Fast and accurate surface alignment through an isometry-enforcing game

Andrea Albarelli [a,*], Emanuele Rodolà [b], Andrea Torsello [a]

[a] Department of Environmental Science, Informatics, and Statistics, Ca' Foscari University of Venice, Italy
[b] Department of Computer Science, Technische Universität München, Germany

ABSTRACT

Surface registration is often performed as a two step process. A feature matching scheme is first adopted to find a coarse initial alignment between two meshes. Subsequently, a refinement step, which usually operates in the space of rigid motions, is applied to reach an optimal registration with respect to pointwise distances between overlapping areas. In this paper we propose a novel technique that allows to obtain an accurate surface registration in a single step, without the need for an initial motion estimation. The main idea of our approach is to cast the selection of correspondences between points on the surfaces in a game-theoretic framework, where a natural selection process allows matching points that satisfy a mutual rigidity constraint to thrive, eliminating all the other correspondences. This process yields a very robust inlier selection scheme that does not depend on any particular technique for selecting the initial strategies as it relies only on the global geometric compatibility between correspondences. The practical effectiveness of the approach is confirmed by an extensive set of experiments and comparisons with state-of-the-art techniques.

1. Introduction

Surface alignment is a pervasive problem throughout Computer Vision literature. In fact, it finds useful applications as a tool for in-line quality control [1], 3D object recognition [2], advanced human interfaces [3], and SLAM [4], just to name a few. For this reason, surface registration is one of the most studied topics in the field of 3D data acquisition and processing. With this paper we try to introduce a fresh view on the problem by proposing a game-theoretic approach that is robust to noise and allow to attain a very accurate registration without requiring an initial motion estimation.

1.1. Fine and coarse registration techniques

The distinction between fine and coarse surface registration methods is mainly related to the different strategies adopted to find pairs of corresponding points to be used for the estimation of the rigid transformation. Almost invariably, fine registration algorithms exploit an initial guess in order to constrain the search area for compatible mates and minimize the risk of selecting outliers. On the other hand,

coarse techniques, which cannot rely on any motion estimation, must adopt a matching strategy based on the similarity between surface-point descriptors or resort to random selection schemes. The tension between the precision required for fine alignment versus the recall needed for an initial motion estimation stands as the main hurdle to the unification of such approaches.

The vast majority of current fine alignment methods are modifications to the original ICP proposed by Zhang [5] and Besl and McKay [6]. These variants generally differ in the strategies used to sample points from the surfaces, reject incompatible pairs, or measure error. In general, the precision and convergence speed of these techniques is highly data-dependent and sensitive to the fine-tuning of the model parameters. Several approaches that combine these variants have been proposed in the literature in order to overcome these limitations (see [7] for a comparative review). No matter which variant is used, ICP, being an iterative algorithm based on local, step-by-step decisions, is susceptible to the presence of local minima. Some recent variants mitigate this problem by avoiding hard culling assigning a probability to each candidate pair by means of evolutionary techniques [8] or Expectation Maximization [9]. Other, non-ICP-based, fine registration approaches include the well-known method by Chen [10] and signed distance fields matching [11].

Coarse registration techniques can be roughly organized into three main classes: global methods, feature-based methods and techniques based on RANSAC [12] or PROSAC [13] schemes. Global

* Corresponding author. Tel.: +39 041 2348465 (IT).
E-mail addresses: albarelli@unive.it (A. Albarelli), rodola@in.tum.de (E. Rodolà), torsello@dsi.unive.it (A. Torsello).

methods such as PCA [14] or Algebraic Surface Model [15] exploit some global property of the surface and are thus very sensitive to occlusion. Feature-based approaches aim at the localization and matching of interesting points on the surfaces. They are more precise and can align surfaces that exhibit only partial overlap. Nevertheless, the unavoidable localization error of the feature points prevents them from obtaining accuracies on par with fine registration methods.

A completely different coarse registration approach is the one taken by RANSAC-based techniques. DARCES [16] is based on the random extraction of sets of mates from the surfaces and their validation based on the accuracy of the estimated transformation. The more recent Four Points Congruent Sets method [17] follows a similar route, but filters the data to reduce noise and performs early check in order to reduce the number of trials.

A recent and extensive review of many different methods can be found in [19]. Regardless of the criteria used to obtain pairs of mating points, the subsequent step in surface registration is to search for the rigid transformation that minimizes the squared distance between them. Many mature techniques are available to do this (see for instance [20]).

1.2. Feature detection on 3D objects

Feature detection and characterization is a key step in many tasks involving the recognition, registration or database search of 2D and 3D data. Specifically, when suitable interest points are available, all these problems can be tackled by working with the set of extracted features rather than dealing with the information carried by the whole data, which is less stable and noisier. For an interest point to be reliable it must exhibit two properties: repeatability and distinctiveness. A feature is highly *repeatable* if it can be detected with good positional accuracy over a wide range of noise levels and sampling conditions as well as different scales and transformations of the data itself. Further, description vectors calculated over interesting points are said to be *distinctive* if the descriptors related to different features lie far apart in feature space, while descriptors associated to multiple instances of the same point lie within a small distance from one another. These properties are somewhat difficult to attain since they are subject to antithetical goals: In fact, to achieve good repeatability despite of noise, larger patches of data must be considered, which unfortunately leads to a lower positional precision and a less sharp culling of uninteresting points. Moreover, for descriptor vectors to be distinctive among different features, they need to adopt a large enough basis, which, owing to the well known "curse of dimensionality," also affects their coherence over perturbed versions of the same feature. In the last two decades these quandaries have been addressed with great success in the domain of 2D images, where salient points can be localized with sub-pixel accuracy using detectors exploiting strong local variation in intensity, such as Harris Operator [21] and Difference of Gaussians [22], or using techniques that are able to locate affine invariant regions, such as Maximally Stable Extremal Regions (MSER) [23] and Hessian-Affine [24]. Among the most used descriptors are the Scale-invariant feature transform (SIFT) [25], the Speeded Up Robust Features (SURF) [26] and Gradient Location and Orientation Histogram (GLOH) [27]. While these approaches work well with 2D intensity images, they cannot be easily extended to handle 3D surfaces since no intensity information is directly available. On the other hand, there has been huge effort to use other local measures, such as curvature or normals. One of the first descriptors to capture the structural neighborhood of a surface point was described by Chua and Jarvis, who with their Point Signatures [28] suggest both a rotation and translation invariant descriptor and a matching technique. Later on, Johnson and Hebert introduced Spin Images [29], a rich characterization obtained by binning the radial and planar

