag{15}$$

where $\lambda_1 \leq \lambda_2 \leq \lambda_3$ are the eigenvalues of the local covariance matrix $\mathbf{C_i^{(k)}}$ which is defined as in equation (16).

$$C_i^{(k)} = \begin{bmatrix} \mathbf{g_{i,1}^{(k)}} - \bar{\mathbf{g}}_\mathbf{i}^{(\mathbf{k})} \\ \dots \\ \mathbf{g_{i,m}^{(k)}} - \bar{\mathbf{g}}_\mathbf{i}^{(\mathbf{k})} \end{bmatrix}^T \begin{bmatrix} \mathbf{g_{i,1}^{(k)}} - \bar{\mathbf{g}}_\mathbf{i}^{(\mathbf{k})} \\ \dots \\ \mathbf{g_{i,m}^{(k)}} - \bar{\mathbf{g}}_\mathbf{i}^{(\mathbf{k})} \end{bmatrix}, \tag{16}$$

where $\bar{\mathbf{g}}_\mathbf{i}^{(\mathbf{k})}$ is the mean value of $\mathbf{g_{i,l}^{(k)}}$. The normal vector ($\mathbf{n_i^{(k)}}$) at a given point $\mathbf{q_i^{(k)}}$ is given by the eigenvector of $\mathbf{C_i^{(k)}}$ corresponding to its lowest eigenvalue, i.e. λ_1. The normal vectors can now be used to determine the second feature parameter, the angle parameter. The angle parameter can be determined from equation (17),

$$\Omega_a(\mathbf{q_i^{(k)}}) = \sum_{l=1}^{m} \theta_{i,l}^{(k)}, \tag{17}$$

where $\theta_{i,l}$ is the angle between the normal at the query point and the normal at its l^{th} neighbor. This angle can be obtained from equation (18),

$$cos(\theta_{i,l}^{(k)}) = \frac{\mathbf{n_i^{(k)} n_l^{(k)}}}{|\mathbf{n_i^{(k)}}||\mathbf{n_l^{(k)}}|}. \tag{18}$$

These 2 parameters are first used to reduce the data. In fact, it is difficult to distinguish points in regions where the curvature and the angle parameters are low by only using those parameters. Only the reduced data is used to determine the translation and rotation matrices. The points of interest are points with high curvature and high angle parameter, which are more likely to be feature points. For each point of both point cloud, its characteristic parameter thus need to be determined. The characteristic parameter is defined as shown in equation (19). If a point has a characteristic equation below a certain threshold (t_{char}) the point is not used to determine the translation and the rotation matrices. The 2 reduced point clouds are called $\mathbf{Q_{red}^{(1)}}$ and $\mathbf{Q_{red}^{(2)}}$.

$$\omega(\mathbf{q_i^{(k)}}) = \Lambda \tau_i^{(k)} + \Omega_a(\mathbf{q_i^{(k)}}), \tag{19}$$

where Λ is the surface curvature coefficient.

From those 2 parameters, it is now possible to find common pairs of point between the 2 reduced point clouds. The algorithm first tries to find, for all points from $\mathbf{Q_{red}^{(1)}}$, the point in $\mathbf{Q_{red}^{(2)}}$ that minimizes the function shown in equation (20) . Then, if the difference between the 2 feature parameters of these 2 points is below a certain threshold, the points are used as corresponding points. The threshold used for the curvature parameter is called t_{curv} and the threshold used for the angle parameter is called t_{ang}.The translation and rotation matrices can then be determined from equation (10) and (11).

$$f_{i,j} = \sqrt{(\tau_i^{(1)} - \tau_j^{(2)})^2 + (\Omega_a(\mathbf{q_i^{(1)}}) - \Omega_a(\mathbf{q_j^{(2)}}))^2}. \tag{20}$$

Results

This methodology is to be tested on 2 cases. The parameters used in those tests are $t_{char} = 2$, $t_{curv} = 0.0001$, $t_{ang} = 0.1$ and $\Lambda = 200$. 10 neighbors were used to determine the feature parameters.

The pieces used in the first case are empreinte plâtre 1 (**T**) and empreinte plâtre 1 coupée (**S**). Those 2 pieces can be seen in the previous sections. In this case, the matching obtained after applying the CDB is shown in figure (32). The pieces used in the second case are empreinte plâtre 2 (**T**) and empreinte plâtre 1 coupée (**S**). The matches that were obtained are shown in figure (33).

From these results, it can be seen that this approach is not robust when it is subjected to noise. In fact, the points used in the first case are exactly the same as empreinte plâtre 1 coupée is the same as empreinte plâtre 1 with a missing part. In this case, the angle and curvature parameters are to be approximated with the same error on both point clouds. In the second case, due to the random distribution of the points in both point clouds, the angle and curvature parameters of common points are to be approximated differently. In this case, the points that minimize the distance function might be point located in another region of the cloud. An algorithm presented in [21] can be used to reduce the noise and to detect properly the common features between point clouds. However, this was not tested in this study.

4.1.2 Algorithm 2

The second methodology tested to improve the coarse registration is only based on the curvature. The curvature can be deduced once again from equations (14), (15) and (16). A certain predefined number (n_{reg}) of sub point clouds ($Q_j^{(k)}$) are to be determined for both point clouds. Each sub point clouds contains the highest curvature region of the piece. The sub point cloud is determined on the basis of equation (21), where t_{reg} is a threshold to determine if the points share a common curvature. The CDB algorithm is then applied to the resulting point clouds. The sub point clouds with an error higher than a certain threshold (t_{rem}) are then removed and considered non corresponding features, and iterations are made until convergence of the algorithm. The error of a sub point cloud is defined as the mean error of all the points it contains. The scheme of the algorithm is shown in figure (37).

$$\begin{cases} t_j^{(k)} = \max_{\mathrm{j}} \tau_i^{(k)}, \\ \mathbf{q_i^{(k)}} \in \mathbf{Q_j^{(k)}}, \ \ if \ \ |\tau_i^{(k)} - t_j^{(k)}| < t_{reg}. \end{cases} \tag{21}$$

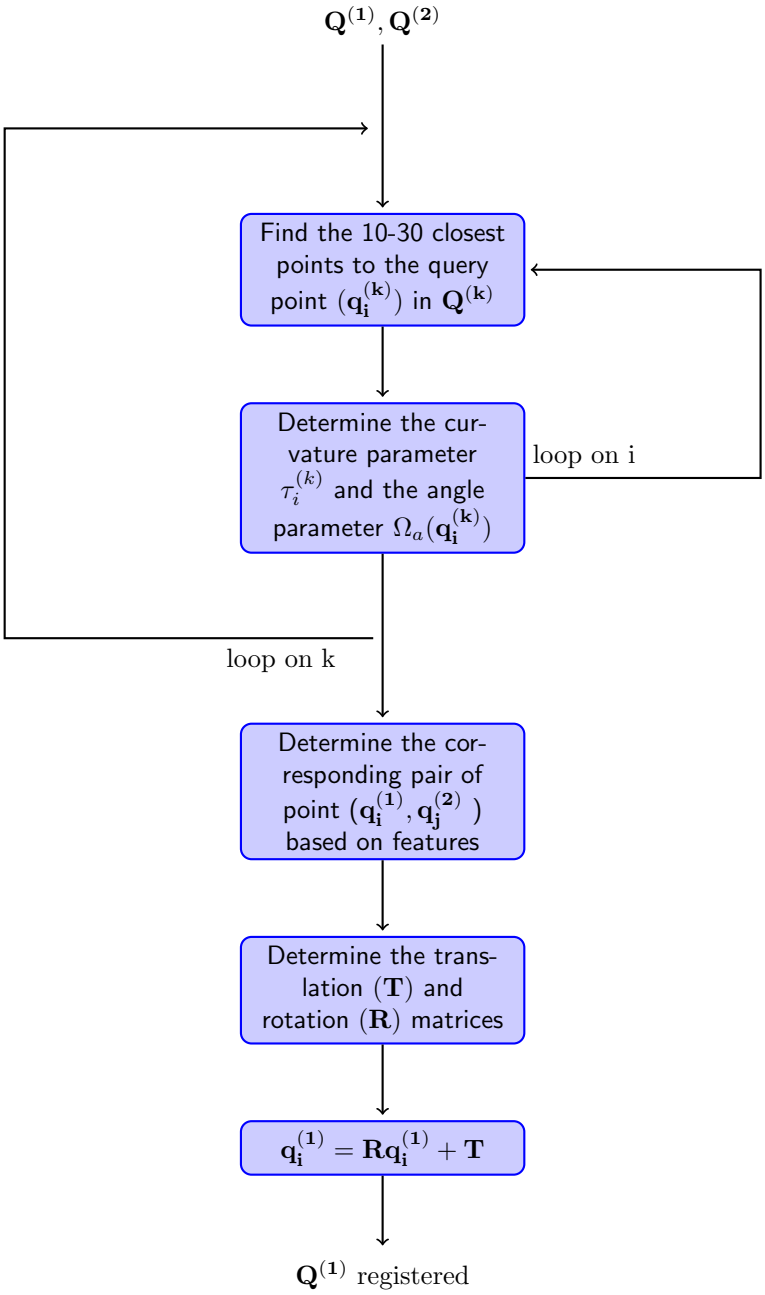

Figure 31: Scheme of the coarse registration using features

Results

The first case used to test the algorithm is the simple model build with NX (see section 2.3.1). The number of sub point clouds is set to 10, t_{reg} is equal to 0.005 and the number of neighbors used to determine the curvature is set to 15. The high curvature region of the target point cloud are shown in black in figure (34). The low curvature region are shown in yellow. The matching obtained when the algorithm 2 is performed is shown in figure (35). The black points correspond to $\mathbf{T}$ and the blue points correspond to $\mathbf{S}$

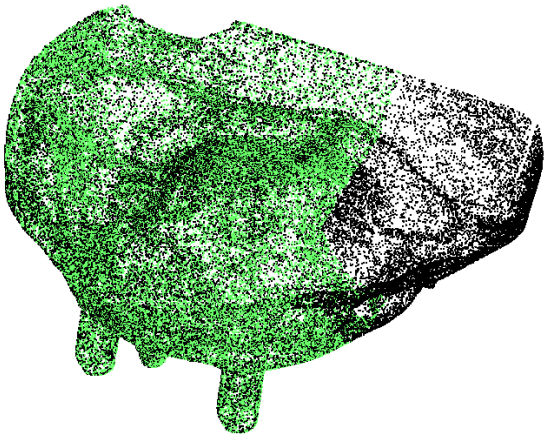

Figure 32: Coarse registration using features : Case 1

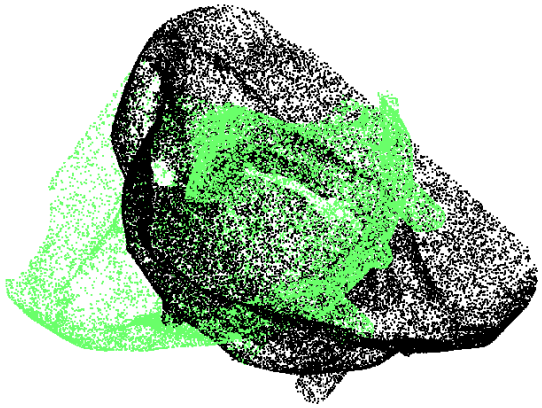

Figure 33: Coarse registration using features : Case 2

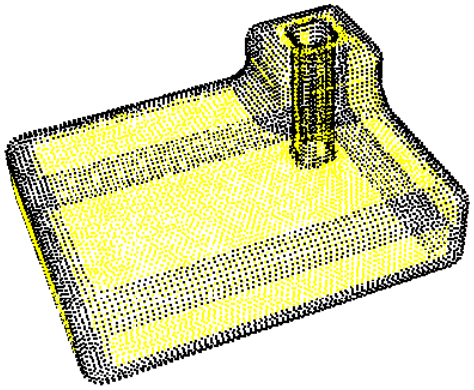

Figure 34: Sub point clouds of $\mathbf{T}$

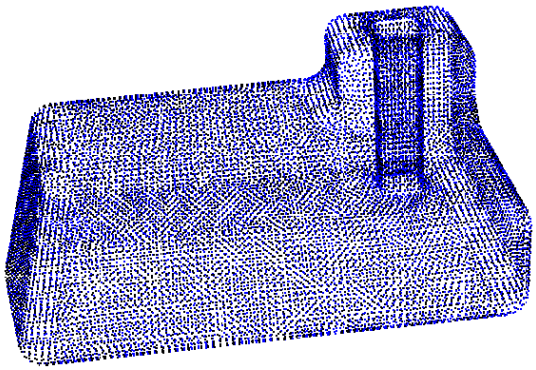

Figure 35: Coarse registration using curvature : Case 1

In the second case, empreinte plâtre 1 coupée and empreinte plâtre 2 were used. The matching after applying the algorithm 2 is shown in figure(36).

