A coarse to fine 3D registration method based on robust fuzzy clustering

Jean-Philippe Tarel

INRIA

Domaine de Voluceau, Rocquencourt,

B.P. 105, 78153 Le Chesnay Cedex, France.

E-mail: Jean-Philippe.Tarel@inria.fr

Fax: (33) 1 39 63 57 71

Nozha Boujemaa

LI / Ecole d'Ingénieurs en Informatique pour l'Industrie

64 Avenue Jean Portalis-Technopole boite 4,

37913 Tours Cedex 9

E-mail: Nozha.Boujemaa@inria.fr

January 26, 1998

Abstract

An important problem in computer vision is to determine how features extracted from images are connected to an existing model. In this paper, we focus on solving the *registration* problem, i.e. obtaining rigid transformation parameters between several 3D data sets, whether partial or exhaustive. The difficulty of this problem is to obtain a method which is robust with respect to outliers and at the same time accurate. We present a general method performing robust 3D localization and fitting based on a fuzzy clustering method. The fuzzy set approach is known for its practical efficiency in uncertain environments. To illustrate the advantages of this approach on the registration problem, we show results on synthetic and real 3D data.

List of symbols

α alpha.

β beta.

$\mathbb{R}$ special double R, set of real number.

$\sum$ mathematical sum, sigma.

σ standard deviation of a Gaussian noise.

$\forall$ mathematical for all.

$\surd$ mathematical square root.

(R, T) rigid transformation: rotation R and translation T.

1 Introduction

Model-based interpretation of the output of 3D reconstruction algorithms has received growing attention in recent years. Specifically, pose estimation and matching of a 3D data set relative to a reference 3D geometric model, is of crucial importance in a large range of applications. Recent examples can be found in topics as different as 3D medical imaging [1], aerial site observation [9], range images for cartographic mapping [12] and augmented reality [11].

There exists a large literature [7, 4, 10, 17, 18, 2] dealing with rigid 3D-3D registration, also known as the 3D-to-3D alignment problem, and several extensions to related problems such as 3D-2D registration [14]. For 3D-3D rigid registration tasks in an industrial environment, an algorithm based on geometrical matching between primitive surfaces (plane and quadric) was first proposed in [7]. Then, surface patches were introduced to model 3D objects as a collection of planar, cylindrical and spherical patches and used as a basis to estimate the 3D pose with a probabilistic approach [4]. It was proved that this algorithm performs locally optimal estimation. To find a global optimum, systems where hypothesis are generated step by step was proposed, such as the 3DPO system where search-trees are used [10]. An alternative solution to search-trees is to use a Hough like transform generalized to rigid transformations [17], or in the same spirit a hash table can be used [18]. To deal with smooth objects defined as surfaces, the "Iterative Closest Point" (ICP) approach was proposed [2]. However this approach only performs a refinement of the registration without outliers. In sum, numerous extensions were proposed in the recent years to handle, for example, different kinds of representations such as curves [20], or to take into account outliers [13, 21].

We distinguish three major approaches for 3D registration:

- Combinatorial approaches where the matching is pre-processed before the pose estimation computation, such as generalized Hough transform [17] and hash table techniques [18].
- Construct approaches where the matching is performed step by step during the pose estimation process, such as search-tree methods [7, 10] and [4, 1, 8],
- Implicit approaches where the matching is revised iteratively during the pose estimation such as ICP

methods [2, 20, 13] and Monte Carlo random search [21].

Implicit methods are, in general, sensitive to initialization and may not converge to the optimum solution, especially when the initial position is not close enough to this optimum. An exception to this statement is the algorithm proposed in [21], which uses a Monte Carlo random search and is less sensitive to initialization. On the contrary, combinatorial and constructive approaches do not require good initialization, but are often less robust to outliers and noise than implicit ones. For example, hash table and generalized Hough transform techniques are not very robust because the result is sensitive to noisy data as well as to the choice of the resolution in the accumulation space. In the particular case of the registration problem the accumulation space is the space of Euclidean transformations which is of dimension 6.

Since our application is augmented reality, we have to estimate the 3D pose of several given object models in a 3D reconstruction obtained from stereopairs. After registration, synthetic objects can be added with a realistic interaction into the 3D scene [11]. The representation of the given CAD model is a collection of patches defined by its vertices. The number of patches can be large (until 1000 patches). The reconstruction algorithm we use also provides a planar patch representation. We do not address here the problem of the conversion of a 3D data set into patches. Generally, the number of reconstructed patches is not large (no more than 100 patches), but the shape of the obtained patches is very complex and noisy. Since reconstructions are very noisy and occluded in comparison to geometric models, our objective is to develop first a robust algorithm with no real time requirements. In this paper we propose a method for the registration of two 3D data sets which takes advantage of both implicit and combinatorial approaches to achieve robustness. We focus on the Hough like transform, because we think that this technique is more powerful than the others. Its main potential advantage is to perform registration of several instances of the same object. To do the same task, other techniques have to first converge to one of the possible rigid transformations, eliminate it, return to find another instance, and so on.

In the approach we deal, registration becomes at the end a minimization problem. We focus on the three requirements of robustness for minimization algorithms:

- robustness with respect to noisy data (noise is given by a fixed centered Gaussian for instance),

- robustness with respect to outliers (often only a part of the input data set is significant),
- robustness with respect to initialization.

To achieve these three types of robustness, we have based our approach on fuzzy sets known for their efficiency in uncertain environments. Our method proceeds in three steps as described in section 2 (see Figure 1). First, we look for the set of all potential rigid transformations between both 3D data sets with a feature-to-feature correspondence algorithm. In the transformation space, rigid transformations which closely register a large part of the two 3D data sets are out numbered in an outliers set of non-significant rigid transformations. The second step provides the more relevant Euclidean transformations in a fuzzy way. A confidence value, based on feature-to-feature invariants, is introduced in order to better cluster transformations among outliers. Consequently, the first and second steps comprise a global registration. Each hypothesis of rigid transformation provided by step 2 is applied to the original data set. Finally, in the third step, local and fine fitting is achieved between the 3D data sets. In section 3, we discuss the advantages of the proposed method and show results on synthetic and real 3D data. In this section, we estimate the similarity between two 3D data sets with a tool we developed and asses the performance of our registration method.

2 General scheme

2.1 3D feature-to-feature matching

In this first step, a rough 3D matching is obtained by a combinatorial approach. 3D features are selected in each data set. Of course, we assume that a technique to compute rigid transformations between any two subsets of the same type is available. In this paper, planar patches were chosen as features (Figure 2). Other geometric features can be used, such as 3D bases of 4 points [18], pairs of segments, or local Frénet reference systems on curves [8].

Each hypothesized match provides N_{displ} Euclidean transformations which superimpose the two selected