distances of the surface samples respectively from the feature point and from the tangent plane. Given their ability to perform well with both surface registration and object recognition, spin Images have become one of the most used 3D descriptors. More recently, Pottmann et al. proposed the use of Integral Invariants [30], stable multi-scale geometric measures related to the curvature of the surface and the properties of its intersection with spheres centered on the feature point. Zaharescu et al. [31] presented a comprehensive approach for interest point detection (MeshDOG) and description (MeshHOG), based on the value of any scalar function defined over the surface (i.e. curvature or texture, if available). MeshDOG localizes feature points by searching for scale-space extrema over progressive Gaussian convolutions of the scalar function and thus by applying proper thresholding and corner detection. MeshHOG calculates a histogram descriptor by binning gradient vectors with respect to a rotationally invariant local coordinate system. Finally, the recent SHOT descriptor [32], introduced by Tombari et al. exploits a novel 3D reference frame to offer enhanced descriptive power and robustness. The reader interested in a comparison between recent geometric descriptors and feature descriptions can find an in-depth coverage within [33].

In the following sections we introduce a novel pipeline that can be used to obtain an accurate surface registration without requiring an initial motion estimation. The contribution is twofold. First we propose very simple descriptors, named *Surface Hashes*, that span only 3–5 dimensions. As their name suggests, we expect Surface Hashes to be repeatable through the same feature point, yet to suffer from a high level of clashing due to their limited distinctiveness. In order to overcome this liability we also adopt a robust game-theoretic inlier selector which exploits rigidity constraints among surfaces to guarantee a global geometric consistency. The combination of these loosely distinctive features and our robust matcher leads to an effective and robust surface alignment approach.

2. Game theoretical matching

The use of tools coming from Game Theory for matching purposes is rather novel: it has been introduced as an effective heuristic in the domain of graph matching in [34] and has recently been applied successfully to other Computer Vision scenarios [38,35,37]. In particular, the concept of Game-Theoretic Matching (GTM) has a central role within the framework discussed in this paper and will be explored thoroughly in the next two sections.

2.1. Basic ideas

This kind of matching process is performed on the basis of two premises. The first is the ability to model a set of potentially matching features as strategies in a non-cooperative game. The second requirement is the availability of a suitable payoff function between strategies that measures how well two matches behave together. The main idea that underlies the proposed technique is quite simple: if both the strategy definition criterion and the payoff function are chosen properly with respect to a given problem, it is reasonable to think that a subset of strongly mutually compatible strategies (i.e. putative matches) can be found embedded in the initial set. Those matches must be found and isolated from the outliers with some selection technique.

In principle, those assumptions are not that different from those that subtend to the very popular RANSAC inlier selection method. In fact, any RANSAC or PROSAC flavor assumes that a subset of large consensus exists in correspondence with the correct solution. In addition, some compatibility function is always defined in order to evaluate the consensus that a proposed solution receives from all the remaining samples. However, the loose connection between GTM and

RANSAC breaks as we analyze the selection process itself. Specifically, in GTM there is no majority validation for a random subset. Rather, the selection happens by letting the strategies compete in a non-cooperative game. The game starts with an initial population where each strategy is played by an equal percentage of players. Such population is then evolved through the action of discrete time replicator dynamics until it reaches some stable state from which a (conceivably) correct matching set can be extracted.

More formally, we model the evolutionary success of each feasible match (among an initial set of n) through a probability distribution represented by a point in the n-dimensional standard simplex

$$\mathbf{x} \in \Delta^n = \left\{ \mathbf{x} \in \mathbb{R}^n : x_i \geq 0 \text{ for all } i \in 1 \dots n, \sum_{i=1}^{n} x_i = 1 \right\}.$$

Furthermore, we model the mutual support between matches i and j as a non-negative, symmetric payoff function $\Pi = (\pi_{ij})$.

It can be shown [39] that under very loose assumptions, any initial distribution $\mathbf{x}$ that is evolved through the replicator dynamics [40]

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) \frac{(\Pi \mathbf{x}(t))_i}{\mathbf{x}(t)^T \Pi \mathbf{x}(t)} \tag{1}$$

converges to an asymptotically stable state corresponding to a local maximum of the average payoff $\mathbf{x}^T \Pi \mathbf{x}$.

Previous research has demonstrated that by properly choosing function Π, such local maximum can correspond to locally optimal configurations for a given problem. Furthermore, despite the locality of the solution, obtained results have often been able to compete with the state-the-art for each context. In the following subsections two closely related approaches are described, afterwards, the contribution of this paper with respect to the use of the general framework within the specific context of surface registration is discussed.

2.2. Structure from motion

Correspondence selection using Game Theory has been widely explored in [35]. In such work, however, the matching happens between images rather than 3D surfaces. This results in a rather different scenario, since the loss of information due to the projective imaging process hinders the ability to use Euclidean constraints for matches validation. This limitation is solved by exploiting the scale and orientation information provided by 2D feature descriptors. Namely, corresponding features are expected to exhibit similar affine transformations, at least within a local neighborhood. Thus, as the transformation associated to one match is applied to one other, the latter it is expected to be projected near the location of its mate, and vice versa. If this does not happen, then the two correspondences are not compatible.