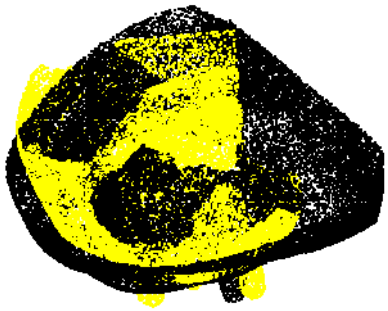

Figure 36: Coarse registration using curvature : Case 2

Once again, it can be seen that this approach is not robust enough to noise. In fact, as the

feature points are roughly the same in simple cases such as case 1, the CDB at the first iteration converges and gives good results. Unfortunately, when parts are missing, this also means some sub point clouds do not correspond. At the first iteration of the algorithm, the CDB is not necessarily able to determine the correct mirror transformation. In this case, the algorithm may remove corresponding sub point clouds and keep non corresponding sub point clouds as the error is poorly determined. The action of removing sub point clouds must thus be done before applying the CDB to the point clouds instead of through iterations. Such methods have been briefly tested but without relevant results.

4.2 Improvement of the CBD using multiple ICP

The last algorithm tested to try to improve the robustness of the coarse registration relies on multiple ICP which we call CDBm. In fact, the CDB is first applied to the source point cloud. Then, the ICP is applied to all the possible mirrors. The mirror that minimized the error after convergence of the ICP is kept as the correct registration S. Unfortunately, this approach is time consuming as an iterative process has to be performed multiple times. In order to reduce the computation cost, the point clouds can be reduced and the threshold used can be larger. The translation and rotation matrices can then be used on the complete point clouds to perform the coarse registration. The scheme of such an algorithm is shown in figure (38). Finally, an ICP algorithm with tighter threshold can be applied on the complete point clouds to obtain the fine registration.

4.2.1 Results

This algorithm is applied to empreinte plâtre 2 (T) and empreinte plâtre 1 coupée (S). In this example, the point clouds are not reduced during the CDBm and the thresholds used are $t = 0.005$ and $t_{rem} = 10$. The matching of those 2 is shown in figure (39) and the convergence analysis is shown in figure (40). For the sake of simplicity, only one of the 7 incorrect mirrors convergence curve is used as a comparison to the correct mirror.

It can be seen that the algorithm is now able to detect the correct mirror transformation and thus converges to the correct minimum. However, as the 2 geometries to match have huge differences due to the cut, the mirror transformation only gives more opportunities to converge to the correct minimum instead of a local minimum. It can be seen from the convergence analysis that the correct mirror nearly gets stuck on a local minimum. However, as the minimum was small, the algorithm was able to go through it. This means that the algorithm still has some limits when large part of pieces are removed. Nevertheless, this algorithm seems robust enough for registration of CT images with respect to CAD models. If the algorithm were to provide incorrect results, this would mainly be due to the bad quality of the CT images.

5 Determination of the defects and deformations

The last step of this work is to find defects or regions of high deformations in the workpiece. In fact, the error used to perform the ICP analysis can easily give the distance between the CT image

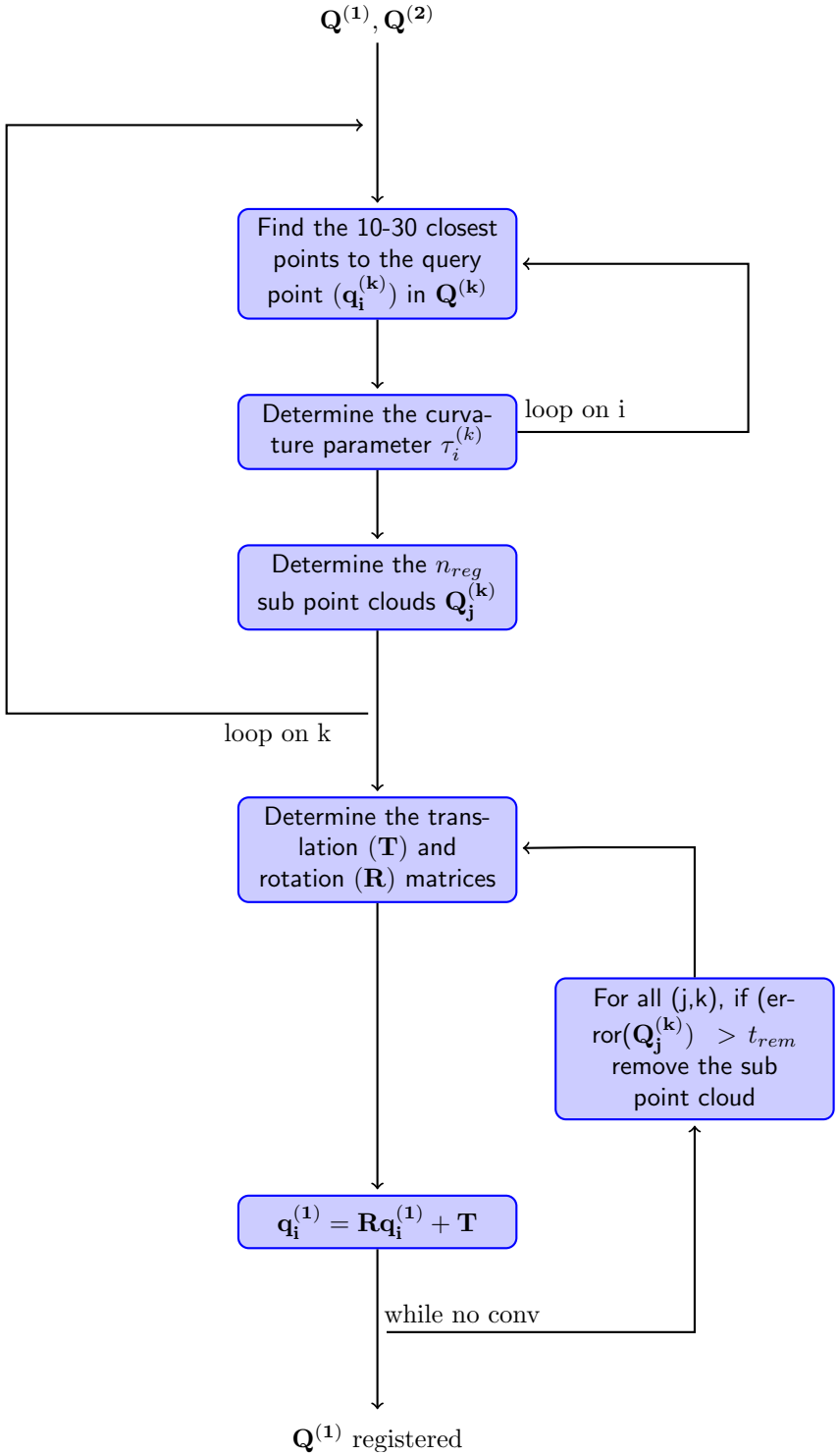

Figure 37: Scheme of the coarse registration using feature region

and its projection on the CAD point cloud as the local error is simply its square distance. However, this error is only an approximation of the real deformation at a given point of the workpiece as the surface of the target model was approximated using planes. This kind of approximation mainly

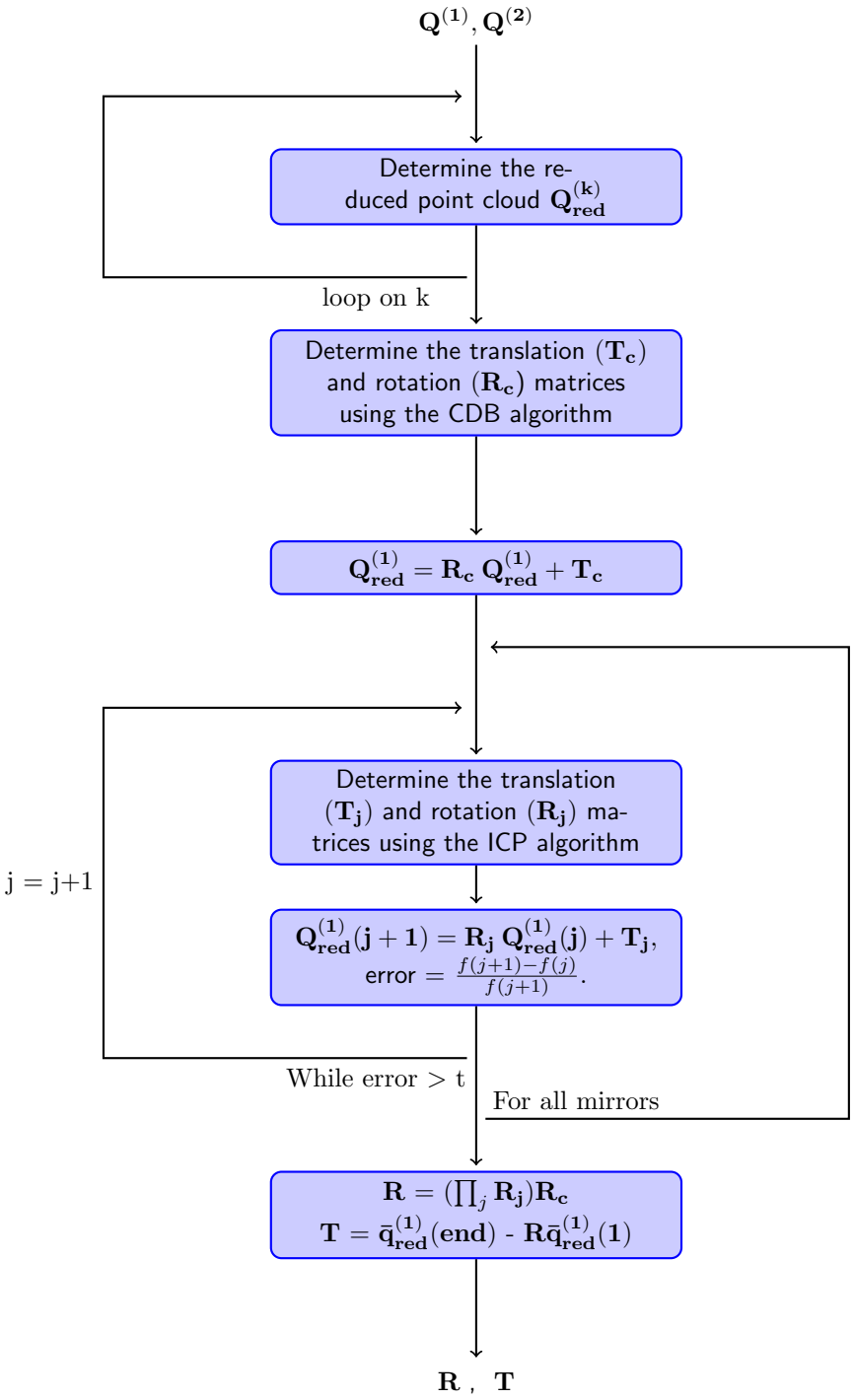

Figure 38: Scheme of the coarse registration using multiple ICP

affects regions of high curvature and low point density.
The results are displayed using a color graduation depending on the distance of each point to its projection on the model. This distance is the square root of the error previously computed.

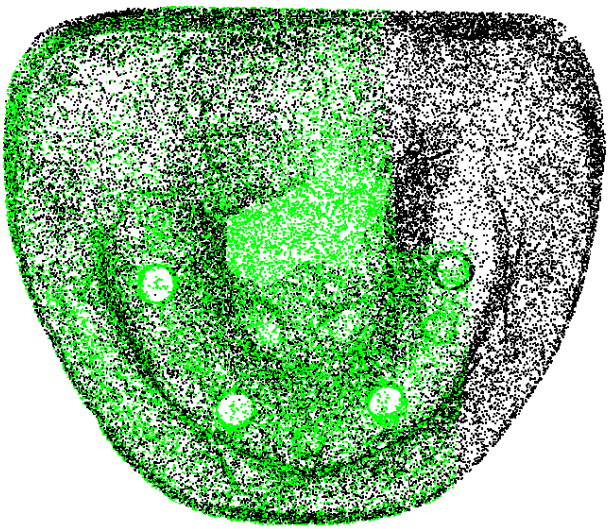

Figure 39: Coarse registration using multiple ICP

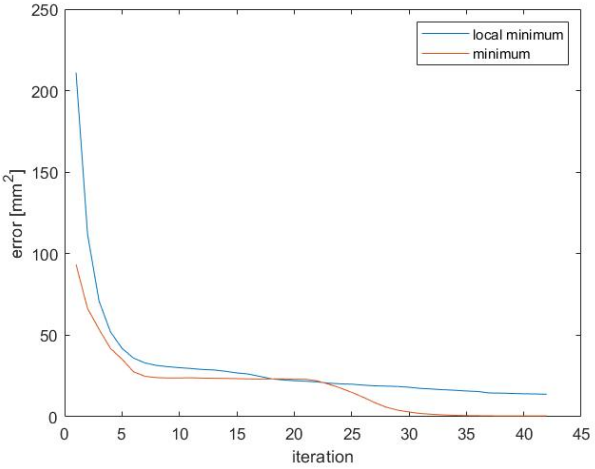

Figure 40: Convergence analysis

The defects are shown on 5 different cases. The first one uses the NX models. The second one correspond to the dental prosthesis models, empreinte plâtre 1 and empreinte plâtre 2. Then it is performed on a symmetrical and an asymmetrical mechanical model. The last case is a spherical model.