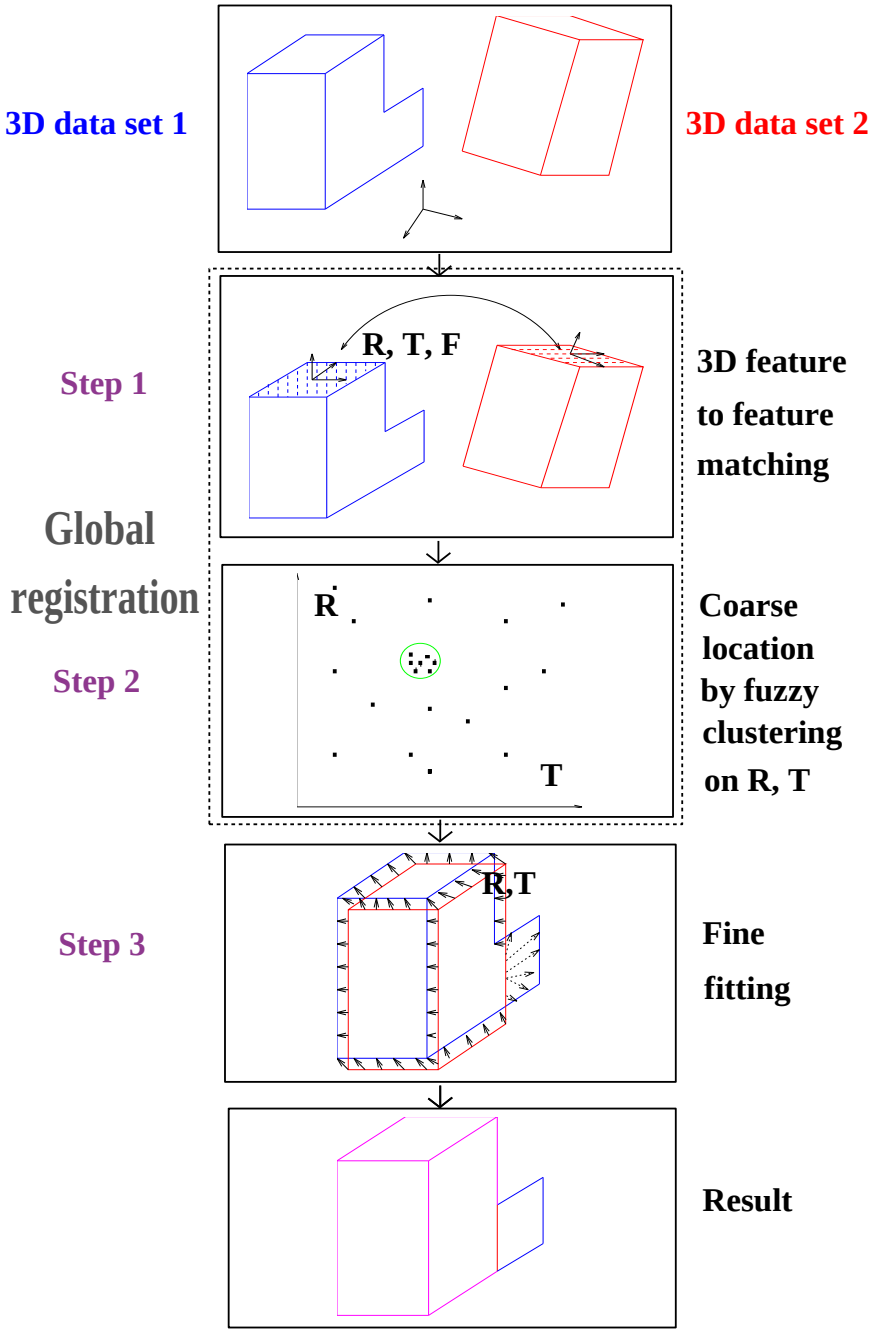


Figure 1: *General scheme of the fuzzy 3D registration approach.*

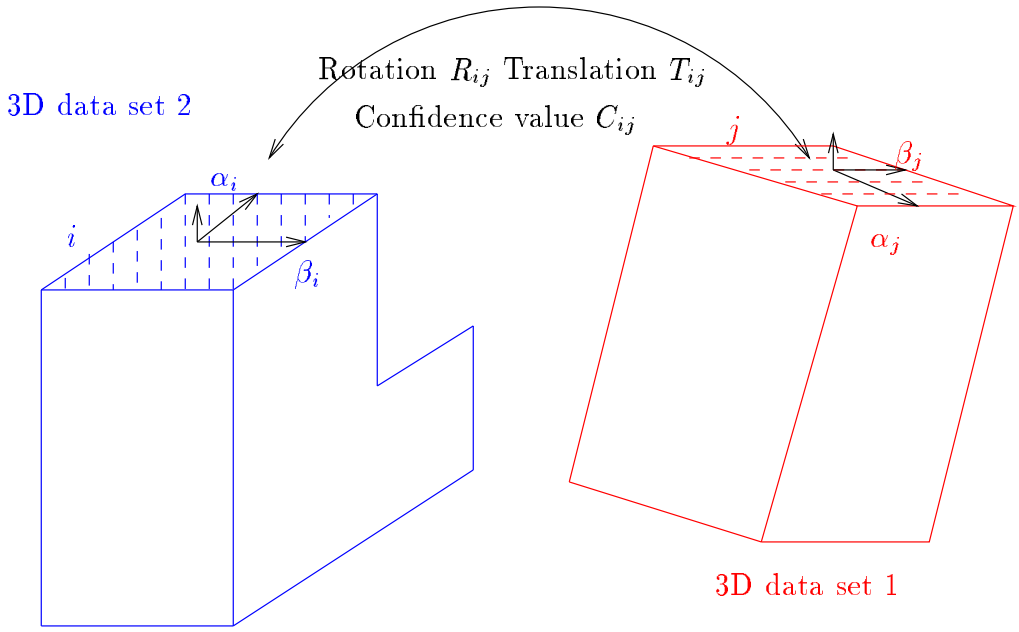


Figure 2: *3D planar patch to planar patch matching. For two rectangular patches, 4 possible rigid transformations superimpose the two shapes.*

features. In the general case, the number N_{displ} depends of the shape and nature of the chosen feature. For example, in our particular application, a planar patch can be partially represented by its equivalent ellipse in 3D space i.e, by its center and its inertia matrix. Thus, with a pair of elliptical patches, 4 solutions are proper ($N_{displ} = 4$). When the patch is equivalent to a circle, the rotation computation is degenerate, and thus the set of possible rotations is randomly sampled. Consequently, for each pair of patches (i, j):

- we obtain the rotation matrix R_{ij} by the diagonalization of the inertia matrices,
- we compute the translation vector T_{ij} from the center of gravity,
- and we estimate the confidence value C_{ij} of the match.

Effectively, one can estimate the similarity of the two matched features from feature characteristics that are invariant under rigid transformation. For example, for equivalent ellipses, we have used the first and second eigenvalues α and β of the inertia matrix as invariants. For each pair (i, j), we compute a

match confidence value C_{ij} given by:

$$C_{ij} = \frac{1}{\max(1, \frac{|\alpha_i - \alpha_j|}{I})} \; \frac{1}{\max(1, \frac{|\beta_i - \beta_j|}{I})} \tag{1}$$

where I is the variance of the error observed on the localization of the 3D data. I can be interpreted as the resolution of the localization. This criterion gives a confidence value between 0 and 1 for two planar patches (i, j). The better the similarity of patches i and j, the more C_{ij} is near 1.0. If the difference between the eigenvalues is under the resolution on the 3D data, the confidence value is always one (see Figure 3).

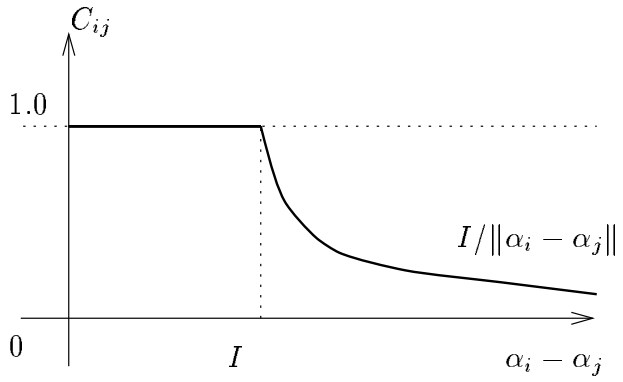


Figure 3: *Variation, given by equation (1), of the confidence value C_{ij} with the difference between the eigenvalues $\alpha_i - \alpha_j$. I is the mean error on the 3D data.*

The parameter I can be estimated and fixed off-line for any given computer vision algorithm (see section 3.6 for an example where we estimate I with our stereo reconstruction algorithm). Moreover, additional feature information, such as color, can be integrated into the scheme by designing an adequate criterion (see section 3.5).

Rigid transformations produced by bad matches generate outliers which systematically out number correct transformations. Therefore, an important characteristic of the data is the ratio of the number of right transformations over the total number of transformations. In the case where both 3D data sets are composed of $N_{patches}$ similar patches, the SNR ratio is $\frac{N_{patches}}{N_{displ} N_{patches}^2} = \frac{1}{N_{displ} N_{patches}}$. For instance, for a

cube, we have $N_{patches} = 6$ and then $SNR = \frac{1}{24}$. But for a box, by using the fact that sides has different sizes, we obtain $SNR = \frac{6}{4\times6\times2} = \frac{1}{8}$. Nevertheless this ratio decreases rapidly with $N_{patches}$. Consequently, it is better to choose relatively abstract 3D features with intrinsic characteristics, like planar patches, to avoid a combinatorial explosion.

2.2 Coarse 3D localization

From the first step, we obtain a set of n rigid transformations (R_{ij}, T_{ij}), each of which is assigned a confidence value C_{ij} (Eq. (1)). A 3D rigid transformation is represented by 3 rotation angles and 3 translation components, or a point x_i in $\mathbb{R}^6$. In what follows, we change the transformation representation and indexation:

$$(R_{ij}, T_{ij}) \rightarrow x_{k(i,j)} \tag{2}$$

$$C_{ij} \rightarrow C_{k(i,j)}$$

In this set $(x_k)_k$, *a local accumulation point expresses a significant hypothesis of registration* between the 3D data sets. To determine such accumulation centroids in a robust way, a fuzzy clustering method is applied to the data set $(x_k)_k$. Moreover, the clustering method can handle several transformation hypotheses at the same time.