To this end, the payoff function has been defined to be decreasing with the growth of the distance between correspondence hypotheses and reprojected features (see Fig. 1). Of course this is a coarse approximation of the reality and only works if features are close enough for the "affine hypothesis" to hold. In fact, to be able to select enough correspondences between two images, the game has to be repeated over and over, producing several partial results, each one corresponding to an ideal parallax plane. Moreover, the coarse nature of the result, required a further refinement of the selection based on the enforcing of epipolar constraints between the images.

Even considering these limitations, Ref. [35] represents a seminal work in the field of Game Theoretical matching, and we would direct the interest user to it for more in-depth details about this kind of selection process.
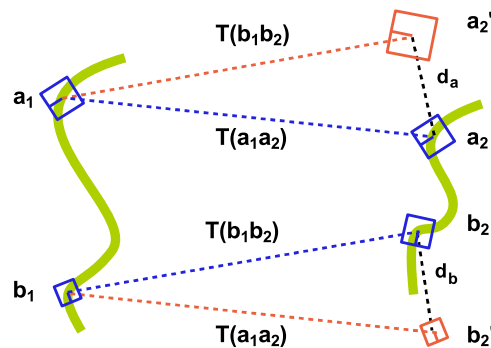

Fig. 1. The payoff between two pairs of matching features is inversely proportional to the maximum reprojection error obtained by applying the affine transformation estimated by one to the other.

2.3. Object recognition in cluttered scenes

Three dimensional surfaces are first addressed using Game Theory in [36]. The focus of this work is on the detection of an object within cluttered scenes. The reference object is supplied as a complete model and the scene into which it can be searched can be anything, from stereo reconstruction, to 3D scans or synthetic layouts.

Given the general conditions and the specific scope, some scenario-specific choices have been made during the design of the method.

For starters, since the goal of the technique is not precise surface alignment, but just object detection, the accuracy of the matches has been sacrificed in spite of coverage. Namely, the initial point selection is performed using just unary implicit information (curvature), rather than seeking for a small number of very distinctive points. This is needed since, as clutter and occlusion grow, the probability of finding correspondences decreases steeply if the initial sample becomes too tight. In addition, occlusion could result in artificially distinctive points around border areas that are not repeatable in the model as they do not actually exist.

Furthermore, the descriptor adopted to compute the payoff has been explicitly designed to boost recall over precision, as it only gathers mono dimensional information along the direction of the two involved correspondences. This was needed for several reasons, including the requirement of performing a scale-independent recognition and of being robust to strong occlusion. In addition, recall is further boosted by the fact that, with the proposed scenario, the model is complete and noiseless, allowing for an off-line pre-computation of a large number of dependable descriptors that can be collected in a reference dictionary.

Finally, differently from [35], Euclidean distances could be (in principle) enforced within the payoff function. Still, this is not actually done, since such choice would have ruled out the ability of making the method scale independent.

All these trade-offs allowed the approach to be really effective with the problem it was created for, that is matching a model to a general scene. However, as shown in the experimental section of this paper, its performance is not first-rate if directly adopted to seek accurate surface registration.

2.4. Contribution of this paper

With this paper we are proposing to address the shortcomings exhibited by [36] and [46] as a surface registration tool. Clearly, this goal is reached by leveraging on the different characteristic of the alignment problem.

The scenario we are dealing with is the one arising when a 3D reconstruction technique is adopted to produce partial views of an

object or a scene. The goal is to align such partial reconstructions yielding a low residual error between surfaces. While this is not required, all the surfaces are usually been produced by the same (or similar) sensor and a complete and noiseless reference model is not available. The overlap is usually partial, however the clutter is limited since, besides effects due to occlusion, objects do not appear or disappear form the scene. In fact we can deem the scene to be made up of rigid objects which do not move and which preserve size and scale between subsequent acquisitions.

The first consequence of this set of conditions is that we are interested in a (relatively) small number of very accurate correspondences rather that trying to be as much inclusive as possible. To this end we propose an initial feature culling (in Section 4.1) that is very aggressive and aim at keeping only very distinctive and well localized points.

Such points, in turn, are characterized and initially matched through quite loose surface descriptors (see Section 3.4) that are fast to compute and can combined together to reach the desired level of descriptiveness. Experiments will show that, according to the premises, stricter characterization is to be preferred, even at the expense of the overall number of correspondences.

Note however, that the descriptors are not used at all during the selection process. This is in strong contrast with [36], where they are actually part of the payoff function. This is possible because the rigidity assumption and the fixed scale allow to use a payoff that is computed using only Euclidean distances, thus enforcing the isometry between different surfaces. Additionally, this assumption allows to guarantee a sound theoretical relation between the payoff function and optimal alignments, as described in Section 5.

Besides the theoretical properties of the proposed method, its practical effectiveness is discussed at length in the experimental section where it is tested under several different conditions and compared with state-of-the-art alignment methods.

3. An isometry-enforcing alignment game

In our approach, we let matches compete with one another, each obtaining support from compatible associations and competitive pressure from all the others. At equilibrium, only pairings that are mutually compatible should survive and are then taken to be inliers. In practice, the full process happens through the following steps: First, we extract a set of candidate matches, then we define a payoff function between such candidates, finally, we use the replicator dynamics (1) to evolve towards an equilibrium state. These steps will be detailed in the following three sections.

3.1. Matches as strategies

Since we will deal with the registration of two different surfaces we will refer to the points belonging to the first surface with the term *model points*, while we will use the term *data points* with respect to the second surface. This distinction is captious since there is no actual difference in role between the two surfaces, however it

is consistent with the current registration literature and helps in defining an order within matches.