5.1 NX models

The first step is to perform the deformation analysis on an undeformed model. The NX models represent the exact same geometry. In this case, there should be no deformations at all. The observed deformation comes from numerical errors, either because of a non perfect matching of the registration or because of a poor determination of the ICP local error. The deformations of the source point cloud are shown in figure (41) and (42).

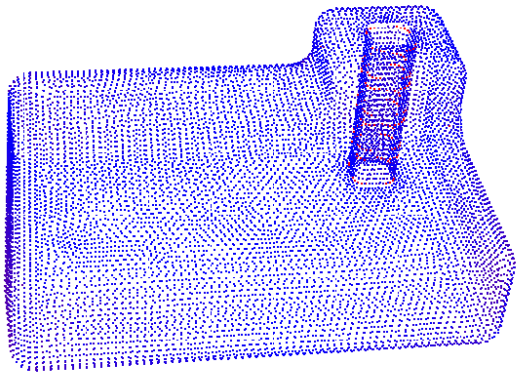

Figure 41: Deformation analysis of the NX models

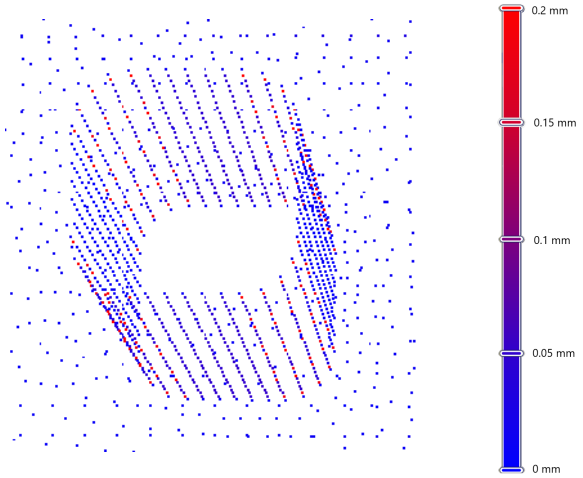

Figure 42: Zoom : Deformation analysis of the NX models

It can be seen that the error is poorly estimated on the highly deformed points as those errors

are local and no error was expected. This means the algorithm had some difficulties determining the error on the fillets radius due to high curvature. Moreover, the point density of this model is low, leading to a worse point to plane approximation. Even if the maximum error is of only 0.2 mm, which is low compared to the dimensions of the part (order of magnitude 100mm), these points can not be used as they might be out of tolerances. This may lead to the unwanted rejection of a piece.

Thus, another analysis with the same sets of points is performed. This time the points with a poorly estimated error are rejected during the ICP registration in order to remove the computational error made in this example by tuning t_{rem}. This methodology is not ideal as information is removed from the CT image. The results are presented in figure (43).

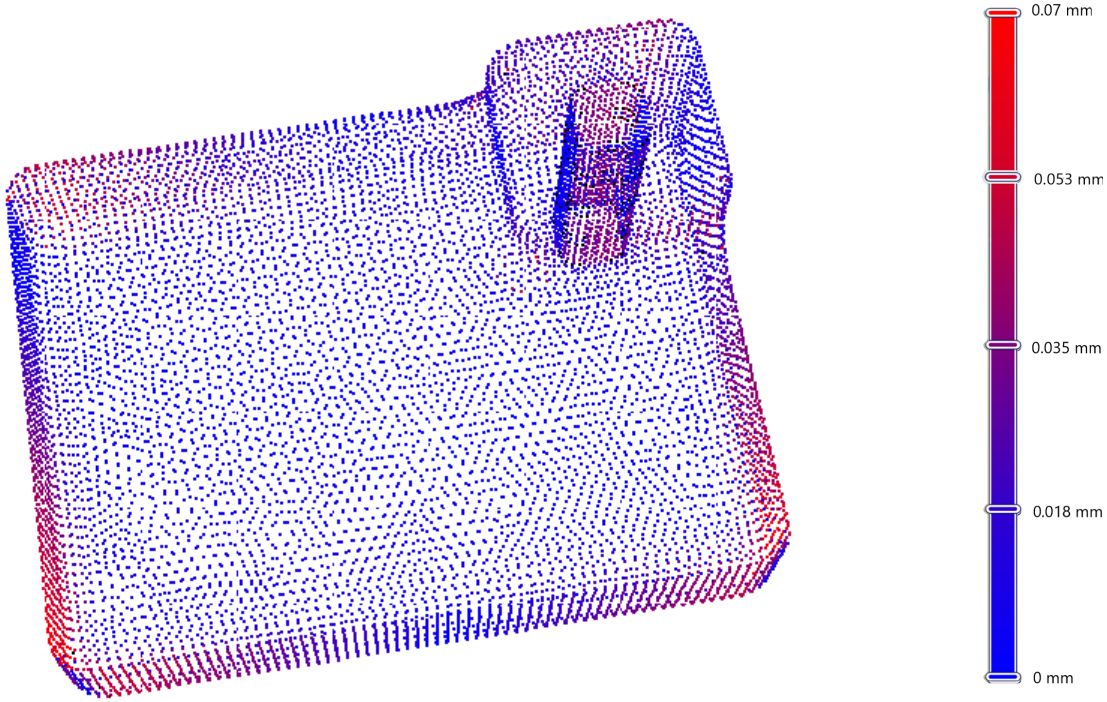

Figure 43: Deformation analysis of the NX models (2)

In this case, the matching of the ICP provides better results, leading to a better determination of the error as the points with a poorly determined error are rejected. These points are shown in black. It is now possible to see some deformed regions. However, the ICP can never converge to a perfect registration of the points cloud due to the planar approximation. In this case, a slight residual rotation can be seen. In practice, one should make sure the precision of the ICP is small enough to detect the deformations of interest.

The results of this subsection show the limitations of the algorithm. An approximation of the error using points to planes projection might not be enough to check the high curvature region of the workpiece. In this case, it is easy to tell when the error is poorly approximated as there is no geometrical error between the 2 models. In the case of real pieces, it might be harder to tell if the error is due to a local deformation or to a computational error. It would be interesting to build higher order local surfaces to give a better approximation of the projected error. Those kind of surfaces provide a better representation of the target geometry. Nevertheless, the error

on low curvature surfaces can be accurately estimated using points to planes projections. Thus, this algorithm could be used to check surfacing, for instance. Different distance estimations are discussed in section 7 to improve the current ICP registration.

5.2 Dental prosthesis models

The deformation analysis is now performed on empreinte plâtre 1 and empreint plâtre 2. In this case, there are additional parts on the source point cloud due to the fixation of the workpiece used to get the CT image. The results obtained are shown in figure (44).

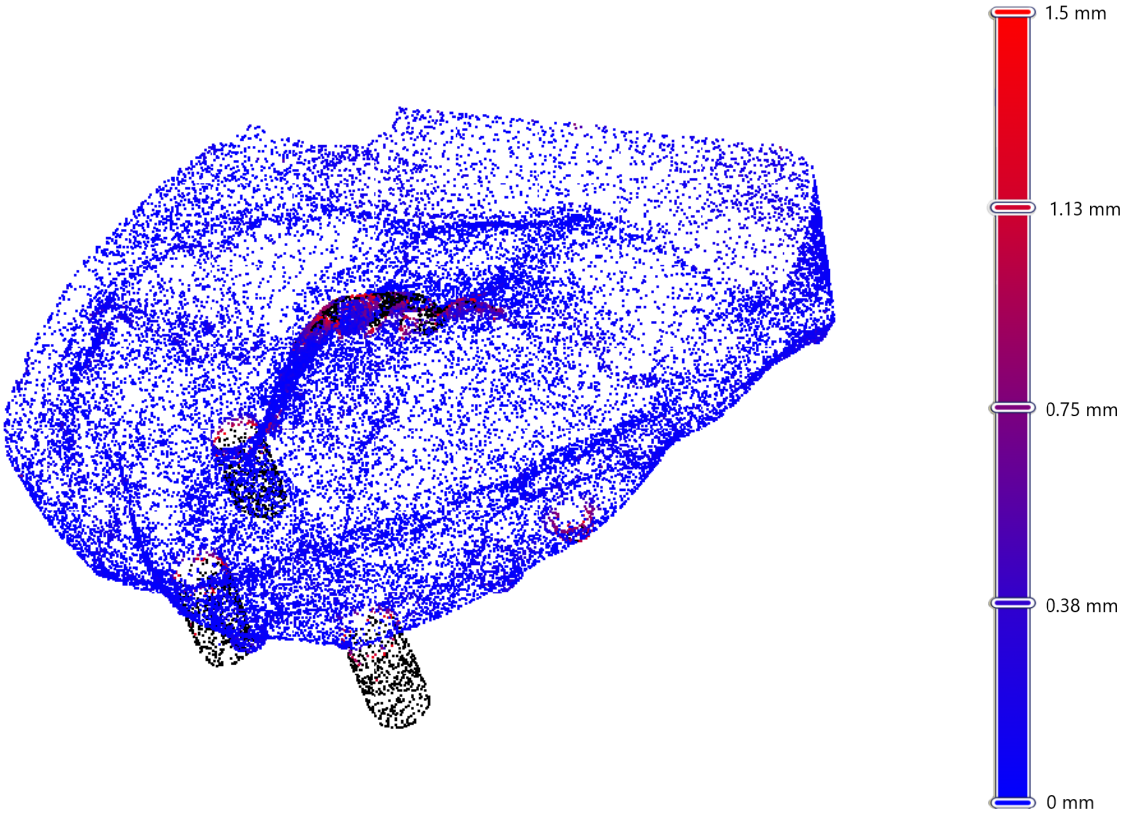

Figure 44: Deformation analysis of the dental prosthesis models

It can be seen that the additional points make it hard to visually see where the deformations are on the workpiece. In fact, in this case, only additional points have large deformations. Unfortunately, the ICP algorithm does not evaluate errors properly if $t_{rem} < 1$: this means points can not be further removed using the ICP. The reason of this bug must still be determined. Thus, the graphical scale has to be modified. The registration is slightly worse as some non corresponding points can not be removed but the readability of the deformations is improved. Such a case is presented in figure (45).

In that case, it becomes difficult to tell if the points of high deformation are due to the numerical error or are due to real deformations in the workpiece as the maximum error is fairly low. However, the high deformation of some regions can be considered as surface deformation. These deformations can be considered as real deformations as the neighboring points are gradually deformed. Thus, it might be possible to filter the results to remove a single point with a high error based on its neighboring points. Such method reduces the noise of the solution. Once again, the use of

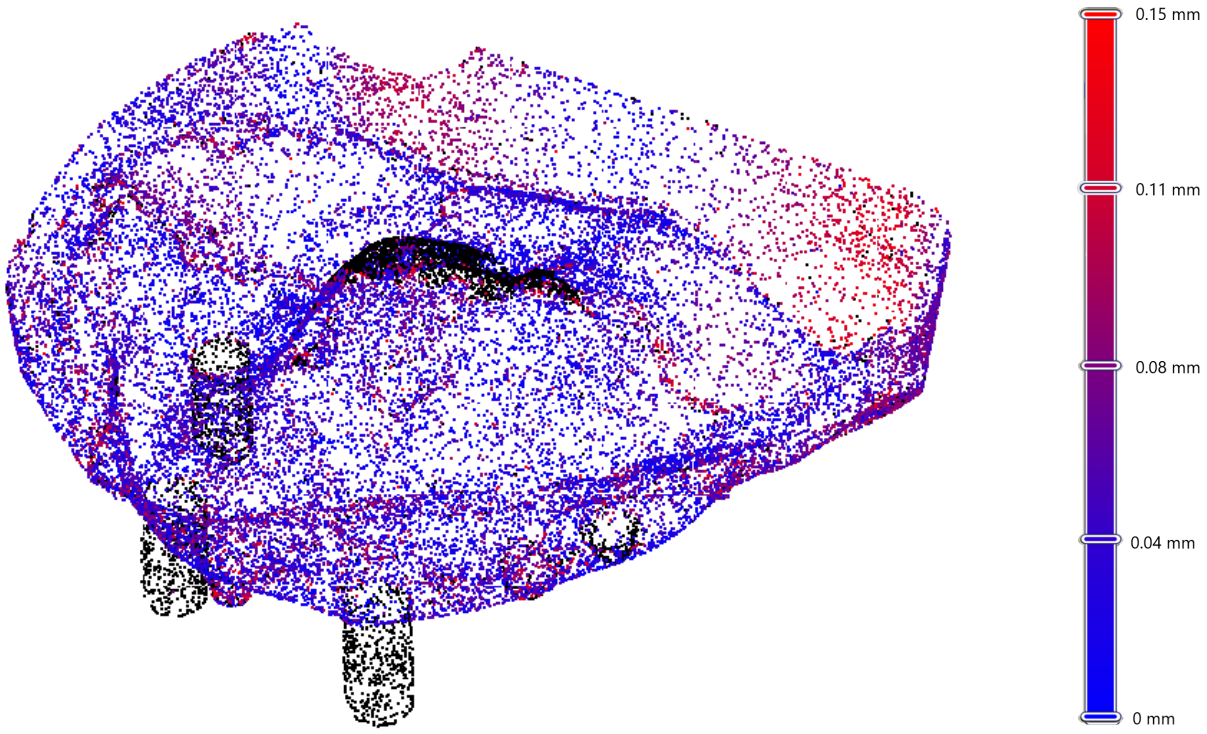

Figure 45: Deformation analysis of the dental prosthesis models (2)

projection on higher degree surface might lead to less numerical errors. However, it is impossible to determine how it changes the results as it was not tested in this work.