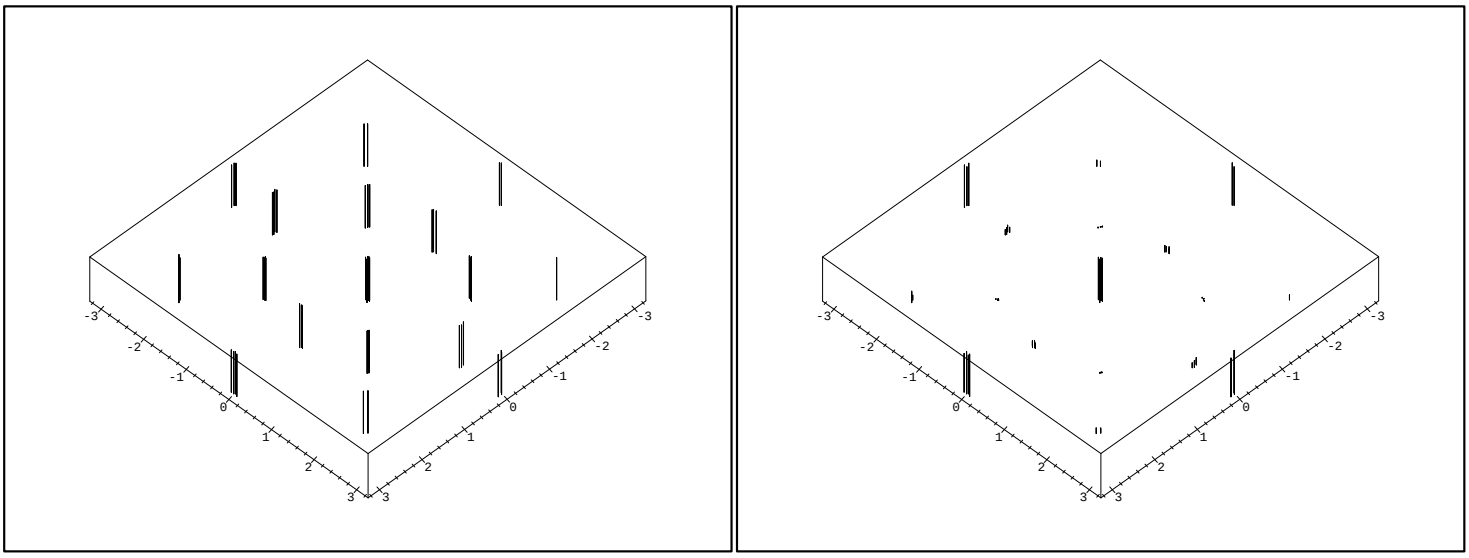
It is important to note that step 2 is very generic and can be used in numerous cases such as point, segment, curve, planar and quadratic patch-based rigid or non-rigid registration, by choosing a more adequate choice of working space.

2.2.1 Fuzzy approach

Fuzzy set theory provides mathematical tools to handle uncertainty properties of computer vision data. Clustering techniques are appropriate, in this case of study, to compute global membership degrees in a robust way. The main idea is to allow distributed membership of a given pattern to several subsets in the working space. Fuzzy C-Means clustering algorithms were first introduced and generalized by Bezdek

(FCM) [3]. Several developments and applications have been used in image segmentation (see for example: range images [15], medical imaging [5], indoor scene and texture [5]).

For 3D registration problems, we consider the set of Euclidean transformations as a working space. In this case, significant clusters are flooded by outliers so that it is essential to model outliers. Davé [6] has proposed an algorithm introducing the concept of a noise cluster. This additional cluster is designed to collect the noise data points. In particular, all the points in the data set are considered to be at the same fixed distance, the parameter δ, from the noise prototype.



(a) (b)

Figure 4: *The transformations set between a box (6 patches with sizes* 2 *mm,* 4 *mm and* 6 *mm) and another box in the same position. In (a) a 2D projection on the first and second angular components of the set is shown. In this Figure it is clear than the 5 valid accumulation points are flooded in numerous none-valid points. After adding the confidence value on the third coordinate as in (b), it becomes easier to find the 5 valid accumulation points. The accumulation points are located in* $(0,0)$, $(0,\pi)$, $(0,-\pi)$, $(\pi,0)$ *and* $(-\pi,0)$ *because of the symmetries of a box.*

Davé's method gives excellent results when the specification of the SNR ratio is not very important (ratio larger than 5% of correct points). In our problem, this ratio is very critical (Figure 4) so that the

clustering algorithm, without additional information, is not able to discriminate between correct and noisy data.

2.2.2 Proposed algorithm

To improve the efficiency of Davé's algorithm, we introduce the confidence value C_k associated with each $\mathbb{R}^6$ data point x_k (see section 2.1) which is invariant to whole rigid transformations, in addition to its membership degree u_{ik} in cluster i. The participation of a given data point to the clustering process is weighted by the associated confidence value in addition to the membership degree.

Consequently, the new objective function to minimize is:

$$J_m = \sum_{i=1}^{c} \sum_{k=1}^{n} C_k u_{ik}^m d^2(x_k, v_i) \tag{3}$$

where $d(x_k, v_i)$ is the distance between the point x_k and the cluster prototype v_i, and u_{ik} is the fuzzy membership of the point x_k to cluster i. The value of u_{ik} is always positive and its sum across all clusters equals 1. This constraint $\sum_{i=1}^{c} u_{ik} = 1$ forces the clustering method to explain the whole data set by c fuzzy sub-sets (see [3] for an extended explanation on this particular aspect).

The demonstration of the convergence of the extended algorithm can be deduced from the proof of the original algorithm (see Appendix and [6]). Here, x_k is an element of the data set of n points in $\mathbb{R}^p$ ($p = 6$ in our application) and its confidence weight is C_k. The distance $d(x_k, v_i)$ is defined as the Euclidean distance between the data point x_k and the cluster prototypes v_i for $i = 1$ to $c - 1$, and is the constant δ for the noise cluster ($i = c$).

The modified algorithm can be summarized as follows:

step 1 : Fix the number of clusters c, and fix the exponent m (usually $m = 1.5$). Compute the initial position of the cluster centers v_i (by an FCM algorithm, for example). Specify the parameter δ linked to the localization noise on both data sets.

step 2 : Generate a new partition using the expression for the fuzzy memberships:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d(x_k, v_i)}{d(x_k, v_j)} \right)^{\frac{2}{m-1}}}$$

step 3 : Calculate new cluster centers v_i using the following expression (for $i = 1$ to $c - 1$) where the confidence weight C_k is introduced :

$$v_i = \frac{\sum_{k=1}^{n} C_k u_{ik}^m x_k}{\sum_{k=1}^{n} C_k u_{ik}^m}$$

step 4 : If the partition is stable, stop; else goto step 2.

More generally, the use of the C_k confidence weights is a natural and simple extension of all kinds of fuzzy clustering methods (see Appendix) to deal with a highly noisy context. It allows us to keep the ratio of correct points up, and to manage additional knowledge on each point.

2.2.3 Gaussian sphere: improved working space

In Figure 4, we see that the accumulation points $(0, \pi)$ and $(0, -\pi)$ are in reality the same accumulation point because an angle is only known modulo 2π. In our case, every angle is defined between $-\pi$ and π. To fix this difficulty we modify the working space from a 6D linear space to the Gaussian sphere for the angular terms [19] (the first three terms of x: (x^1, x^2, x^3)), and to stay within the 3D linear space for the translation terms of transformations (the last three terms of x: (x^4, x^5, x^6)).

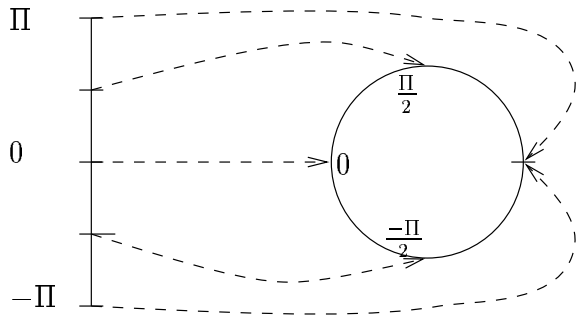


Figure 5: *For angles, the correct working space is a circle and not an interval.*

This means that, as shown in Figure 5, every angle is at the same distance of π and $-\pi$. Moreover, it is not difficult to adapt the Euclidean distance d to take care of this angular property. Consequently, we

define the distance used in the computation of fuzzy memberships u_{ik} as:

$$d^2(x,y) = d_{angular}(x^1,y^1)^2 + d_{angular}(x^2,y^2)^2 + d_{angular}(x^3,y^3)^2 + (x^4-y^4)^2 + (x^5-y^5)^2 + (x^6-y^6)^2$$

with the angular distance $d_{angular}(\theta,\phi) = |\theta-\phi|$ if $|\theta-\phi| < \pi$, and $d_{angular}(\theta,\phi) = 2\pi - |\theta-\phi|$ if not. The computation of new cluster centers v_i, is adapted in an similar way.

In practice, the introduction of the Gaussian sphere do not modify the convergence of the algorithm.

2.3 Fine fitting

The clustering method gives $c-1$ different hypotheses of rigid transformation, such that a portion of both 3D data sets superimpose. But it is often needed to improve the obtained result to optimal 3D registration by local fitting.