Given the set of all model points M and the set of all data points D we call a *matching strategy* any pair (a_1, a_2) with $a_1 \in M$ and $a_2 \in D$ (see Fig. 2). We call the set of all the matching strategies S. Our goal is to extract from S the subset of correct matches, that is, strategies that associate a point in the model surface with the same point in the data surface. In principle, all the model and data points could be used to build the matching strategies, thus giving $S = M \times D$. In practice, however, this would lead to a very big set of candidates with a huge portion of outliers. In order to reduce the number of the initial candidates, we subsample model points by keeping only points that are deemed to be interesting, and we use a semi-local surface descriptor (Section 4) to assign to each of them a small set of feasible matches. This is done simply by choosing the mates with the nearest descriptor in the Euclidean sense. The amount of model subsampling, the level of distinctiveness required, and the number of matching candidates to select for each model point is parameters of the method and can be modulated to balance speed and accuracy. Of course, the distinctiveness of the descriptor used to characterize the data points has a big influence in fixing a reasonable number of candidates for each match. This observation, in turn, raises the quandary between the repeatability and the distinctiveness of feature descriptors. In fact, while a high distinctiveness is always desirable, this often comes at the price of much more instability with respect to noise and thus can lead to a poor repeatability. Fortunately, with our approach, the descriptor itself is only used to construct the set S and has no role in the evolutionary selection, which is purely driven by the payoff function. For this reason we find it reasonable to resort to a feature characterization that is scarcely distinctive and to allow for several candidate matches, leaving the game-theoretic selection process to operate a severe culling. The loose descriptor that we are introducing has many other advantages, such as being easy to implement and fast to compute, and it will be described in full depth in Section 4.

3.2. Definition of a payoff function

Since the set of strategies S is built by proposing several attainable matches for each considered model point it is clear that the number of outliers in S will be far superior to the number of correct correspondences. In fact, given that the surfaces are not warped, we expect the correspondences to be one-to-one, thus at most one match for each set of strategies that involve the same model or data point can be correct. In order to extract this minority of correct matches buried into S, the GTM framework must exploit the consistency of any pair of those strategies with respect to some property. By contrast, all the other wrong and thus randomly paired matches should not exhibit a wide agreement of the same property. This degree of compatibility is usually expressed through a (symmetric) real valued function $\pi : S \times S \to \mathbb{R}^+$ which is usually materialized in a symmetric payoff matrix Π (see Fig. 3). Again, this concept is somewhat similar to the basic idea that drives RANSAC. However, by casting the search in an evolutionary process, we are not relying on a one-shot vote counting validation, but rather we are using self-validation to drive to a consensus what can be seen as an iterated voting process.

The very nature of the property to which pairs of correct matches must adhere depends on the exact scenario to which GTM is applied. For instance, in [34] adjacency relations of graphs built on segments extracted from the images are used; in [35] the payoff between pairs of potential matches is boosted by the compatibility of the locally affine transformation assigned to points by the SIFT descriptor, and in [38] the similarity between potential matches is associated with a measure of metric distortion on deformable 3D shapes. Also, in [37] the payoff is used both to cluster patches of nondistinctive features and to match the remaining ones.
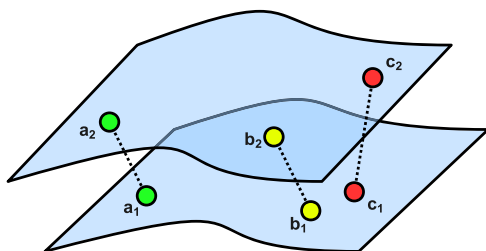

Fig. 2. Example of three matching strategies.

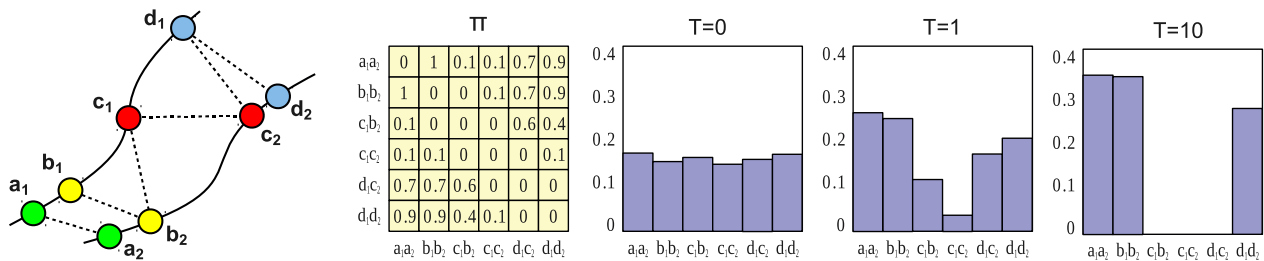| π	a₁a₂	b₁b₂	c₁b₂	c₁c₂	d₁c₂	d₁d₂
a₁a₂	0	1	0.1	0.1	0.7	0.9
b₁b₂	1	0	0	0.1	0.7	0.9
c₁b₂	0.1	0	0	0	0.6	0.4
c₁c₂	0.1	0.1	0	0	0	0.1
d₁c₂	0.7	0.7	0.6	0	0	0
d₁d₂	0.9	0.9	0.4	0.1	0	0