5.3 Symmetrical mechanical models

The next model used for the deformation analysis is called the symmetrical mechanical model. This model is shown in figure (46). In this case, a deformation is applied on a surface (1) of the source workpiece in order to simulate a deformation. This model come from [25].

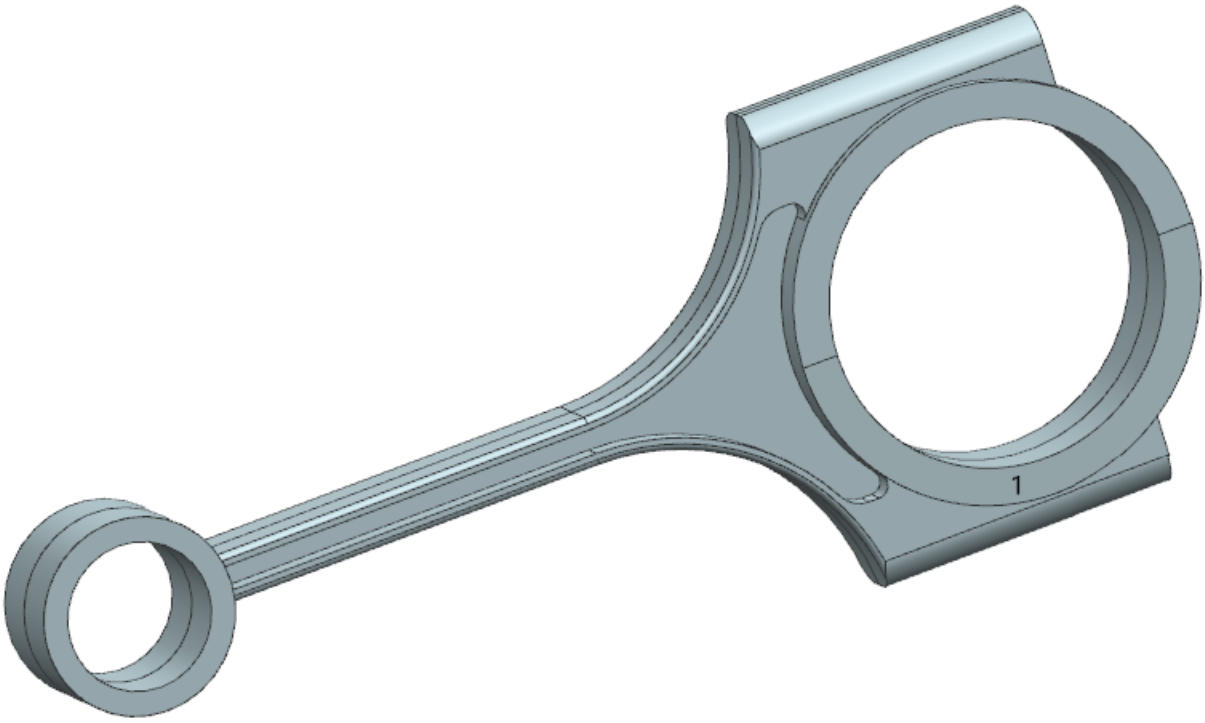

Figure 46: Symmetrical mechanical model

In this case, there are no added part or removed part on the workpiece. Thus, t_{rem} is set to 100000. Thus, no points are rejected. The threshold t is set to 0.0001. The results of the deformation analysis are shown in figure (47). The black points are points where the local error could not be estimated properly. Thus, they were removed from the registration. These points are located at the symmetry plane of the workpiece and are caused by a bad quality of the data set. However, the algorithm was still able to detect the deformed region of the workpiece. A singular highly deformed point can be seen by looking closely to the results. This point can be rejected as it is cased by a poor local estimation.

Unfortunately, as the workpiece is symmetrical, the CBDm cannot properly detect the correct mirror. In this case, 4 mirrors leads to similar errors: the difference between the 4 errors is caused by numerical estimations. 2 of them give correct results as they represent a 180 degree rotation. Unfortunately, the 2 others correspond to an nonphysical mirror of the workpiece. Those 2 are problematic as it modified the location of the deformed region. As previously stated, the non physical mirror can be rejected *a priori* but it was not tried in this work. In this case, it is easy to reject the non physical mirror manually as the deformed region is known a priori. In the case of a real CT image, it is impossible to determine where the deformed region is. The convergence of the 4 mirrors is shown in figure (48).

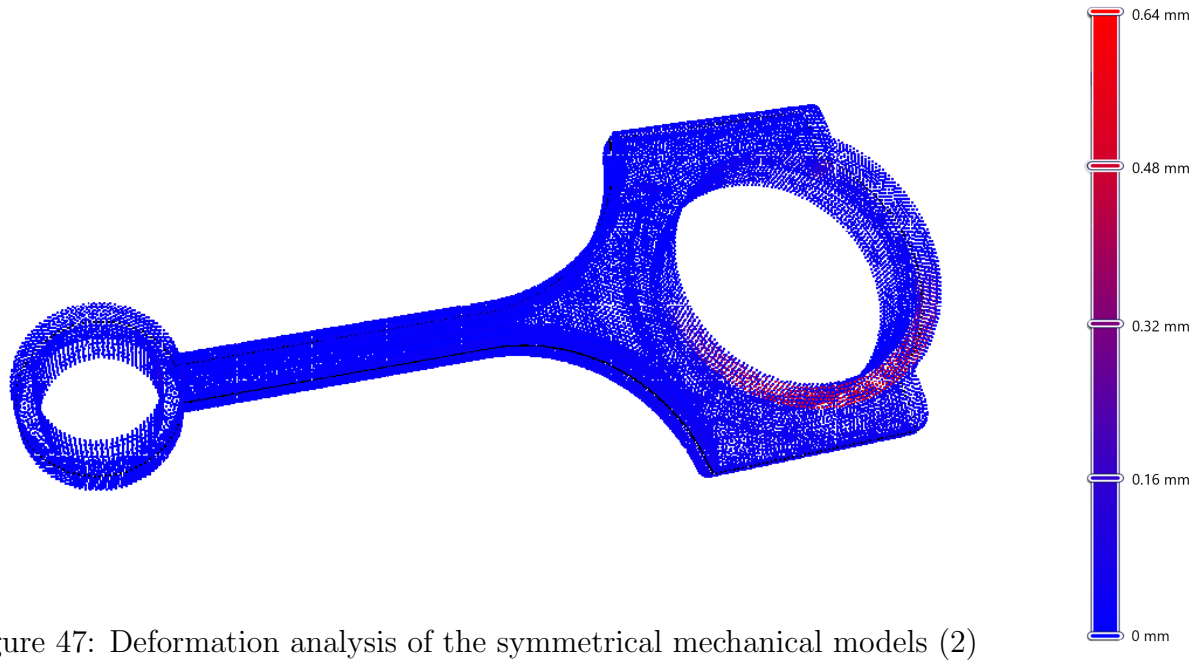

Figure 47: Deformation analysis of the symmetrical mechanical models (2)