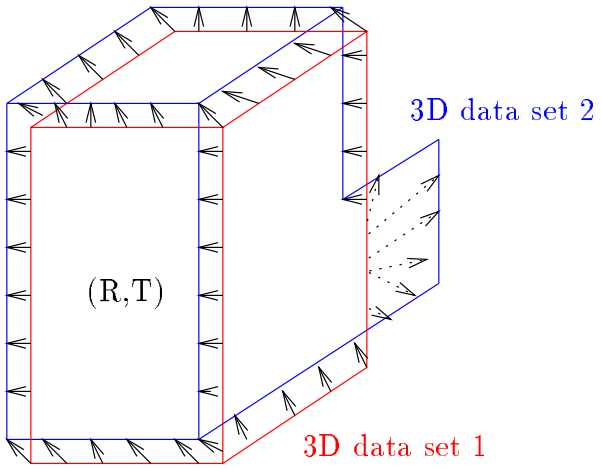


Figure 6: *Accurate registration with ICP algorithm. Matching is done implicitly by computing the closest point at each step of the algorithm and by minimizing the set of obtained displacements.*

We use a registration procedure based on implicit correspondences established between points of one data set and the closest point of the reference data set (ICP) [13]. When implicit matching is done, a better transformation (R,T) between the points in correspondence is estimated. This iterative method converges to a position where the distance between the input data and reference data is locally minimal.

ICP algorithms generally have a high computational cost, however in our approach, only a few iterations are necessary because the clustering method gives solutions close to a local minimum.

Since the two 3D data sets are incomplete, it is necessary for good accuracy to eliminate bad point-to-point correspondences. When the initial guess is close to the real solution, these bad matches are rejected automatically [13] using the distance between the two points of the match (see Figure 6).

Consequently, an accurate 3D registration is obtained for each hypothesis. This step provides accurate and robust results since the initial guess is close to the real solution. The mean distance between both data sets gives a criterion for hypothesis selection and sorting. The lower the mean distance, the better the solution. Another interesting parameter to help select the optimal registration is the relative size of the overlapping part of the 3D data sets after transformation (see section 3.6).

3 Results and discussion

The described algorithm is integrated in an indoor scene interpretation system [11] based on stereoscopic images, which performs camera calibration, region image segmentation, region matching for each pair of images and finally 3D reconstruction.

We have validated our approach on synthetic, as well as real, data. Control parameters of the complete method are not numerous and have a physical meaning. The number of clusters c is determined by an examination of the symmetry of the object model. The parameter δ is directly related to the resolution of localization, thus in practice, it is equal to I. These parameters can be estimated by an a posteriori analysis, or set with an approximated knowledge of the accuracy of the input data. We want to emphasis the fact that we overcome the difficulty of choosing a threshold, by a using fuzzy clustering approach.

3.1 Robustness to noise

First, we test the robustness of the proposed algorithm under noise on data points. And we investigate how sensitive the choice of the value of the parameter $I = \delta$ is, relative to the noise standard deviation.

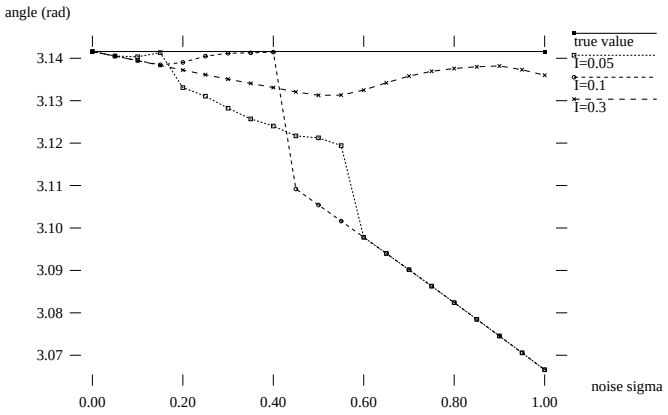


Figure 7: *Estimation of one possible rotation angle between a box of size* $(2.0, 4.0, 6.0)$ *and its copy, when Gaussian noise with a standard deviation of* σ *is added to each point of the first data set. Each curve is for a different choice of* I*. This Figure illustrates the fact that only an approximated value of the resolution* I *is needed in order to specified the magnitude of the noise.*

For several values of the standard deviation σ of a Gaussian zero mean noise, the influence of the choice of I is shown in Figure 7. Here the registration is performed between a single box and its noisy version. The size of the box is 2 mm, 4 mm and 6 mm. The estimated angle is plotted for a standard deviation σ of the noise in a range from 0 to 1 mm. This Figure shows that when the noise is lower than $3I$, the angle is always estimated with an error lower than one percent for the box. For upper values of the noise, the error increases rapidly in a non-linear way. Similar results are observed for more complex objects with a different range of values. This illustrates that with our fuzzy approach, only the range of the noise is needed, not its exact value, as in common thresholding approaches. Experiments with more complex objects show that if the value of I is too large, the obtained cluster centers can be biased, producing incorrect results. This can be explained by the fact that if radii of clusters is chosen too large, then the result is biased by points in the neighborhood. Indeed, radii of clusters is of the same order of magnitude than the value of δ.

The next experiment is applied on the 8 objects shown in Figure 8. This set of objects contains very simple objects such as the box or the L-shape, and complex ones such as the roomdesk, the face and the

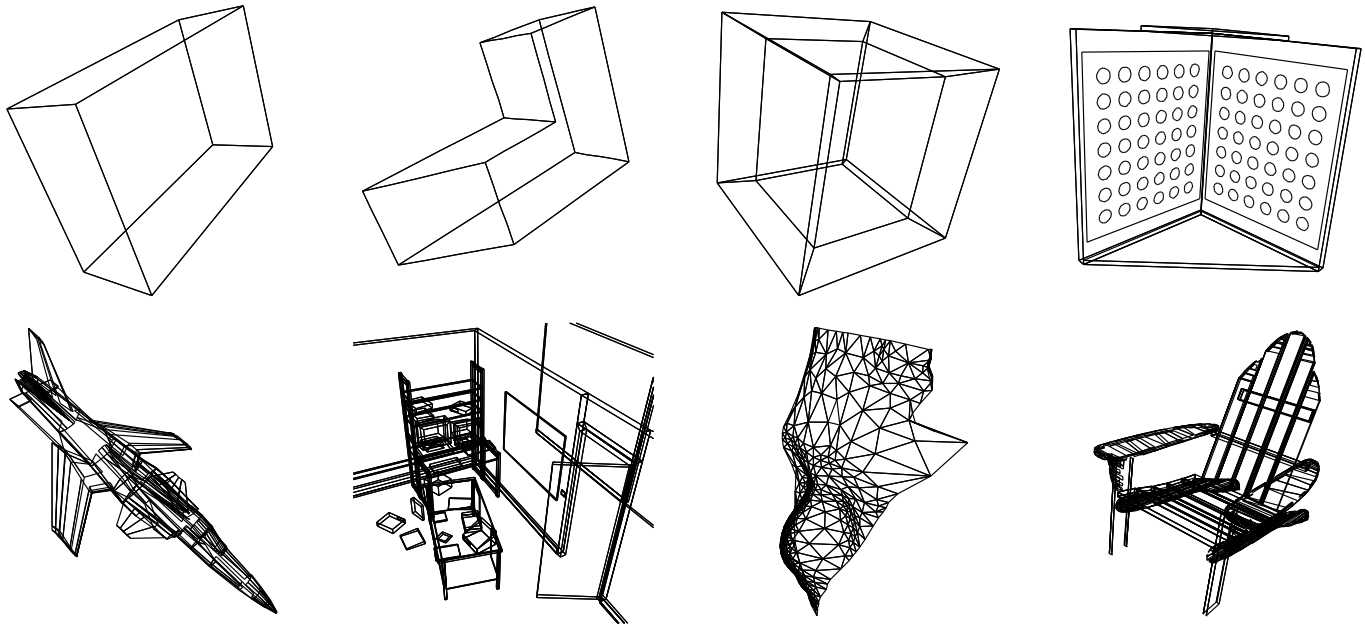


Figure 8: *The 8 objects used for experiments sorted in order of increasing number of faces: box=6, L-shape=8, arch=16, 2-plane=104, X29=270, desk=532, face=817 and chair=1000 faces.*

chair. Some objects are highly detailed such as the roomdesk and the chair, others are simple global shape such as the model of the Defense's arch in Paris (3rd one), and the face. In these data sets, there are several types of planar patches: rectangular patches for the box, the L-shape, the arch, the room desk, the X29, triangular patches for the face, circular patches for the 2-plane object, and quadrilateral patches for the chair.

The computational complexity of the more intensive step (step 2) is proportional to the product of the number of iterations, the number of points in the accumulation space, and the number of clusters c. The number of iterations do not exceed 50. The number of points in the accumulation space equals $4 * N_{patches} * N_{patches}$ in the worst case. Consequently, on data sets with much more than one thousand patches the proposed algorithm requires more than 1 hour to run it on a sparc station 10 (the code is not optimized).