Fig. 3. An example of the evolutionary process. Four points are sampled from the two surfaces and a total of six mating strategies are selected as initial hypotheses. Matrix Π shows the compatibilities between pairs of mating strategies according to a one-to-one rigidity-enforcing payoff function. Each mating strategy got zero payoff with itself and with strategies that share the same source or destination point (e.g., $\pi((b_1, b_2), (c_1, b_2)) = 0$). Strategies that are coherent with respect to a rigid transformation exhibit high payoff values (e.g., $\pi((a_1, a_2), (b_1, b_2)) = 1$ and $\pi((a_1, a_2), (d_1, d_2)) = 0.9$), while less compatible pairs get lower scores (e.g., $\pi((a_1, a_2), (c_1, c_2)) = 0.1$). Initially (at $T=0$) the population is set to the barycenter of the simplex and slightly perturbed (3–5%). After just one iteration, (c_1, b_2) and (c_1, c_2) have lost a significant amount of support, while (d_1, c_2) and (d_1, d_2) are still played by a sizable amount of population. After ten iterations ($T=10$), (d_1, d_2) has finally prevailed over (d_1, c_2) (note that the two are mutually exclusive). Note that in the final population $(a_1, a_2), (b_1, b_2)$ have a larger support than (d_1, d_2) since they are a little more coherent with respect to rigidity.
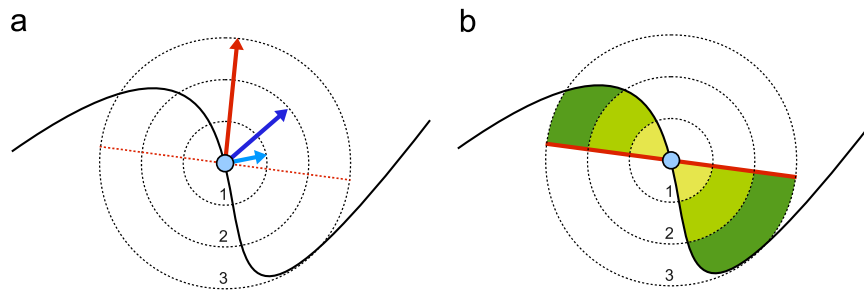

Fig. 4. Example of the two basic Surface Hashes proposed in this paper: (a) Normal hash and (b) integral hash.

Since in this paper we are dealing with rigid alignment of surfaces, it is quite natural to exploit the rigidity constraint to measure the feasibility of a pair of matches. In fact (as described in Section 5) we relax the rigidity assumption to an isometry assumption, assigning a high payoff to pairs of matching strategies that preserve the Euclidean distance between the corresponding points on model and data (see Fig. 3). Also, we will assign a payoff equal to zero to pairs that share the same source or destination point, so as to enforce a one-to-one matching.

3.3. Evolution through replicator dynamics

The goal of the selection process is to evolve an initial mixed strategy $\mathbf{x}$ to a stable state from which the non-extinct matches will be extracted. Since $\mathbf{x}$ is associated to strategies in S, it can be seen as a point in $\Delta^{|S|}$. Each coordinate of such a point expresses the proportion of population playing a strategy, and for this reason we will refer to $\mathbf{x}$ as a *population vector*. At the beginning of the process, $\mathbf{x}$ is initialized near the barycenter of $\Delta^{|S|}$. In practice this means that to each element of $\mathbf{x}$ is assigned the amount of population $1/|S|$ and then $\mathbf{x}$ is slightly perturbed. The perturbation serves two goals: First, to introduce some randomness in the process (which can be useful if different alternative configurations are sought); second, to avoid some rare stall conditions in replicator dynamics that could occur with some highly symmetric payoff configurations.

This initial population is evolved by means of Eq. (1) until convergence. In order to assess when this happens, a threshold on the speed of $\mathbf{x}$ in $\Delta^{|S|}$ or a maximum number of iterations can be set. After a stable state has been reached all the matching strategies that thrive under the replicator dynamics should be retained. Since no strategy will be completely extinct (as the dynamics do not allow to reach the faces of the simplex) another threshold is needed to select the strategies that succeeded. This is not done through an absolute threshold, but rather by fixing a ratio over the maximum value in the population vector. In the experimental section we will show that the value of this threshold has little influence over the quality of the registration obtained. Further, we can weight the contribution of the surviving matches according to the corresponding value in $\mathbf{x}$. This has the consequence of reducing the impact of relatively low-quality points and was shown to allow a slight enhancement in [34]. In Fig. 3, a complete example of the process is shown. While the example is kept simple on purpose and the data does not come from real surfaces the illustrated evolution is computed exactly with the payoff matrix Π using Eq. (1).

Since the proposed framework relies on geometric consistency, we choose to adopt a very loose feature descriptor that enhances the probability for a feature point to be repeatable, albeit allowing a much higher number of outliers to get into the set of initially proposed matches.

It is interesting to note that, in order to avoid feature points that carry little useful information for registration purposes (such as flat areas or regions of constant curvature), a minimal matching game will be run also between model points.

3.4. Surface hashes

Intuitively, a Surface Hash is a concise point feature descriptor which exhibits the property of being highly repeatable at the cost of a relatively high probability of clashing. In practice this happens with any low-dimensional descriptor, such as the Gaussian or Mean Curvature (1 dimension), the first two Principal Components of a patch (2 dimensions), or the normal vector associated to a point (2 dimensions). While those descriptors could be used with our registration pipeline, we prefer to introduce two multi-scale Surface Hashes based respectively on the dot product between normals and a local surface integral. Each of our descriptors corresponds to a vector of scalar measures evaluated at different scales. By increasing or reducing the number of scales, we are able to obtain vectors of different length, thus being more or less distinctive.

The *Normal Hash* (see Fig. 4(a)) is obtained by setting as a reference the average surface normal over a patch that extends to the largest scale (red arrow in figure) and then, for each smaller scale, calculate

the dot product between the reference and the average normal over the reduced patches (blue arrows in figure). This measure finds its rationale in the observation that at the largest scale the average normal is more stable with respect to noise and that the dot product offers a concise representation of the relation between the vectors obtained at various scales.

The *Integral Hash* (see Fig. 4(b)) is similar in spirit to the Normal Hash. In this case we search for the best fitting plane (in the least squares sense) with respect to the surface patch associated to the largest scale. Then we calculate the volume enclosed between the surface and such a plane. In practice, it is not necessary to evaluate this volume accurately: even naive approximations, such as the sum of the distances of the surface points from the plane, have shown to provide a reasonable approximation in all the empirical tests. Note that Normal Hashes evaluated over n scales yield descriptor vectors of length $n-1$ (since the larger scale is used only to calculate the reference normal), while Integral Hashes provide n-dimensional vectors. In Fig. 5 a Normal Hash of dimension 3 (respectively from (a) to (c)) evaluated over 4 scales is shown. Note that the descriptor is not defined on the points for which the larger support is not fully contained in the surface, i.e., points close to the surface boundary.