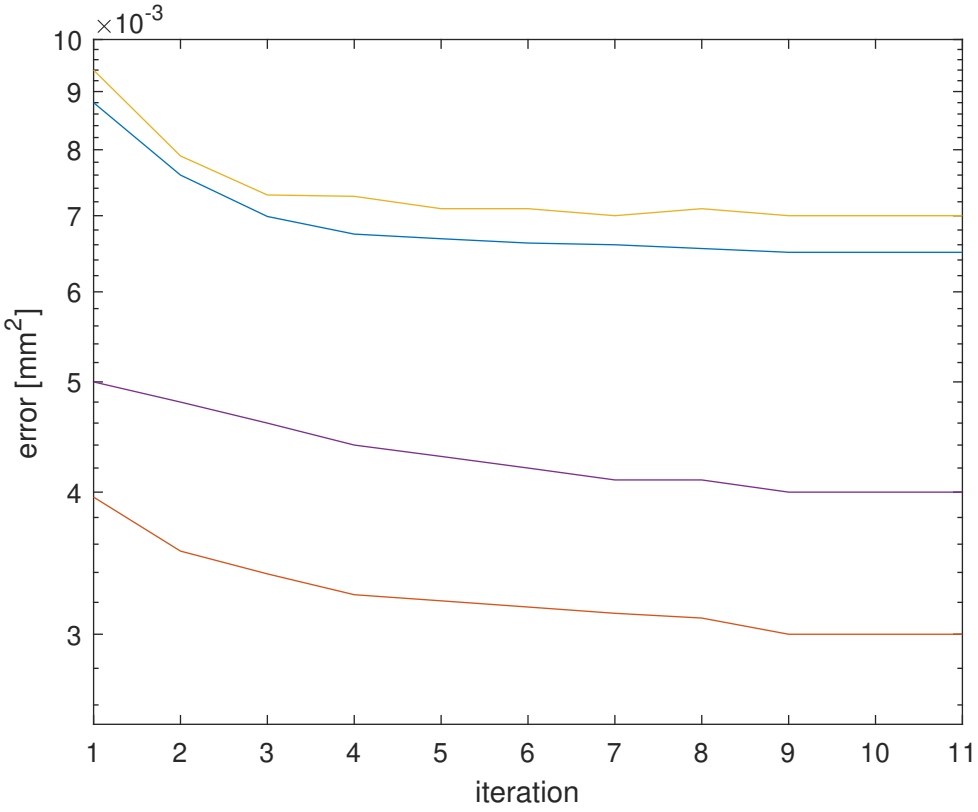

Figure 48: Convergence analysis of the symmetrical mechanical model

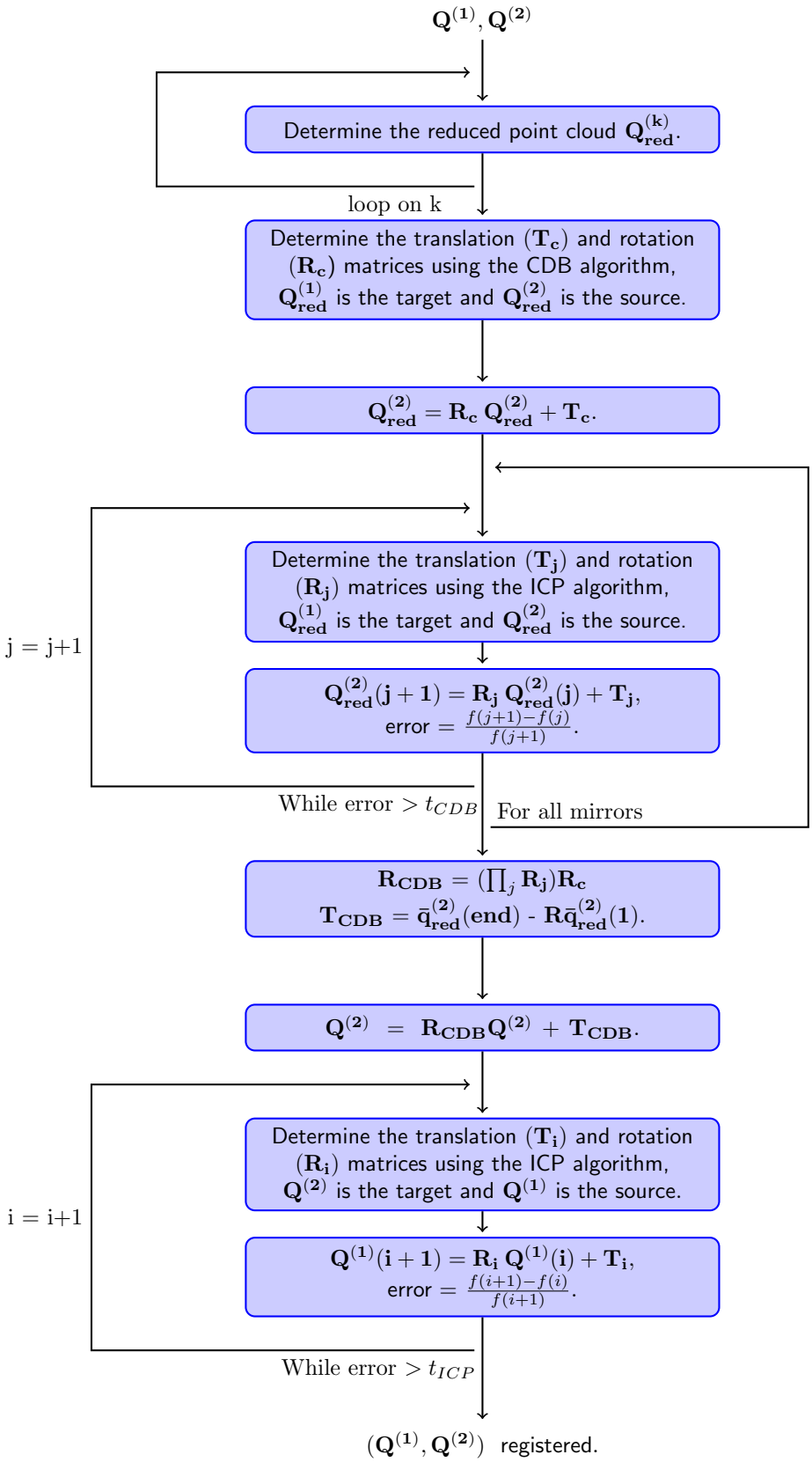

Figure 49: Scheme of the entire process

In order to get rid of the issues coming from non physical mirrors in the case of symmetric pieces, the procedure of the registration has to be changed. The CDBm algorithm using multiple ICP is now performed with the CT image as the target point cloud and the CAD point cloud as the source point cloud. In this case, the equivalent mirrors are applied to the CAD model, which is relevant as this model is symmetric. Then, the ICP is performed as previously, i.e. the CT image is used as the source point cloud. The ICP has to be applied with the CT image as the source point cloud as this algorithm gives the deformations $\mathbf{S}$ with respect to an approximation of $\mathbf{T}$. The procedure of the entire registration is shown in figure (49), where $\mathbf{Q^{(1)}}$ is the point cloud coming from the CT image and $\mathbf{Q^{(2)}}$ is the point cloud coming from the CAD model. t_{CDB} and t_{ICP} are respectively the convergence threshold of the CDBm and of the ICP.

The new procedure is tested on the symmetrical mechanical pieces with the same parameters as those used at the beginning of this section. As the point cloud coming from the CAD is used for the CDBm coarse registration as a source, the error is determined differently. The results show the error when the CAD model is projected on the CT image model. These results are not interesting to determine the deformation of the workpiece. However, it makes it possible to roughly match the 2 point clouds without applying any non physical mirrors on the CT image. Moreover, the coordinate system of the workpiece remains fixed during the CDBm. Therefore, the correspondence between the CT image and the real workpiece remains unchanged. The ICP is still applied with the CT image as a source, leading to a rotation and a translation of its system. However, those transformations remain low. The rotation and the translation applied to the scan of the symmetrical mechanical piece are shown in equation (22) as an example. It is thus easy for the user to determine which regions are deformed or not on symmetric pieces.

$$\mathbf{R} = \begin{bmatrix} 1 & 6.75e(-5) & -2.88e(-5) \\ -6.75e(-5) & 1 & 1.087e(-4) \\ 2.88e(-5) & -1.087e(-4) & 1 \end{bmatrix}, \quad \mathbf{Tr} = \begin{bmatrix} 1.099e(-3) \\ 7.68e(-3) \\ -3.07e(-3) \end{bmatrix}. \tag{22}$$

The convergence obtained for the 4 small mirrors is shown in figure (50). It can be seen that the difference between the errors is very tiny and only comes from the discretization using point clouds. Thus, the algorithm picks the one with the smallest error, even though this choice is random. However, the results obtained are equivalent as the CAD model is supposed to be perfectly symmetrical. The differences between the 4 results is only caused by slight discretization errors that emerged when the CAD model was converted to a point cloud. It is impossible to get rid of these kinds of errors in the case of numerical discretization. In order to show the equivalence of the results, each one of these mirrors was selected manually during the CDBm. Then the ICP was applied to the CDBm results. Each pictures is taken without modifying the view in the graphical viewer. Only 2 of the mirrors results are presented in figure (51) and (52) as the 2 others give similar results.

The results of the mirrors are similar. However, the error scale is not exactly the same. When looking closely, a singular high deformed point can be seen in the figure (52). This singularity is caused by a poor estimation of the error on that point. The scale is thus changed to the scale of mirror 1, with all the points above the maximum value being rejected, i.e. they are displayed in black on the matching shown in figure (53).

From these results, it can be expected that the algorithm will be able to detect the defects in a symmetric piece while keeping its coordinate system nearly constant. The results of the algorithm

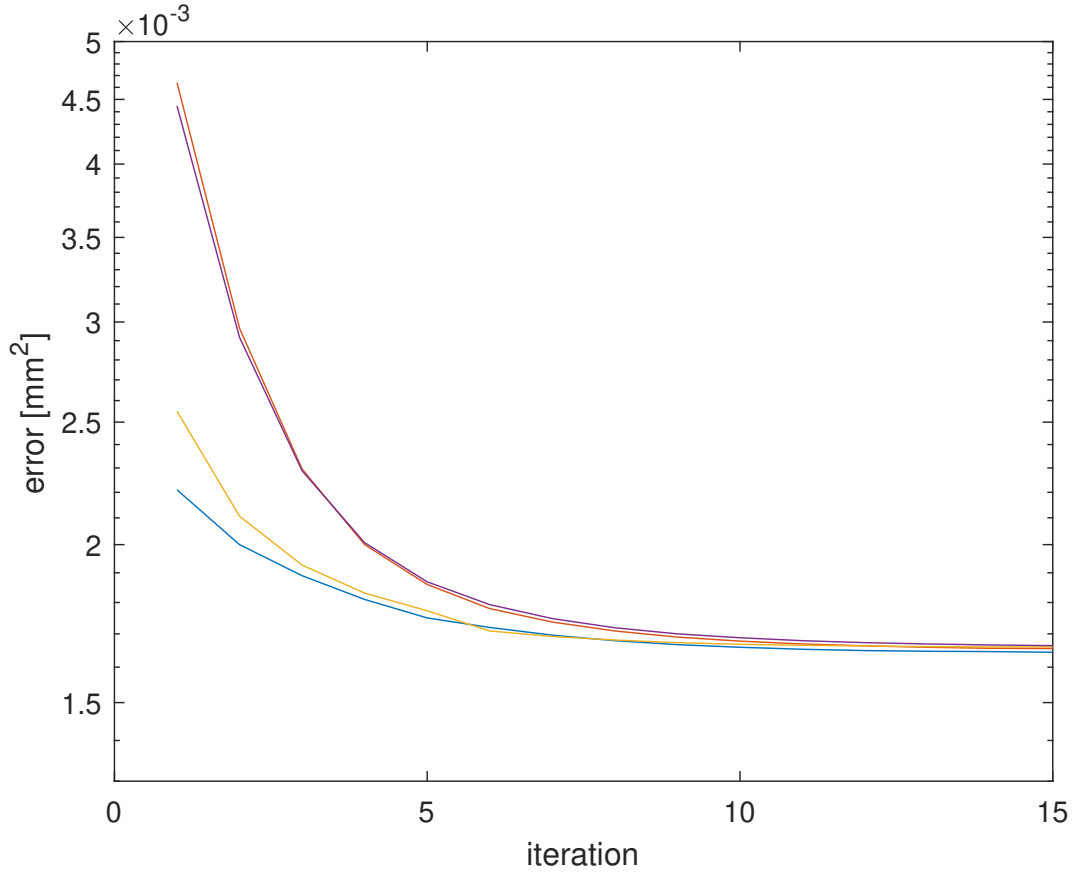

Figure 50: Convergence analysis of the symmetrical mechanical model using the updated procedure

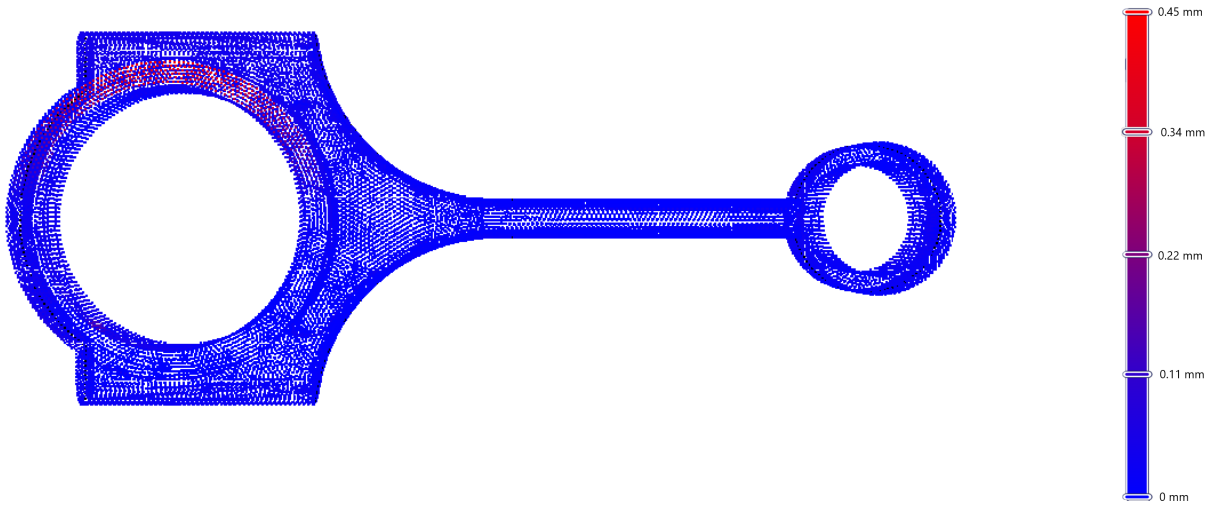

Figure 51: Mirror 1

can thus easily be used on symmetrical pieces. However, the mirror that minimizes the rotation angles the most is to be selected when the workpiece is considered symmetrical. This is based on