Since only an approximate value of the resolution is necessary, we assume in this test that the value of the noise σ is well estimated by the user and we set I equal to σ. Then, we add Gaussian noise and

object	max. noise	min size	average size	max size	$N_{patches}$	SNR	SNR'
box	2.26	2	4	6	6	0.042	0.12
L-shape	0.81	1	1.4	3	8	0.031	0.12
arch	5.7	2.8	7.8	11	16	0.016	0.078
2-plane	5	0.1	0.33	10	104	0.0024	0.0037
X29	10	1.9	93	490	270	0.00093	0.13
desk	22	2	500	6400	532	0.00047	0.036
face	0.5	0.96	7	70	817	0.00031	0.0087
chair	3	1.3	1000	15500	1000	0.00025	0.055

Table 1: *Assuming that the parameter I is equal to the standard deviation of the noise σ, this table shows the maximum value of σ before our algorithm estimates the pose angle with an error up to* 10%. *To have an idea of how robust the algorithm is, the minimal, average and maximal sizes (mm) between two vertices in every object is also shown. The value of SNR in the case without noise is estimated. SNR is defined as in section 2.1, and SNR' is the signal/noise ratio when the confidence value is taken into account.*

apply a rigid transformation on a copy of each object. For every object, the rotation part of the pose to be estimated (up to object symmetries) is $x^1 = 0$, $x^2 = 1$ and $x^3 = 0$ rad, and the translation part $x^4 = -5$, $x^5 = 0$ and $x^6 = 0$ mm. Then, we increase the value of the noise step by step, until the error between the estimated and the real angle is at 10 percent. Obtained values of the noise for every object are shown in table 1. To know how large the value of the noise supported by our algorithm is, the smallest and biggest edge size of the patches are also shown in this table. Notice that the more detailed the object is, the more likely the noise is to be large. Most of the time, the standard deviation of the noise (before the angle has an upper error of 10%) is larger than the size of the smallest patch. The only exception is for the face. This can be explained by analyzing the histogram of patch sizes. This object has a large number of patches of approximately the same size, because it is a triangulation. Thus, when noise is added, most of patches seem to be of similar sizes. In this situation, the first step of our algorithm performs poorly, and as a consequence there are too many points in the accumulation space. This fact also explains the small SNR' value. The value of SNR is defined as in section 2.1 where the confidence weights are not used. Using the confidence weights allows us to increase the SNR to SNR'.

3.2 Robustness to occlusion

Another important issue for computer vision methods is the robustness to occlusion. We apply the same kind of tests as in the previous section, where the data set is occluded. To simulate occlusion, vertices on patches are randomly eliminated. The percentage of occlusion is given by the probability p to eliminate a vertex .

Under occlusion, the choice of the value I is not important, contrary to the findings of the previous section. Figure 9 shows under different amount of occlusion the amount of correct estimation on twenty realization of the noise. The registration is applied between the box and its occluded version. Notice that the obtained curve is very similar to the theoretical one. Since the probability of a box side not being occluded is $q = (1-p)^4$, the probability of having at least 3 non-occluded patches is $20q^3(1-q)^3 + 15q^4(1-q)^2 + 6q^5(1-q) + q^6$. This equation gives the theoretical curve shown in Figure 9. In this figure, the number

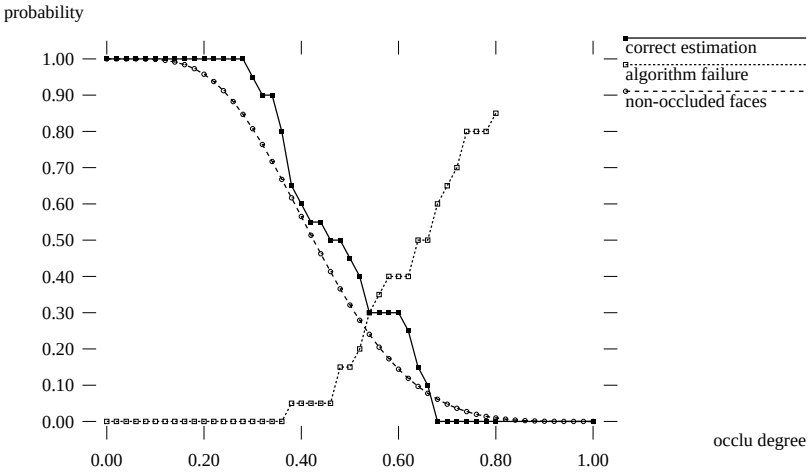


Figure 9: *Estimation of one possible rotation angle between the box and its copy, when the percentage of occlusion on vertices of the first data set is increased. Notice that the angle is always well estimated when the noise is lower than* 30%*, which is in accordance with what the theory predict.*

of failures of our algorithm is also plotted.

Table 2 summaries the results of the occlusion tests for the 8 objects shown in Figure 8. In these tests the percentage of occlusion is increased until the error on the angular estimation of the rotation is more than 10 percent. In the first row of this table, no noise is added, and the value of I equals approximately ten percent of the size of the smallest patch. In the second row, Gaussian noise with a standard deviation of 0.5 mm (about half of the size of the smallest patch) is added to occlusion. The robustness to occlusion is satisfying, and most of the time the algorithm can handle up to 50% of the object occluded. In our case, notice that the more patches the object has, the more occlusion can occur. There are two exceptions for the airplane and the face. The medium robustness to occlusion on the airplane is due to the fact that most of the patches of this object are very thin, and hence, the object is more sensitive to occlusion. The poor results on the face means that, when triangulation are used, step 1 of the algorithm must be adapted to handle triangular shapes in the manner that if does rectangles.

object	$\sigma = 0$ $I = 0.1$	$\sigma = I = 0.5$
box	57%	36%
L-shape	56%	39%
arch	40%	48%
2-plane	80%	65%
X29	36%	35%
desk	72%	32%
face	40%	10%
chair	76%	35%

Table 2: *Maximum percentage of occlusion before our algorithm estimates the pose angle with an error higher than* 10%. *In the first row, no noise is added, and in the second one, Gaussian noise with a standard deviation* σ *of* 0.5 *mm is also added.*

3.3 Dealing with symmetries

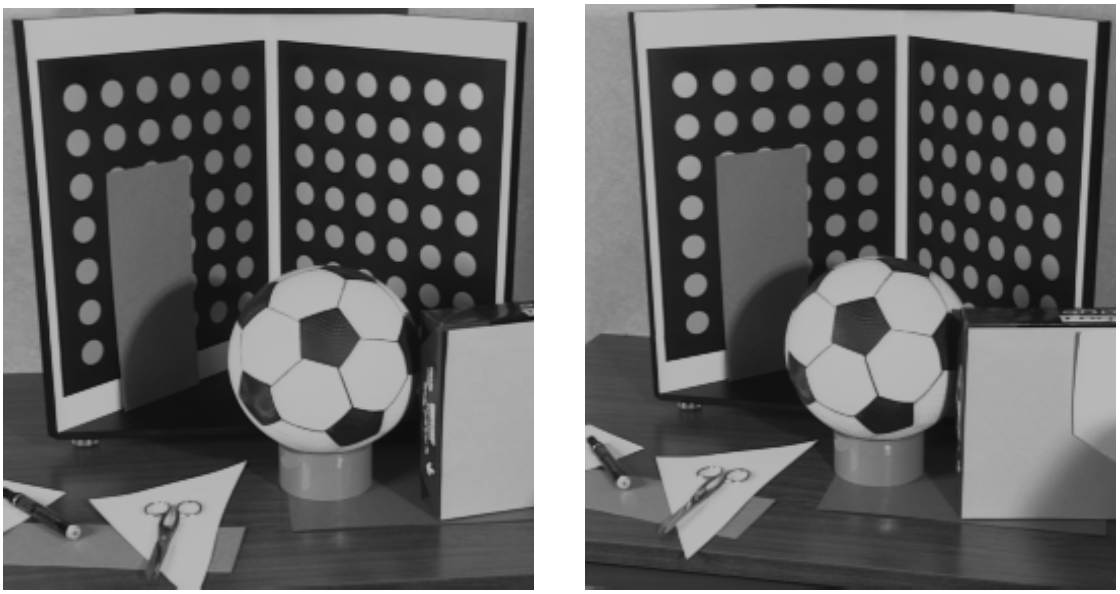
Apart from its robustness to noise on data and occlusion, an advantage of the proposed method is the possibility to manage a set of coarse registrations. In particular, the number of clusters c must be chosen according to a symmetries finding of the 3D object. For example the box shown in Figure 8 can be exactly registered by 4 rigid transformations due to its symmetries. These 4 solutions are obtained by setting $c-1$ equal to 4. If only few solutions are needed, we can set c to a lower value. We deal with the arch in exactly the same way.

If we search for the whole solution set, we have take into account not only exact symmetries, but also approximate symmetries. For instance, the second, fourth or fifth objects shown in Figure 8 have no exact symmetries, but if a few planar patches are eliminated, these objects have approximative planar symmetries. For these objects, two symmetrical positions are possible for global registration. It is possible to obtain the whole set of registrations with $c = 3$ including the noise cluster. For the other objects of Figure 8, only one solution has to be obtained. If the value of c is over-determined, we will obtain some clusters reduced to only few points. This kind of cluster can be easily discriminated from desired ones. Moreover we notice than if c is over-estimated, the good solution has more chance to be selected by the algorithm even in very noisy and occluded situations. However, due to computational cost, we have to choose the smallest value c which allows us to obtain a good solution. If c is underestimated, we notice in our experiments that the consequence is that only a subset of the interesting registrations hypothesis is obtained.