4. Matching using local features

4.1. Interest points selection

Given the large number of points contained in typical 3D objects, it is not practical for any matching algorithm to deal with all of them. In addition, the isolation of a relatively small number of interest points can enhance dramatically the ability of the matcher to avoid false correspondences, usually due to a large number of features with very common characterizations. This is particularly true when using Surface Hashes, which are loosely distinctive by design. Paradoxically, we use exactly this property to screen out features exhibiting descriptors that are too common over the surface. This happens by defining a Matching Game where the strategy set S corresponds to the set of all the surface points and the payoff matrix is defined by $\pi_{ij} = e^{-\alpha|d_i - d_j|}$, where d_i and d_j are the descriptor vectors associated to surface point i and j, and α is a parameter that controls the level of selectivity. Clearly, features that are similar in terms of Surface Hashes will get a large mutual payoff and are thus more likely to be selected by the evolutive process. In this sense, our goal is to let the population evolve to an ESS and then remove from the set of interest points the features that survived the evolutive process. At the beginning we can initialize the set of retained features to the whole surface and run a sequence of Matching Games until the desired number of points are left. At this point, the remaining features are those characterized by less-common descriptors which are more likely to represent good cues for the matching. It should be noted that by choosing large values for α the payoff function decreases more rapidly with the growth of the distance between the Surface Hashes, thus the Matching Game becomes more selective and less points survive after reaching an ESS. In the end this results in a blander decimation and thus in a larger ratio of retained interest points. By converse, a small value for α leads to a more greedy filtering and thus to a more selective interest point detector. In Fig. 5 (from (d) to (f)) we show three steps of the evolutive interest point selection with respect to the 3-dimensional Normal Hash shown from (a) to (c). In Fig. 5(d) we see that after a single pass of the Matching Game most of the surface points are still considered interesting, while after respectively two and three passes only very distinctive points (belonging to areas with less common curvature profile) are left.

5. Isometry-enforcing payoff

As already stated, for this particular application of the GTM framework we decided to assign to each pair of matching strategies a payoff that is inversely proportional to a measure of violation of the surface rigidity constraint. This violation can be expressed in
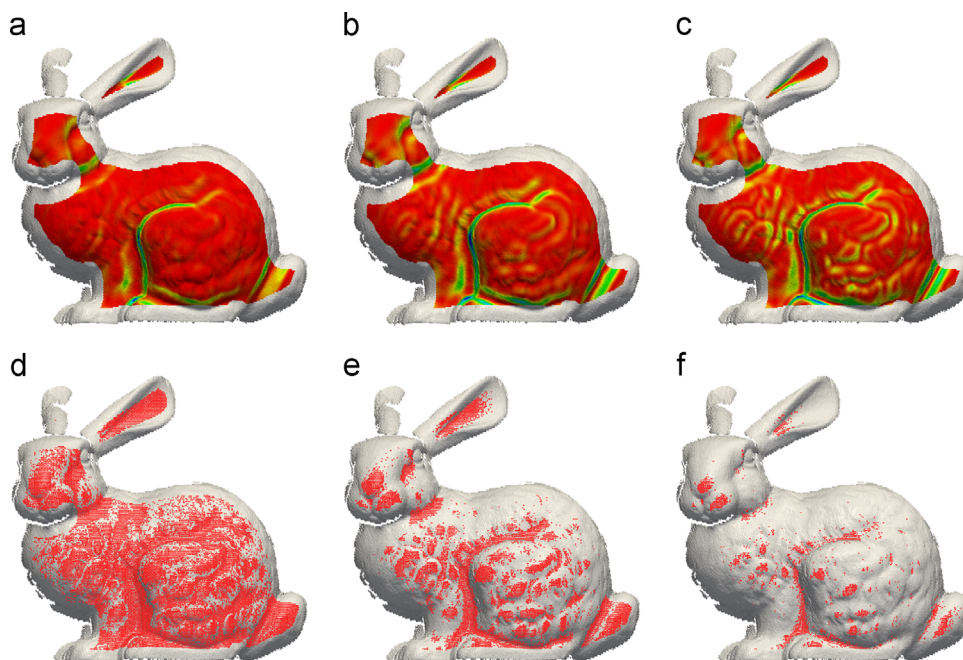

Fig. 5. Example of a 3-dimensional Normal Hash and the related detection process. In the first row we show, for each column, the value of the nth component of the computed Normal Hash using a JET color scale normalized with respect to the higher value. In the second row we used red dots to highlight the surface points that have been selected using a game theoretical matcher as described in Section 4.1. The first column shows the points selected at the first pass of the game and the other two the remaining culled points for the second and third pass of the same game. (a) First dimension; (b) second dimension; (c) third dimension; (d) first pass; (e) second pass; (f) third pass. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

several ways, but since all the rigid transformations preserve Euclidean distances, we choose this property to express the coherence between matching strategies. Clearly this isometry constraint is looser than the rigidity constraint as it cannot prevent specular flips of the surfaces, but the global consistency provided by the game-theoretic framework ensures that only rigid alignments will prevail.

Definition 1. Given a function $\pi : S \times S \to \mathbb{R}^+$, we call it an *isometry-enforcing payoff function* if for any $((a_1, a_2), (b_1, b_2))$ and $((c_1, c_2), (d_1, d_2)) \in S \times S$ we have that $\|a_1 - b_1| - |a_2 - b_2\| > \|c_1 - d_1| - |c_2 - d_2\|$ implies $\pi((a_1, a_2), (b_1, b_2)) < \pi((c_1, c_2), (d_1, d_2))$. In addition, if $\pi((a_1, a_2), (b_1, b_2)) = \pi((b_1, b_2), (a_1, a_2))$, π is said to be *symmetric*.