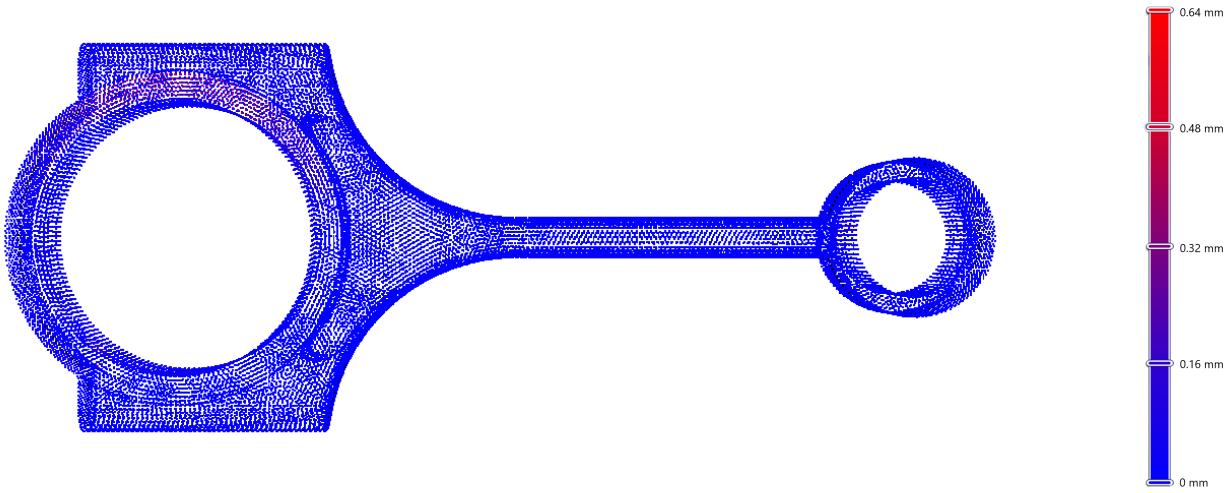

Figure 52: Mirror 2

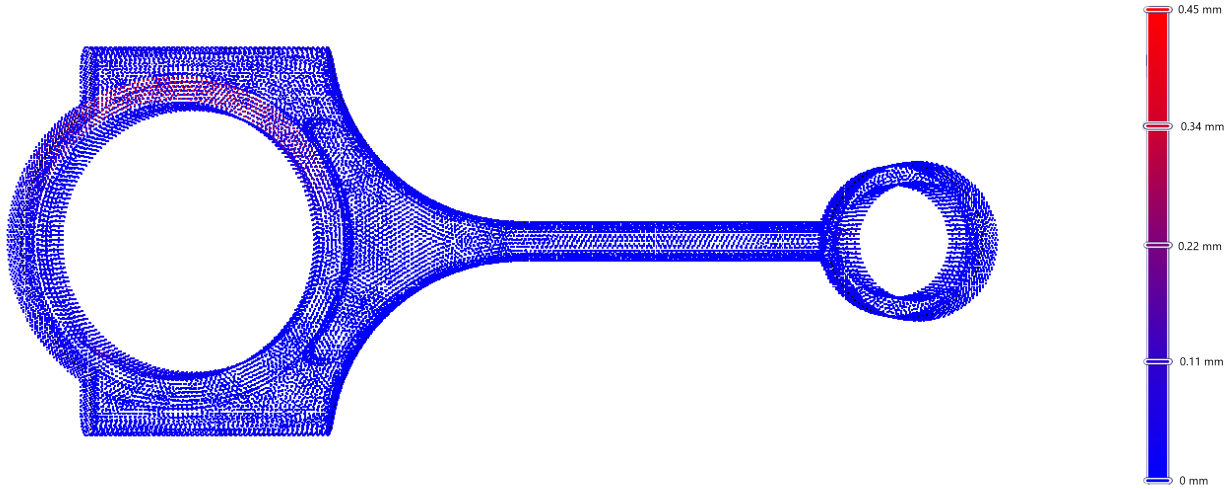

Figure 53: Mirror 2 with a modified scale

the hypothesis that the user places the target and the source close to the correct mirror in the case of symmetrical pieces. This is done to avoid the random picking of a mirror, leading to a better predictability of the results. However, the result should still be usable if the pieces are placed randomly, as the results of symmetrical pieces are similar. A piece is considered symmetrical when some of its mirrors converge to a close value of error. Further research could be done to determine what threshold should be used to consider 2 mirrors to be equivalent.

5.4 Non symmetrical mechanical model

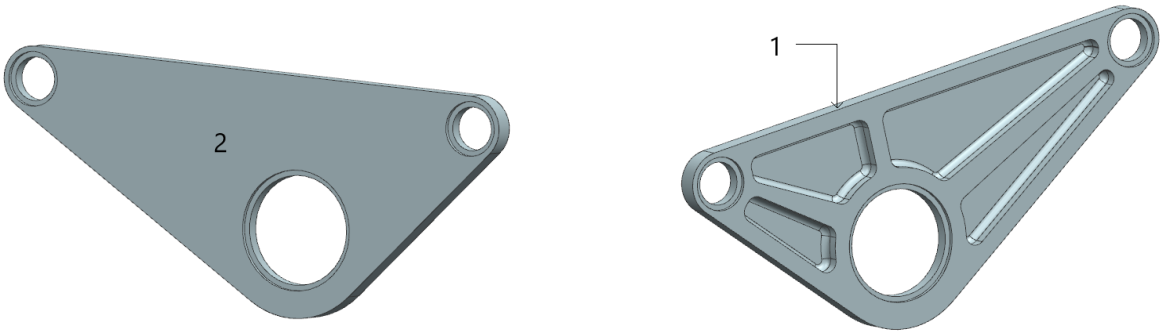

Figure 54: Asymmetrical mechanical model

The next model used to test the algorithm is the asymmetrical mechanical model which is shown in figure (5.4). Two tests are performed, the first one with a deformation on face (1) and the second one with a deformation applied on face (2).

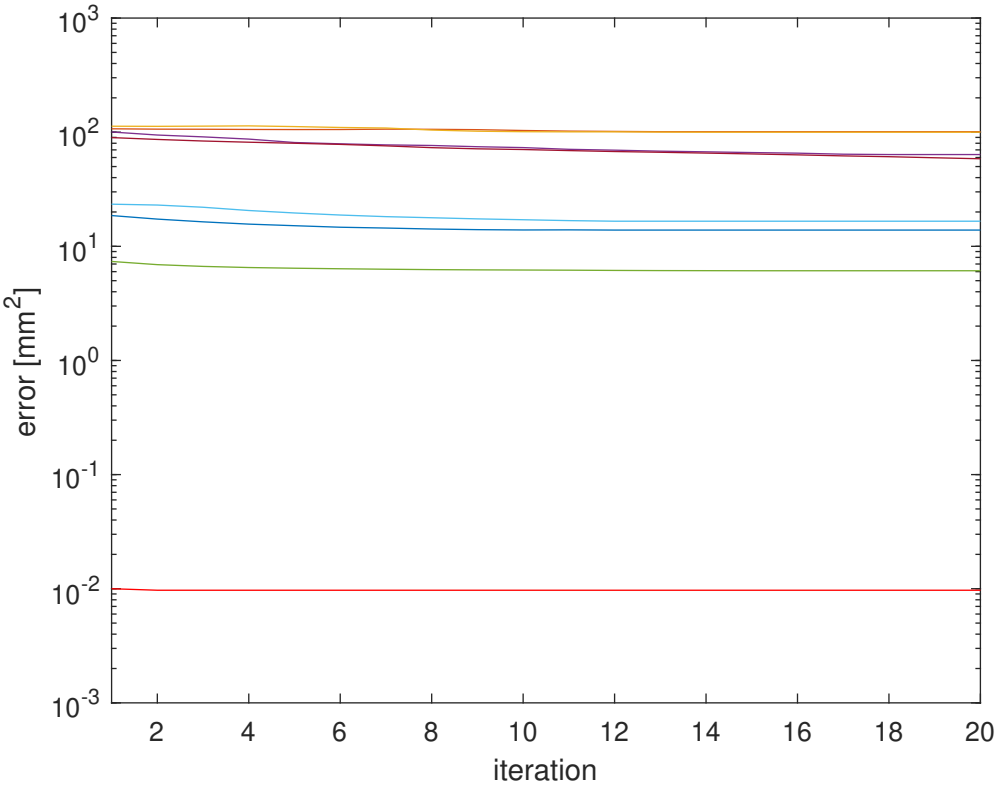

Figure 55: Convergence analysis of the asymmetrical mechanical model

In the first test, a slight deformation is applied on a limited region of the model. In this case, as the piece do not have any symmetries, the right mirror will be selected properly. The convergence

analysis of the mirrors obtained from the multiple ICP is shown in figure (55). The parameters of the CDBm are $t = 0.005$, $t_{rem} = 100$. The maximum number of iterations of the CDBm is set to 20. The ICP is then performed with $t = 0.0002$, $t_{rem} = 100$. The deformation analysis is shown in figure (56).

It can be seen that the error of the correct mirror is really tiny compared to the other errors. For some of the mirrors, the error did not converge as the maximum number of iteration had been reached. However, this had no incidence on the final result as the correct error converges quickly because the CDB matching is already close to the minimum. In fact, the maximum number of iterations of the multiple ICP might be reduced to save time as the incorrect mirrors tend to converge slower. This choice is discussed in section 6.

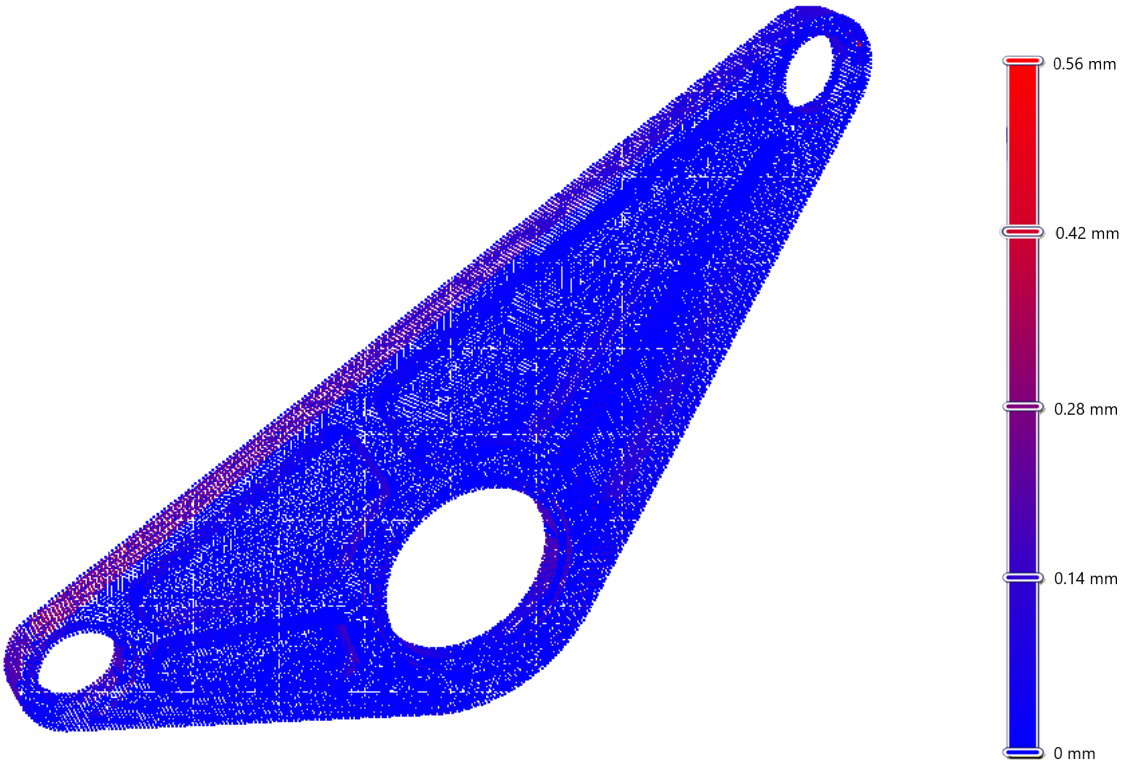

Figure 56: Deformation analysis of the asymmetrical mechanical modem when (1) is deformed

In the second test, a slight deformation is applied on a large region of the model. In that case, many points are deformed, as approximately 1/3 of the points are located on the deformed region. In that case, the algorithm has difficulties determining which regions are deformed and which regions are correct. The deformation analysis is shown in figure (57).

It can be seen that the minimization of the error leads to incorrect results. As the deformed sides of the piece share an equivalent number of points compared to the rest of the piece, the algorithm can not tell which part is correct and which one is deformed. As the error used is the square error, adding a small deformation on the correct surfaces leads to a smaller global error, rather than matching the non deformed faces perfectly. Another type of norm discussed in section 7 might lead to better results when large portion of the pieces are deformed. Another way to

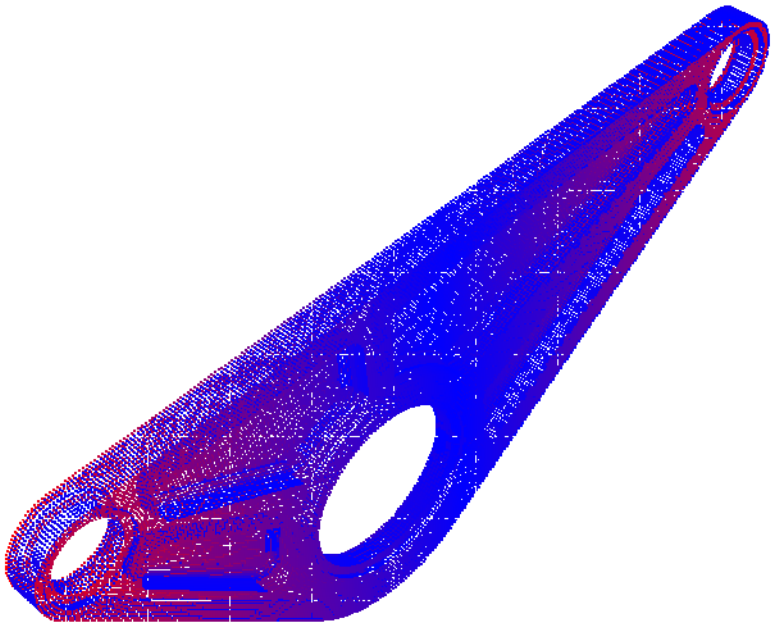

Figure 57: Deformation analysis of the asymmetrical mechanical modem when (2) is deformed

provide a better matching would be to add more information to the point clouds. An example would be to tell the algorithm which faces are correct and which one have deformations. However, the user might not know this information depending on the manufacturing of the piece.