3.4 Registration with several objects

One advantage of the proposed method is its robustness, allowing us to produce registrations with several objects and hence, models. For example, on 3D data (Figure 11(a)(d)) obtained with stereovision algorithms on real images (Figure 10(a)(b)), models of a soccer ball and of a planar object are registered and accurately fit with the proposed method (Figure 11).

The soccer ball is an object with numerous geometric symmetries, so in this example a large number



(a) (b)

Figure 10: *(a)(b) Left and right images of a real scene. Views of the 3D data obtained by stereovision algorithms on these images are shown in Figure 11(a)(d).*

of clusters c is needed. Since we are interested in picking up only one solution, we can restrict c to the value 6, for example. With this value, we have a good overlap of the set of clusters on the accumulation space. A smaller value of c can produce incorrect results, since few computed clusters are reduced to a singleton. With 6 clusters corresponding to 6 registration hypotheses of the soccer ball, the cluster with the maximum number of points is always a correct possible solution for the registration. The parameters used in these experiments are $c = 7$, $m = 1.5$ and $I = 4mm$ (I is estimated as in section 3.6). There are 104 patches for the 2-plane object, 32 for the soccer ball and 83 for the reconstructed data set.

Finally, each output of clustering process are refined by fine fitting with all data information and sorted on the basis of the quality of the data fusion. The Figure 11(e) shows the obtained position where the residual distance after the refinement step is optimized.

The relevance and the precision of the method is illustrated in Figure 12, which depicts different points of view for some registered models on some stereo pairs where the left original image is projected on the 3D models (Figure 10 and Figure 13).

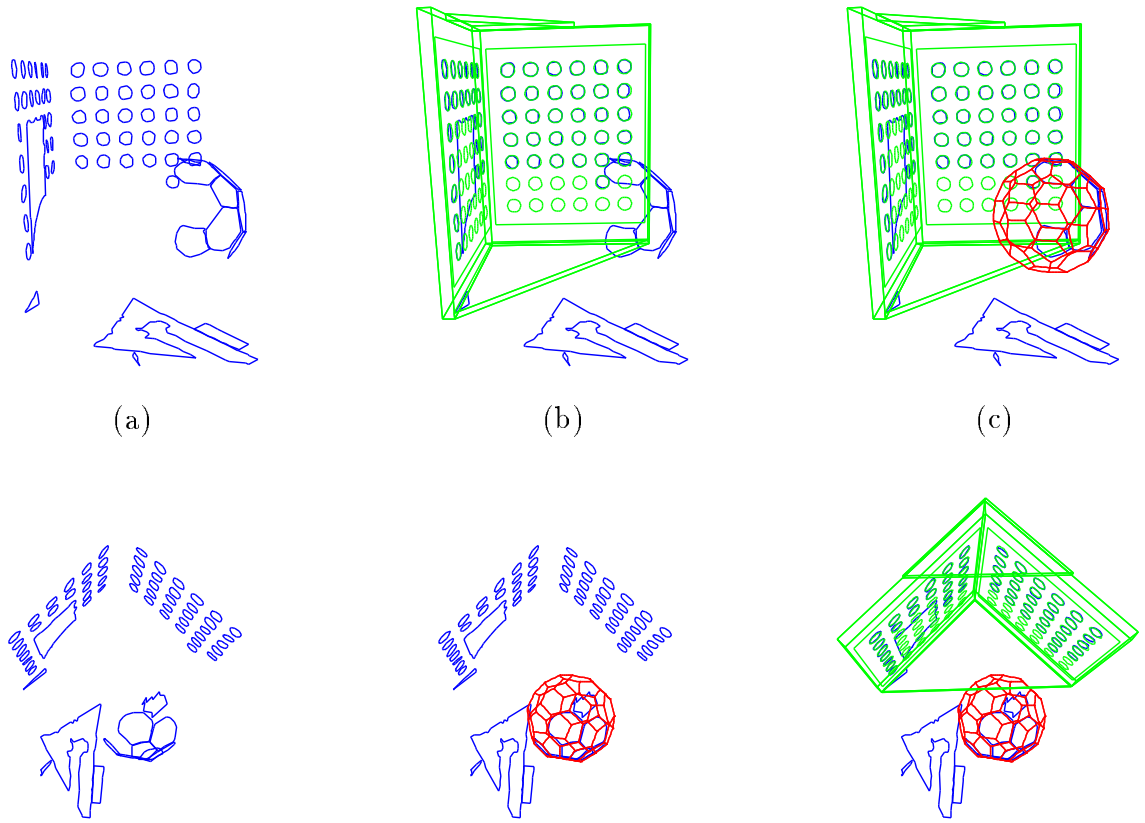


Figure 11: *(a) and (d) are two points of view of the reconstructed 3D data. In (b) the planar object is registered to the 3D data. In (e) only the soccer ball is registered, whereas (c) and (f) show the registration of both planar and 3D objects. Geometric models are registered with the coarse method and a fitting algorithm is used to produce an accurate registration.*

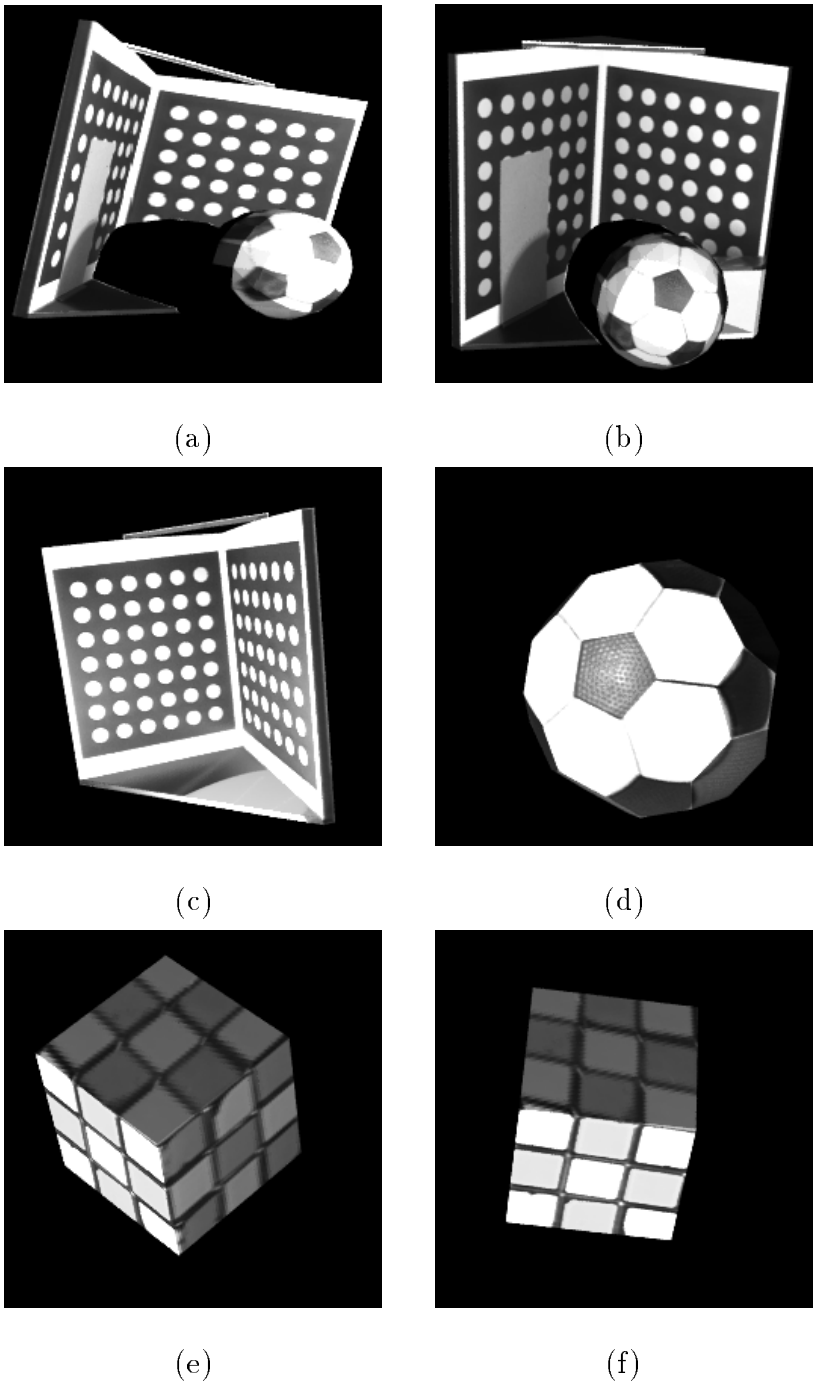
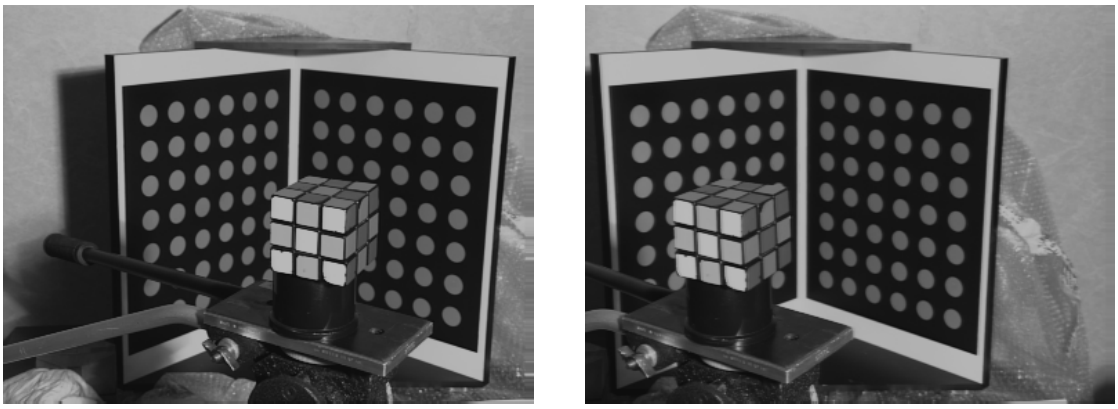