An isometry-enforcing payoff function is a function that is monotonically decreasing with the absolute difference of the Euclidean distances between respectively the model and data points of the matching strategies compared. In other words, given two matching strategies, their payoff should be high if the distance between the model points is equal to the distance between the data points and it should decrease as the difference between such distances increases. In the example of Fig. 2, matching strategies (a_1, a_2) and (b_1, b_2) are coherent with respect to the rigidity constraint, whereas (b_1, b_2) and (c_1, c_2) are not, thus it is expected that $\pi((a_1, a_2), (b_1, b_2)) > \pi((b_1, b_2), (c_1, c_2))$.

Further, if we want mating to be one-to-one, we must put an additional constraint on the payoffs, namely that mates sharing a point are incompatible.

Definition 2. An isometry-enforcing payoff function π is said to be *one-to-one* if $a_1 = b_1$ or $a_2 = b_2$ implies $\pi((a_1, a_2), (b_1, b_2)) = 0$.

Given a set of matching strategies S and an enumeration $O = \{1, \ldots, |S|\}$ over it, a *matching game* is a non-cooperative game where the population is defined as a vector $\mathbf{x} \in \Delta^{|S|}$ and the payoff matrix $\Pi = (\pi_{ij})$ is defined as $\pi_{ij} = \pi(s_i, s_j)$, where $s_i, s_j \in S$ are enumerated by O and π is a symmetric one-to-one isometry-enforcing payoff function. Intuitively, $\mathbf{x}_i$ accounts for the percentage of the population that plays the ith matching strategy. By using a symmetric one-to-one payoff function in a matching game we are guaranteed that ESS's will not include mates sharing either model or data nodes. In fact, given a non-negative payoff function, a stable state cannot have in its support pairs of strategies with payoff 0 [34]. Moreover, a matching game exhibits some additional interesting properties.

Theorem 1. *Given a set of model points M, a set of data points $D = TM$ that are exact rigid transformations of the points in M, a set of matching strategies $S \subseteq M \times D$ with $(m, Tm) \in S$ for all $m \in M$, and a matching game over them with a payoff function π, the vector $\hat{\mathbf{x}} \in \Delta^{|S|}$ defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M; \\ 0 & \text{otherwise,} \end{cases}$$

is an ESS and obtains the global maximum average payoff.

Proof. Let $\hat{S} \subseteq S$ be the set of mates that match a point to its copy, clearly for all $s, q \in \hat{S}, s \neq q$ we have $\pi(s, q) = 1$, while for $s \in \hat{S}$ and $q \in S \backslash \hat{S}$, we have $\pi(s, q) < 1$. For all $s \in \hat{S}$ we have that $\pi(\hat{\mathbf{x}}, \hat{\mathbf{x}}) = (|M| - 1)/|M|$ while, since π is one-to-one, for any $q \in S \backslash \hat{S}$ there must be at least one $s_q \in \hat{S}$ with $\pi(q, s_q) = 0$, thus $\pi(q, \hat{\mathbf{x}}) < (|M| - 1)/|M|$, thus $\hat{\mathbf{x}}$ is a Nash equilibrium. Further, since the inequality is strict, it is an ESS. Finally, $\hat{\mathbf{x}}$ is a global maximizer of π since $(|M| - 1)/|M|$ is the maximum value that a one-to-one normalized payoff function over $|M|$ points can attain. □

This theorem states that when matching a surface with a rigidly transformed copy of itself the optimal solution (i.e., the population configuration that selects all the matching strategies

assigning each point to its copy) is the stable state of maximum payoff. Since well established algorithms to evolve a population to such a state exist, this provides us with an effective mating approach. Clearly, aligning a surface to an identical copy is not very useful in practical scenarios, where occlusion and measurement noise come into play. While the quality of the solution in the presence of noise will be assessed experimentally, we can give some theoretical results regarding occlusions.

Theorem 2. *Let M be a set of points with $M_a \subseteq M$ and $D = TM_b$ a rigid transformation of $M_b \subseteq M$ such that $|M_a \cap M_b| \geq 3$, and $S \subseteq M_a \times D$ be a set of matching strategies over M_a and D with $(m, Tm) \in S$ for all $m \in M_a \cap M_b$. Further, assume that the points that are not in the overlap, that is the points in $E_a = M_a \backslash (M_a \cap M_b)$ and $E_b = M_b \backslash (M_a \cap M_b)$, are sufficiently far away such that for every $s \in S, s = (m, Tm)$ with $m \in M_a \cap M_b$ and every $q \in S, q = (m_a, Tm_b)$ with $m_a \in E_a$ and $m_b \in E_b$, we have $\pi(q, s) < (|M_a \cap M_b| - 1)/|M_a \cap M_b|$, then, the vector $\hat{\mathbf{x}} \in \Delta^{|S|}$ defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M_a \cap M_b; \\ 0 & \text{otherwise,} \end{cases}$$

is an ESS.

Proof. We have $\pi(\hat{\mathbf{x}}, \hat{\mathbf{x}}) = (|M_a \cap M_b| - 1)/|M_a \cap M_b|$. Let $q \in S$ be a strategy not in the support of $\hat{\mathbf{x}}$, then, either it maps a point in M_a or M_b, thus receiving payoff $\pi(q, \hat{\mathbf{x}}) < (|M_a \cap M_b| - 1)/|M_a \cap M_b|$ because of the one-to-one condition, or it maps a point in E_a to a point in E_b, receiving, by hypothesis, a payoff $\pi(q, \hat{\mathbf{x}}) < (|M_a \cap M_b| - 1)/|M_a \cap M_b|$. Hence, $\hat{\mathbf{x}}$ is an ESS. □

The result of Theorem 2 is slightly weaker than Theorem 1, as the face of the simplex corresponding to the "correct" overlap, while being an evolutionary stable state, is not guaranteed to obtain the overall highest average payoff. This is not a limitation of the framework as this weakening is actually due to the very nature of the alignment problem itself. The inability to guarantee the maximality of the average payoff is due to the fact that the original object (M) could contain large areas outside the overlapping subset that are perfectly identical. Further, objects that are able to slide (for instance a plane or a sphere) could allow to move between different mixed strategies without penalty. These situations cannot be addressed by any algorithm without relying on supplementary information. However, in practice, they are quite unlikely, exceptional cases. In the experimental section we will show that our approach can effectively register a wide range of surface types.