5.5 Spherical model

The last test case of this study would be to consider an object similar to a sphere. In fact, in the case of the sphere, it is impossible to determine unique principal axes as all directions are equivalent. The matching of perfect spheres would only be done by the matching of their centroid using the translation matrix. The rotation matrix would only be determine by the noise and would thus be irrelevant. In the case of a perfect sphere, this does not pose any problem. However, if small asymmetries are present, the algorithm might have difficulties locating it. In this case, a sphere with a radius of 100 mm is considered. A small trapezoidal hole is done at the top of the sphere. This hole ensure the rotation matrix to perform the registration is unique. The depth of the hole starts at 20 mm and is then reduced until the algorithm can not find the correct rotation. In order to compare the results, the rotation matrix needed to match the point clouds can be determined during the generations of the data set. This matrix is given by equation (23). As the difficulty of this case relies in the rotations, the translation matrices are not displayed.

$$\mathbf{R} = \begin{bmatrix} 0.98 & -0.2 & 0.1 \\ 0.21 & 0.97 & -0.15 \\ -0.07 & 0.17 & 0.98 \end{bmatrix}. \tag{23}$$

The results obtained when a drill of 20 mm is done in the sphere is shown in figure (58). In this model, the sphere is approximately made of 20000 points and the drill is made of 50 points. No points are rejected, meaning the matching should be correct. The rotation matrix applied to recover this case is determined as it is difficult to check the result of this case visually. This matrix is the same as the expected matrix shown in equation (24). In this case, the asymmetries were sufficient to recover the correct registration of the spherical model.

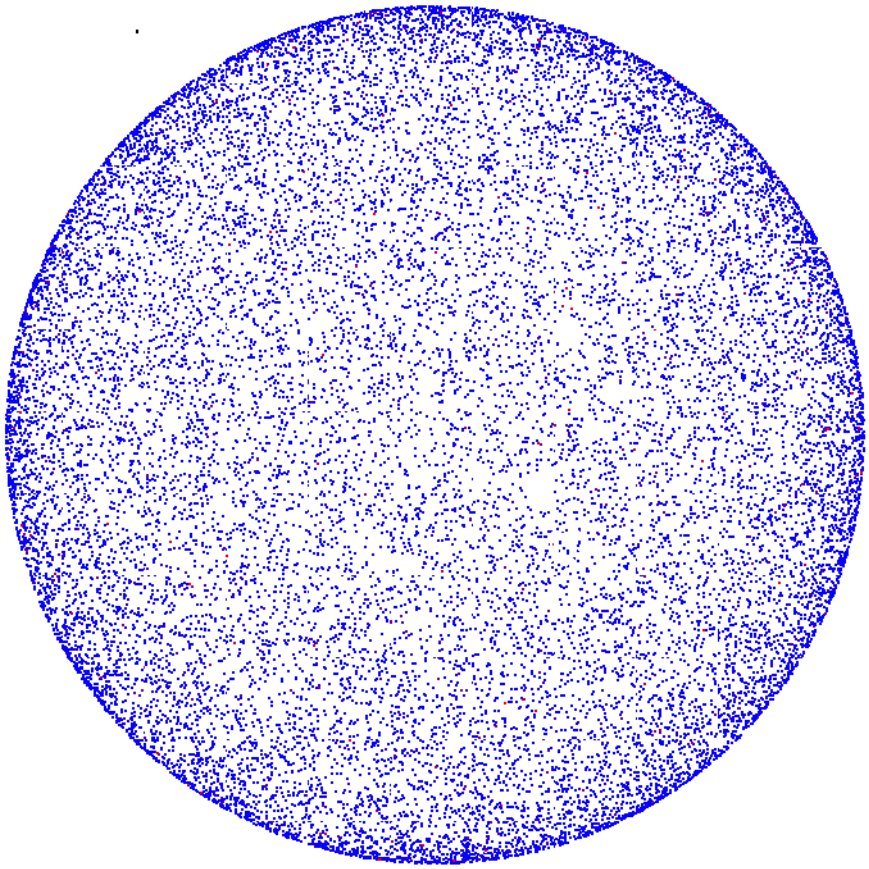

Figure 58: Deformation analysis of the sphere with a 20mm drill of 50 points

The drill is now reduced to 10mm. The sphere is still made of approximately 20000 points and the drill is made of 20 points. The results obtained are shown in figure (59). In that case, the points of the drill are rejected. Thus, the algorithm sees the model as a perfect sphere. Thus, it is unable to converge to the correct minimum. The rotation matrix obtained from the registration is shown in equation (24). As expected, the asymmetries were not sufficient to recover the correct registration of the spherical model. This means the model is too close to be symmetrical for the algorithm.

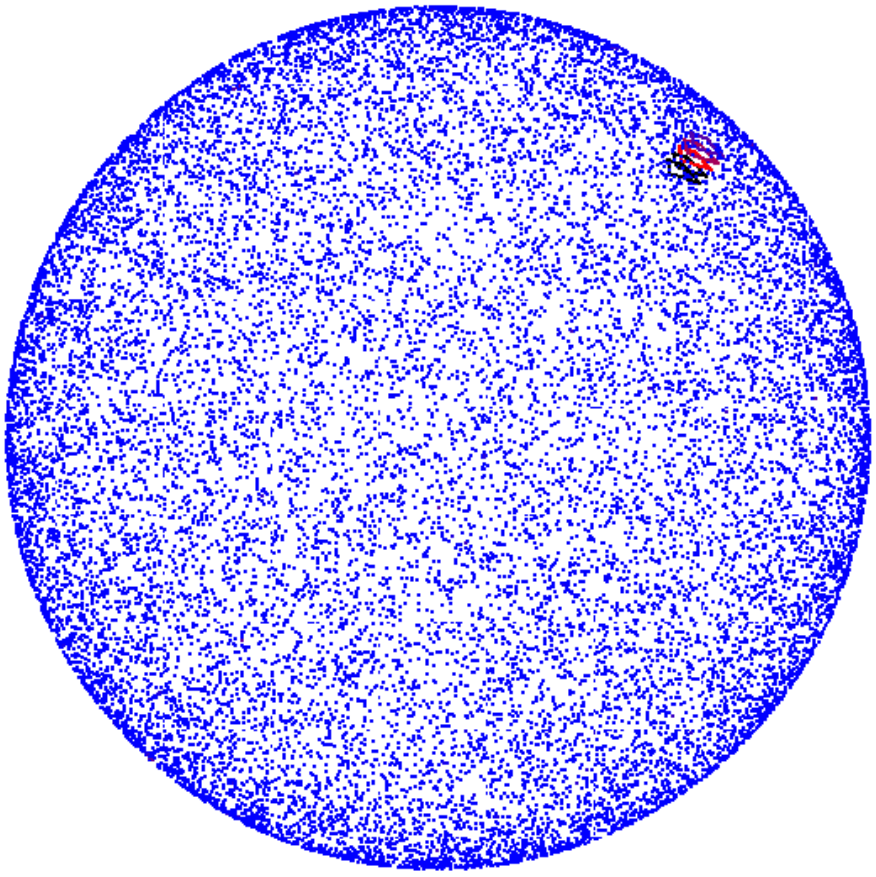

Figure 59: Deformation analysis of the sphere with a 10mm drill of 20 points

$$\mathbf{R} = \begin{bmatrix} -0.28 & 0.92 & 0.28 \\ 0.87 & 0.37 & -0.33 \\ 0.4 & -0.15 & 0.9 \end{bmatrix}. \tag{24}$$

A last test was applied to the sphere. In this test, the drill was still 10 mm and the sphere was still approximately made of 20000 points but the drill was made of 2500 points. In that case, the results are shown in figure (60) and the rotation matrix used for the registration is given in equation (25). This example is an extreme case to illustrate that the size of the asymmetry is important, but also its number of points.

$$\mathbf{R} = \begin{bmatrix} 0.99 & -0.2 & 0.07 \\ 0.18 & 0.98 & -0.17 \\ 0.062 & 0.17 & 0.98 \end{bmatrix}. \tag{25}$$

6 Determination of the parameters

In this section, the default value of the parameters kept in the algorithm are discussed. A different parameter is also discussed to remove the non corresponding point as t_{rem} is a parameter that depends too much on the workpiece. Thus, it is difficult to have any good starting value.

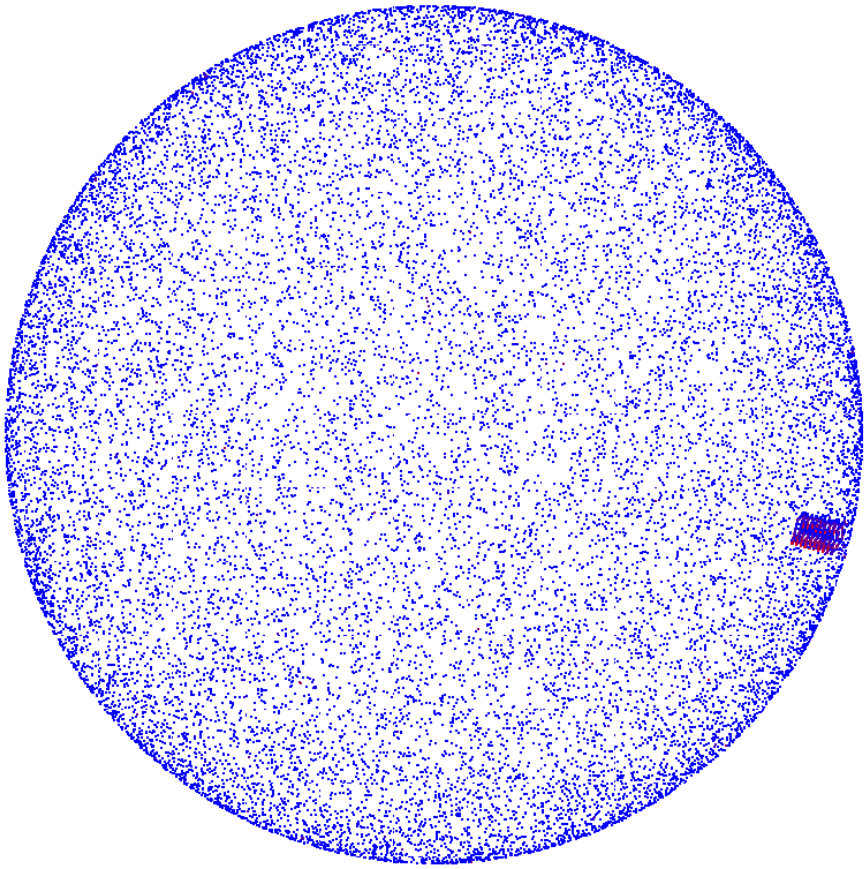

Figure 60: Deformation analysis of the sphere with a 10mm drill of 2500 points

The first parameters discussed are the number of iterations and t. Those parameters are link, as the smaller t is, the more iterations are needed to converge. Those 2 parameters must be used in both the coarse registration and the fine registration as they both use ICP. When the CDBm is performed, the precision of the convergence is not that important as it is only a coarse registration. Moreover, as the convergence has to be applied on 8 mirrors, choosing a small value leads to a high computational cost. Thus, the value of t is fixed at 0.005 for the CDBm. The number of iterations is set to 10. Most of the time, the correct mirror will converge with this precision in only few iterations. As the error is decreasing with the number of iteration until a certain limit, if the correct mirror converged, it always gives the smallest value. As an example, it required 2 iterations for the correct mirror to converge in the case of the asymmetrical mechanical piece, whereas some of the incorrect mirrors took more than 20 iterations. However, workpiece with many non corresponding points such as empreinte plâtre 1 coupée and empreinte plâtre 2 might take longer to converge. However, those types of models are not the main priority of this work as it mostly correspond to a poor CT image quality. Thus, the user might change it if it is necessary but the default value is still $t = 0.005$ and the maximum number of iteration will be 10 in the case of the coarse registration. In the case of the fine registration, it is more interesting to converge to a smaller value of t as the results of the registration correspond to the final output of the algorithm. The more precise the fine registration is, the better the results are. In that case, the threshold is set to 2e(-4). The maximum number of iteration is set to 25. This value of the threshold is

considered sufficient enough to give proper results. As the point cloud is already pre-registered, the number of iterations should not exceed this value even if t is very small. If the algorithm does not converge in 25 iterations, it is most likely due to oscillations of the error. In that case, the piece can be considered converged, even if the algorithm were to do more iterations, it would never reach the threshold. The user might change these values if necessary but the default values are t = 2e(-4)with a maximum number of iterations set to 25 in the case of the fine registration.