Figure 12: *(a)(b) Synthetic points of view of the registered models (see Figure 11(c)(f)) on which the original image 10(a) is projected. (c) Synthetic point of view of the two-plane object from another stereo scene. (d) Synthetic point of view of the soccer ball. (e)(f) Synthetic points of view of the registered model (see Figure 14(b)) on which the original image 13(a) is projected.*

3.5 Further significant features



(a) (b)

Figure 13: *(a)(b) Left and right images of a real scene processed to obtain Figure 14.*

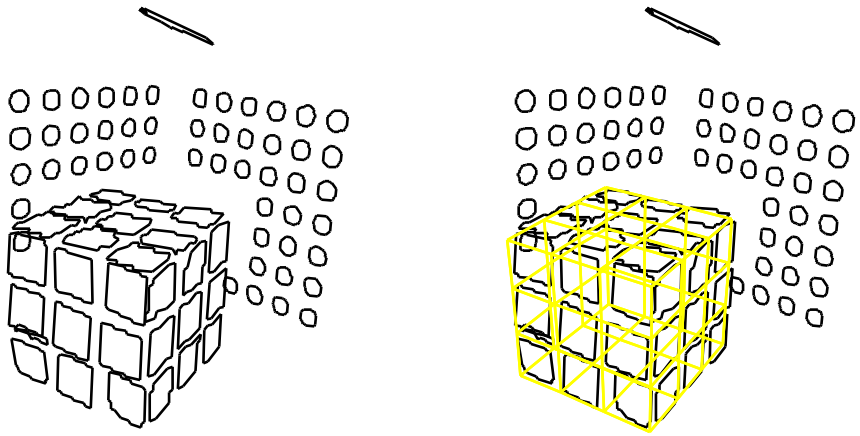
Since all the planar patches have the same shape, the 3D registration of the rubik cube (Figure 14) is difficult. Indeed, the first step of our algorithm can not succeed in the selection of interesting pairs of planar patches. The rubik cube has 54 patches and the reconstructed data set has 78.

To improve the 3D registration, the color of the planar patch is taken into account by replacing the criterion in (1) by the following equation:

$$C_{ij} = \frac{1}{\max(1, \frac{|\alpha_i - \alpha_j|}{I})} \frac{1}{\max(1, \frac{|\beta_i - \beta_j|}{I})} \frac{1}{\max(1, \frac{|r_i - r_j|}{G})} \frac{1}{\max(1, \frac{|g_i - g_j|}{G})} \tag{4}$$

where G is the mean error on the color data. Here, r_i and g_i are the red and green normalized components of the color of the planar patch i. Normalized color is used to compensate variations in intensity as a function of the viewing direction.

Using color ($G = 30$ gray levels), the registration shown in Figure 14 is obtained. Color is taken as an example, but any additional information on relevant features can be used if it is invariant under rigid transformations. In particular, for planar patches, invariants composed of third order moments provide other interesting features.



(a) (b)

Figure 14: *(a) View of the 3D reconstruction obtained by a stereo analysis process on Figure 13. (b) The geometric model of the rubik cube is registered to the 3D reconstruction data.*

3.6 A tool for 3D data sets comparison

Once the final registration result is obtained, it is easy to compute information about the overlapping part of both 3D data sets, such as the relative size of these parts and the mean displacement of the matched points in the common part. These measures are important to quantify the quality of the registration.

We designed a generic tool to estimate the similarity of two 3D data sets in the following manner: we choose one set as the reference set, and each point of this set is matched with the closest point of the second set. To have significant estimations and to overcome the fact that points in the data set and its models cannot be in correspondence, the edges of whole planar patches are uniformly sampled. We assume that the two 3D data sets have an overlapping part that is not too small. Thus, as shown in Figure 15, the histogram of the length of the displacement vector between matched points (these displacements are represented by the arrows in Figure 6) has a peak near zero. Consequently, we cluster the set of rigid transformations of the matched points into two clusters:

- small displacements which are associated with points in the common part of both data sets,

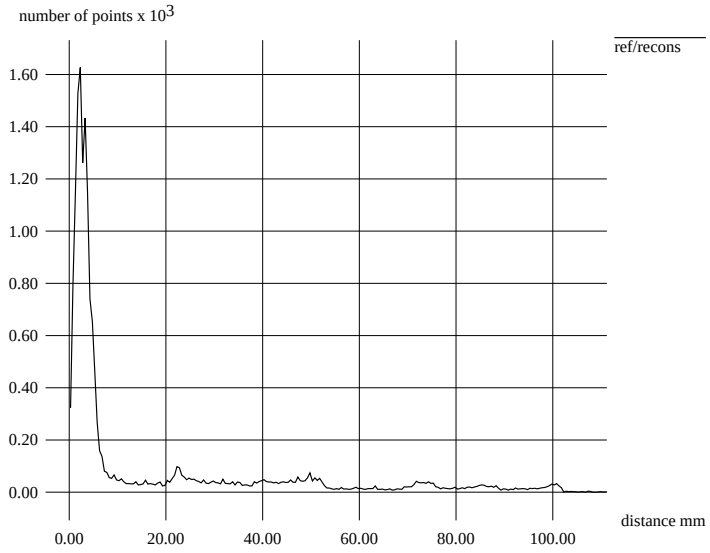


Figure 15: *Histogram of the length of the displacement vectors between a point of the reference data set and the closest point of the other set. In this example, the reference data set is the object shown in Figure 16 and the other set is the 3D reconstruction shown in Figure 11.*

- other displacements which are associated with not well matched points and which cannot therefore be in the common part.

The clustering algorithm used is the Davé's version of the FCM with one cluster and a noise cluster ($c = 2$). Consequently, we take advantage of the fuzzy approach to set the value of the radius of the normal cluster only in approximative way. The magnitude of this radius can be obtained by looking the same kind of histogram than in Figure 15.

The clustering of the displacement set provides a split of the reference 3D data set in two parts where one of these parts is common with the other 3D data set, as shown in Figure 16. A significant measure about the common part between the data set and its model is the relative size of the common part in terms of well matched points. For example, with the 3D data sets shown in Figure 11(c), the size of the common part between the 3D reconstruction and the soccer ball is 40% relative to the ball size, and 19% relative to the reconstruction size.

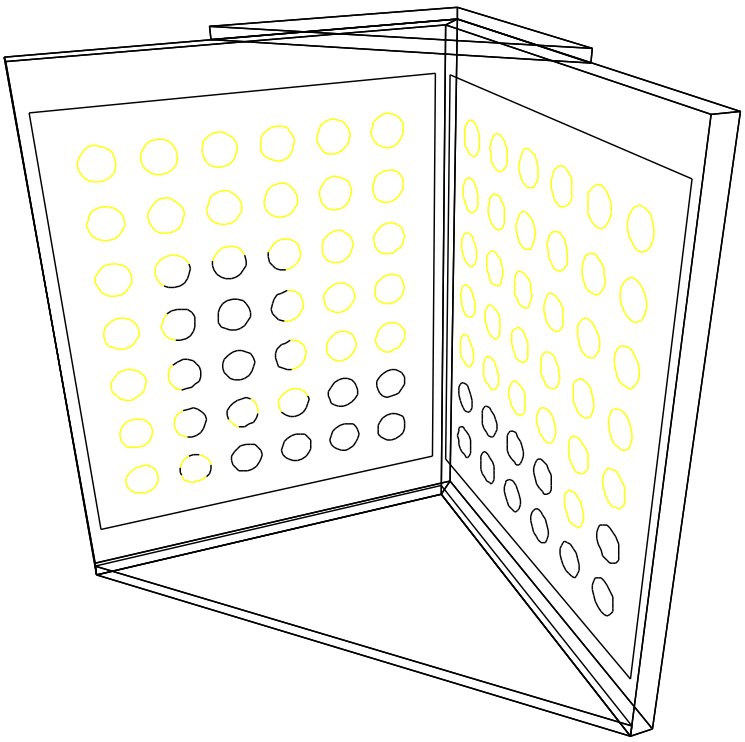


Figure 16: *Split of the reference data set into its part common with the reconstruction (in grey) and its non-common portion (in black). The 3D reconstruction is shown in Figure 11.*

Moreover, on the same data, we compute the bias and standard deviation of the matched points in the common part. These measures give the overall global accuracy of the reconstruction and registration process. We have obtained a mean accuracy of 5.7 mm for the reconstruction and registration process on the soccer ball. The depth of the soccer ball is 1200 mm and the stereoscopic base is 340 mm. In this particular scene, the reconstruction error is in theory around 1.2 mm in the front plane and 9 mm along camera axis. More generally, we have observed with our reconstruction algorithm, a mean accuracy of 4 mm on several registration tests, consequently we set I equal to 4 mm in the criterion of the formula 1.

4 Conclusion

We were proposed a new, robust, 3D registration method. In our approach, a coarse 3D object localization is first obtained. This step is based on a fuzzy clustering algorithm to take advantage of robustness with respect to initialization, outliers and noisy data. The output consists of several hypothesis of registration. Then, a fine local fitting is performed to achieve accurate 3D registration on each hypothesis.

We apply the proposed method to the registration of planar patch objects. Our approach is generic, and this work can be extend and adapted to deal with other kinds of representations. Moreover, the method is easily controlled by a small number of parameters: the most important are the number c of clusters and the resolution I.

Our algorithm can be computationally expensive for objects with more than 1000 patches, but it seems that is the cost to pay to have a good robustness to noise and occlusion.

Interesting results for a geometric model-based interpretation are obtained, which opens a new perspective of investigation in data modeling by doing the registration of several prototypes at the same time.

5 Appendix

In this section, we present the outline of the proof of convergence through strict local minima of the FCM algorithm when the confidence value C_k is introduced as proposed in section 2.2.2. This proof is adapted from the original proof of the FCM algorithm [3] with the same condition ($m > 1$). A similar extension can be done from the proof of Davé's algorithm.

Let matrix $U = (u_{ik})$ and vector $V = (v_i)$. We want to minimize the following functional:

$$J_m(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n} C_k u_{ik}^m d^2(x_k, v_i) \quad \text{with} \quad \sum_{i=1}^{c} u_{ik} = 1 \quad \text{and} \quad u_{ik} \geq 0 \tag{5}$$

where d is the Euclidean distance. The proof of the theoretical convergence is based on the application of the convergence theorem of Zangwill [3], after an infinite number of iterations (as in steepest descent and Newton's method).

First, we fix V. Since the columns of the matrix U are independent we have:

$$\min(J_m(U)) = \sum_{k=1}^{n} C_k \min_{\sum_{i=1}^{c} u_{ik}=1} (\sum_{i=1}^{c} u_{ik}^m d^2(x_k, v_i))$$

Thus, for each term, it is a minimization under constraint. Consequently, we use Lagrange multipliers to solve this minimization and obtain after derivations:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} (\frac{d_{ik}}{d_{jk}})^{\frac{2}{m-1}}} \tag{6}$$

Secondly, we fix U. Let $< . >$ be the scalar product. Then, we minimize the following term:

$$J_m(V) = \sum_{i=1}^{c} \sum_{k=1}^{n} C_k u_{ik}^m < x_k - v_i . x_k - v_i >$$

Thus for each i, it is necessary that the directional derivatives $J'_m(v_i, g)$ vanish for all unit vectors $g \in \mathbb{R}^c$:

$$J'_m(v_i, g) = -2 \sum_{k=1}^{n} C_k u_{ik}^m < x_k - v_i . g >= 0 \qquad \forall g$$

$$\Leftrightarrow \sum_{k=1}^{n} C_k u_{ik}^m (x_k - v_i) = 0$$

We deduce:

$$v_i = \frac{\sum_{k=1}^{n} C_k u_{ik}^m x_k}{\sum_{k=1}^{n} C_k u_{ik}^m} \tag{7}$$

The vector v_i can be interpreted as the center of gravity of the set (x_k) where each point is weighted by $C_k u_{ik}^m$. We note that (6) and (7) allow us to construct steps 2 and 3 of the proposed clustering algorithm described in section 2.2.2.

A dual interpretation of this functional (5) is:

$$J_m(U, V) = \sum_{i=1}^{c} \sum_{k=1}^{n} w_{ik}^m d^2(x_k, v_i)$$

where $w_{ik} = \sqrt[m]{C_k} u_{ik}$. This equation is the same functional as the one in the FCM algorithm, where the constraint on w_{ik} is not $\sum_{i=1}^{c} w_{ik} = 1$ but $\sum_{i=1}^{c} w_{ik} = \sqrt[m]{C_k}$. Consequently, we can understand our proposed generalization, which introduces a confidence weight C_k on each point, as the FCM algorithm with a non-homogeneous constraint on the fuzzy memberships. This remark leads to an interesting connection with the possibilistic approach to clustering [16].

References

[1] N. Ayache. Medical computer vision, virtual reality and robotics. *Image and Vision Computing*, 13(4):295–313, 1995.

[2] P.J. Besl and N.D. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

[3] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, 1981.

[4] R. M. Bolle and D. B. Cooper. On optimally combining pieces of information, with application to estimating 3-D complex-object position from range data. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(5):619–638, 1986.

[5] N. Boujemaa and G. Stamon. Fuzzy modeling in early vision: Application in medical image segmentation. *Progress in Image Analysis and Processing III*, pages 649–656, 1994.

[6] R.N. Dave. Characterization and detection of noise in clustering. *Pattern Recognition Letters*, 12(11):657–664, 1991.

[7] 0. Faugeras and M. Hebert. The representation, recognition, and locating of 3-D shapes from range data. *Inter. J. Robotics Research*, 5(3):27–52, Fall 1986.

[8] A. Guéziec and N. Ayache. Smoothing and matching of 3D-space curves. In LNCS Series Vol. 588 Springer Verlag, editor, *Proceedings, ECCV '92 (Second European Conference on Computer Vision, Santa Margherita Ligure, Italy, May 1992)*, pages 620–629, 1992.

[9] A. Hoogs and D. Hackett. Model-supported exploitation as a framework for image understanding. In Morgan Kaufman, editor, *Proceedings of the ARPA Image Understanding Workshop*, pages 265–268, San Francisco, CA, November 1994.

[10] P. Horaud and R.C. Bolles. 3DPO's strategy for matching three-dimensional objects in range data. *International Journal of Robotics Research*, 5(3):3–26, 1986.

[11] P. Jancène, F. Neyret, X. Provot, J.-P. Tarel, J.-M. Vézien, C. Meilhac, and A. Vérroust. RES: computing the interactions between real and virtual objects in video sequences. In *Second IEEE Workshop onNnetworked Realities (NR'95)*, pages 27–40, Boston, Massachusetts (USA), October 1995. http://www-rocq.inria.fr/syntim/textes/nr95-eng.html.

[12] B. Kamgar-Parsi, J. Lones, and A. Rosenfeld. Registration of multiple overlapping range images: Scene without distinctive features. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(9):857–870, September 1991.

[13] V. Koivunen and J.-M. Vézien. Machine vision tools for CAGD. *International Journal of Pattern Recognition and Artificial Intelligence*, 10(2):165–182, 1996.

[14] D. J. Kriegman and J. Ponce. On recognizing and positioning curved 3-D objects from image contours. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(12):1127–1137, December 1990.

[15] R. Krishnapuram and C.P. Freg. Fitting an unknown number of lines and planes to image data through compatible cluster merging. *Pattern Recognition*, 25(4):385–400, 1992.

[16] R. Krishnapuram and M. Keller. A possibilistic approach to clustering. *IEEE Trans. Fuzzy Systems*, 1(2):98–110, May 1993.

[17] R. Krishnapuran and D. Casasent. Determination of three dimensional object location and orientation from range data. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 11(11):1158–1167, November 1989.

[18] I. Rigoutsos and R. Hummel. A Bayesian approach to model matching with geometric hashing. *Computer vision and image understanding: CVIU*, 62(1):11–26, July 1995.

[19] H.S. Yang and A.C. Kak. Determination of the identification, position and orientation of the topmost object in a pile. *CVGIP*, 36(2/3):229–255, November 1986.

[20] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994. also INRIA Tech. Report #1658.

[21] X. Zhuang and Y. Huang. Robust 3-D 3-D pose estimation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(8):818–824, August 1994.