A third parameter discussed concerns the way the points are considered to be non corresponding. As the actual method is based on the mean error, the more non corresponding points there are, the smaller the parameter t_{rem} should be. It is due to the fact that the non corresponding points take part in the estimation of the mean error. In the case of the coarse registration, as the iterations are only performed to determine the correct mirror, the value of the parameter is set to 100. The value of t_{rem} of the coarse registration should only be changed if the algorithm did not find the correct mirror. This kind of problems occurred only in the case of empreinte plâtre 1 coupée, but is not the main issue of this work. The procedure to determine this parameter in the fine registration would be to start with a large value, and then decrease the value until the results are satisfying. However, this kind of procedure relies heavily on human interactions. Thus, another way to remove the points for the fine registration should be considered. Another option would be to use the tolerances of the workpiece as the reference. In fact, the user must always know the tolerances of the workpiece when analysing its quality. A good value based on a multiple t_{tol} of the maximum tolerances in the workpiece could thus be determined. Then, if the square root of the local error at a given point is above $t_{tol} * Tol_{max}$, the point is rejected from the determination of the translation and of the rotation matrices. This method is tested on empreinte plâtre 1. This is obviously the critical case as there are many non corresponding regions caused by the fixations used for the x-ray tomography. The product $t_{tol} * Tol_{max}$ used is 1. The deformation analysis is shown in figure (61).

If only this single test is taken into consideration, it seems that using the tolerances might be useful as a rejection criteria for the fine registration. The results are similar to the results obtained when the other rejection criteria was used. However, further tests should be performed on industrial sample in order to determine a good practice value for t_{tol} as only the product here is tested because there are no information about the tolerances in this test case. The threshold was only picked up manually until the value provides the expected results.

7 Future improvements

The last section of this thesis is about the possible improvements of the algorithm. At the moment, the algorithm gives a correct registration only if the deformed regions are local. Moreover, the estimation of the error is only good on surfaces with low curvature or on surfaces where the density of points is high enough in comparison to the curvature. However, a lot of numerical errors emerge when the curvature is high, such as on drill surface. The first improvement discussed is a modification of the error norm. In fact, using a square error is fine when the deformation is small, but when the deformation becomes too large, a linear function might be more relevant. Such a method is discussed in [22]. The function used to define the error is a Huber loss function (H). This function is written in equation (26), where k is an empirical defined threshold and d is the distance measured.

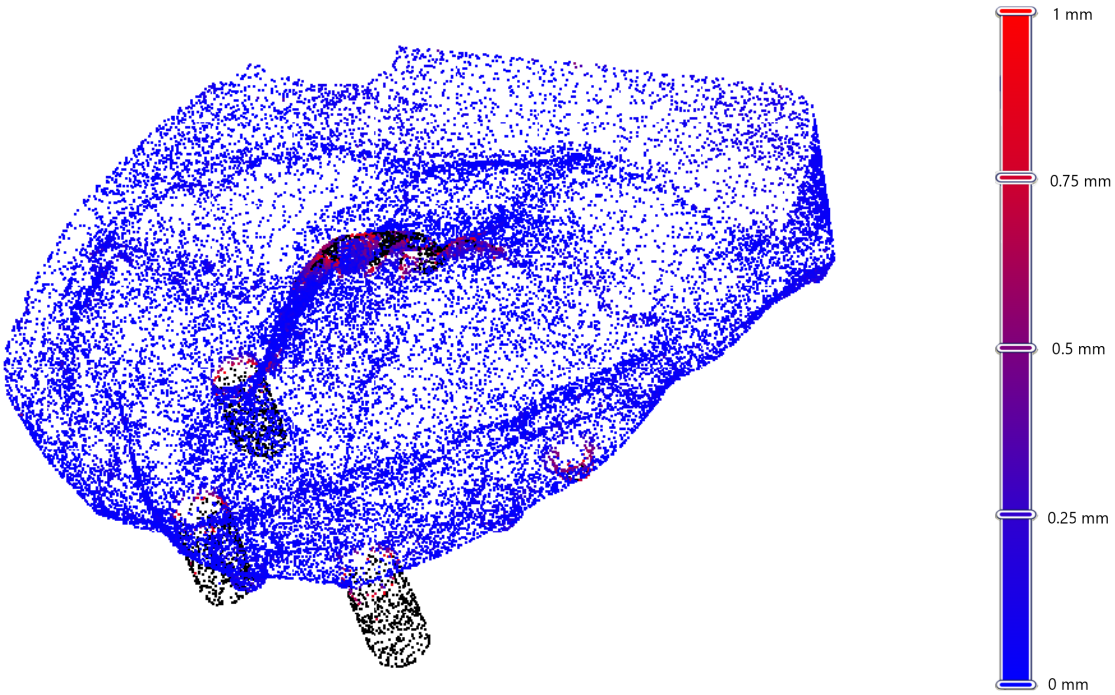

Figure 61: Deformation analysis of empreinte plâtre 1 using tolerances as the rejection criteria

$$H = \begin{cases} d^2/2 & if|n| \leq k, \\ k|n| - n^2/2 & if|n| > k. \end{cases} \tag{26}$$

The translation and rotation matrices are determined by minimizing the Huber function instead of the square error. This minimization is done through iterations using the Levenberg-Marquardt algorithm (LMA) [23].

The other major improvement would be to modify the point to plane projection used to find corresponding point in the ICP. Such an improvement is needed to get proper result out of this algorithm, in order to determine the defects of a workpiece after manufacturing. This could be done by building a local surface of a higher degree instead of a plane. Another method would be to find the plane to plane distance. This methodology was proposed by Segal and al [24]. This method is a generalization of the point to plane distance used previously. The study shows better accuracy of the matching but did not perform a deformation analysis as the aim of their work was different. Thus, this method gives more accurate results overall but might not be relevant to determine local deformations and defects.

Another missing feature of the algorithm is the detection of the poorly approximated points. As noise is present in CT images, some singular points might change the deformation scale and lead to an unwanted discard of a correct workpiece. A simple methodology would be to check the neighbors and averaging the error in order to smoothen the noise during the deformation analysis. This filter might be applied independently of the registration, after the later is performed.

8 Conclusion

Several algorithms used to perform the 3D registration of parts with respect to their models were tested in this work. The error resulting from the registration was then used to detect defects and deformations in the parts. Applying the CDB+improved ICP algorithm from [19], some limitations concerning the precision and the robustness of this algorithm could be highlighted through real scans. This study suggests some modifications to improve the robustness and the precision of the CDB+improved ICP approach. The first improvement concern the rejection of non corresponding points between a part and its model. These non corresponding points added numerical errors into the registration and were mainly caused by CT scan requirement and noise. This improvement led to a more precise registration of the point clouds. However, the choice of the rejection criteria needs to be further researched. The criteria can only give precise results if its parameter t_{rem} is chosen correctly. As of now, this selection is primarily based on human decision. The second improvement concerns the correct selection of the 180° mirrors observed during the CDB registration. The CDB algorithm was not robust when many variations between a part and its model were observed. Three different algorithms were tested to improve the robustness of the CDB. The results of 2 of them were not conclusive when noise was present in the data set. The third one, the CDBm, gave more robustness to the algorithm. The drawback of the CDBm is that its computational cost is high. The choice between the CDB and the CDBm algorithms thus depends on the requirements. In the end, only one of the sets tested in this work required the use of the CDBm. The improvement of the ICP proposed in [21] should also be taken into consideration in order to improve the robustness of the coarse registration but no comparison can be made regarding its computational cost as it was not tested in this work. Another drawback of the CDB+improved ICP approach concerns the difficulty to detect the correct transformation when the model is symmetrical. In such a case, the algorithm is unable to differentiate some of the rotations because they are equivalent. This issue may lead to an incorrect localization of the deformed regions. The CDBm was thus applied on the model as a "source". In such a case, the principal axes of the part are not translated nor rotated during the CDBm. The ICP was still applied with the model as a "target". As the ICP only applies slight translations and rotations to the data set, the principal axes of the part are easily recoverable, i.e. they are the closest to the original principal axes provided to the algorithm. Another approach would be to give the rotation and translation matrices as an output. Thus, the user may easily recover the original principal axes. The current algorithm still has many limitations, the most restrictive one being the point to plane distance used to describe the error. The planar approximation makes it nearly impossible to correctly detect the defects and deformations on high curvature regions without drastically increasing the computational cost. Using higher order surfaces to approximate the geometry of the model may lead to more accurate results in those regions. However, the computational cost required has yet to be determined.

All in all, this work provides a robust algorithm that can perform a rather precise registration of point clouds and suggests some ideas to improve the current state of this algorithm.

References

[1] Y. Sijie, Z. Yunfei, P. Fangyu, et al., Research on the localisation of the workpieces with large sculptured surfaces in NC machining, *International Journal of Advanced Manufacturing Technology* **23**(5-6) (2004), 429–435.

[2] D. Kim, J. Park and K.H. Ko, Development of an AR based method for augmentation of 3D CAD data onto a real ship block image, *Computer-Aided Design*. **98** (2018), 1–11.

[3] F. Liu, C. Quan, W. Wu, et al., Synchronous scanning mode of industrial computed tomography for multiple objects test, *Journal of X-ray Science and Technology*. **25**(5) (2017), 765-775.

[4] A. Jaiswal, M.A. Williams, A. Bhalerao, et al., Markov random field segmentation for industrial computed tomography with metal artefacts, *Journal of X-ray Science and Technology*. **26**(4) (2018), 573-591.

[5] H. Chen, H. Wang, P. Lv, et al., Quantitative Evaluation of Tissue Surface Adaption of CAD-Designed and 3D Printed Wax Pattern of Maxillary Complete Denture, *Biomed Research International* 2015 (453968).

[6] L. Tong and X. Ying, 3D Point Cloud Initial Registration Using Surface Curvature and SURF Matching, *3D Research*. **9**(2018), 41.

[7] F. Bosche, Plane-based registration of construction laser scans with 3D/4D building models, *Advanced Engineering Informatics*. **26**(1) (2012), 90-102.

[8] P. Pinter, S. Dietrich, B. Bertram, et al., Comparison and error estimation of 3D fibre orientation analysis of computed tomography image data for fibre reinforced composites, *NDT & E International* **95**(2018), 26-35.

[9] F. Rao, W.L. Li, Z.P. Yin, et al., Non-rigid point cloud registration based lung motion estimation using tangent-plane distance, *PLOS ONE*. **13**(9) 2018.

[10] F. Bosche, Automated recognition of 3D CAD model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction, *Advanced Engineering Informatics* **24**(1) (2010), 107-118.

[11] C. Kim, H. Son and C. Kim, Fully automated registration of 3D data to a 3D CAD model for project progress monitoring, *Automation in Construction* **35**(4) (2013), 587-594.

[12] F. Yang, M. Ding, X. Zhang, et al., Non-rigid multi-modal medical image registration by combining L-BFGS-B with cat swarm optimization, *Information Sciences* **316**(2015), 440-456.

[13] M. Sdika, A fast nonrigid image registration with constraints on the Jacobian using large scale constrained optimization, *IEEE Transactions on Medical Imaging* **27**(2) (2008), 271-281.

[14] L. Gottesfeld Brown, A survey of image registration techniques, *ACM Comput Surv (CSUR)* **24**(4) (1992), 325-376.

[15] S. Rusinkiewicz and M. Levoy, Efficient variants of the ICP algorithm, *Los Alamitos: IEEE Computer SOC* (2001), 145-152.

[16] D. Akca, Co-registration of Surfaces by 3D Least Squares Matching, *Photogrammetric Engineering and Remote Sensing* **76**(3) (2010), 307-318.

[17] P.W. Theiler, J.D. Wegner and K. Schindler, Keypoint-based 4-Points Congruent Sets - Automated marker-less registration of laser scans, *, ISPRS Journal of Photogrammetric Engineering and Remote Sensing* **96**(2014), 149-163.

[18] H. Chen, H. He, J. Mo, et al., 3D registration based perception in augmented reality environment, *Cogent Engineering* **3**(2016).

[19] G. Yao, Y. Zou, J. Wang, H. Yu and T. Chen, Fully automated registration of 3D CT data to CAD model for surface deviation measurement, *Journal of X-Ray Science and Technology* (2019).

[20] Y. He, B. Liang, J. Yang, S. Li and J. He, An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features, *Sensors, MDPI* (2017).

[21] J. Lu, Z. Wang, B. Hua, K. Chen, Automatic point cloud registration algorithm based on the feature histogram of local surface, *PLOS ONE* **15**(9): e0238802(2020).

[22] B. Bellekens, V. Spruyt, R. Berkvens, M. Weyn, A Survey of Rigid 3D Pointcloud Registration Algorithms, *The Fourth international Conference on Ambient Computing, Application, Services and Technologies* (2014).

[23] S. Fantoni, U. Castellani, A. Fusiello, Accurate and Automatic Alignment of Range Surfaces, & Transmission. *in 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization.* IEEE, (2012), 73-80.

[24] A. V. Segal, D. Haehnel, S. Thrun, Generalized-ICP, *in Proceedings of Robotics: Science and Systems,* Seattle, (2009), 8.

[25] http://www.cgeo.ulg.ac.be/