In theory, any rigidity-enforcing payoff function can be used to perform surface registration. Throughout the experimental section we adopted:

$$\pi((a_1, b_1), (a_2, b_2)) = \left(\frac{\min(|a_1 - a_2|, |b_1 - b_2|)}{\max(|a_1 - a_2|, |b_1 - b_2|)} \right)^\lambda \tag{2}$$

where a_1, a_2, b_1 and b_2 are respectively the two model (source) and data (destination) points in the compared matching strategies. Parameter λ allows to make the enforcement of the Euclidean distance more or less strict.

In Fig. 6 we show a complete example of the evolutionary matching process. In order to make the example easy to understand we restricted our focus to a detail of a range scan of the Stanford dragon [43]. In this example (and throughout all the experimental section) S is built by including all the strategy pairs composed by a feature point in the model and the 5 nearest feature points in the data in terms of Surface Hash (in this example we used an Integral Hash with 3 scales). In Fig. 6(g) we show, on a color scale from 0 to 1, the payoff matrix of the rigidity enforcing function (2). Note that in the diagonal area of the matrix blocks of five strategies with reciprocal 0 payoff can be found: this is related to the way we built S. In fact we
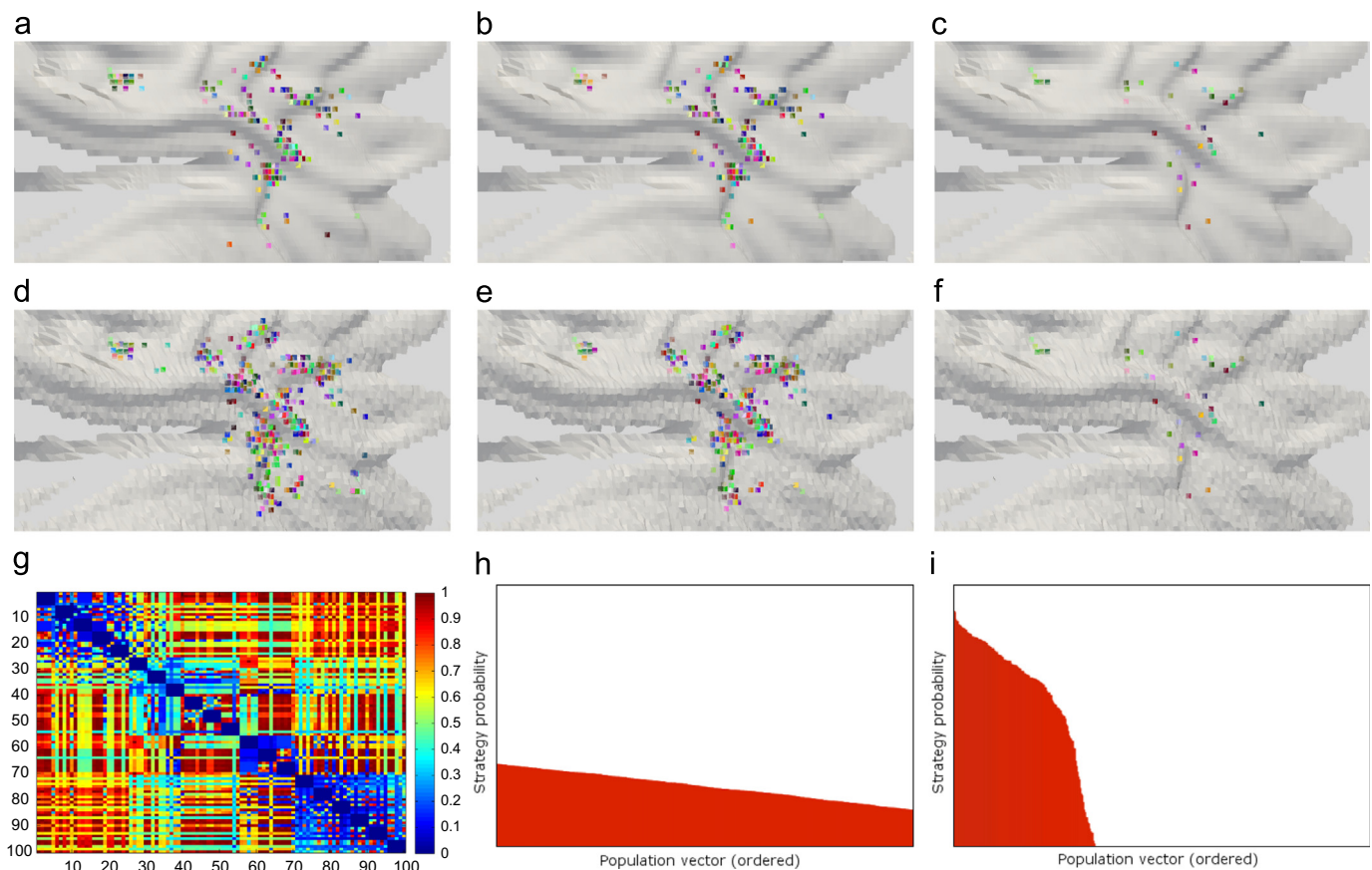
Fig. 6. Example of a rigidity enforcing payoff and of the evolution of the matching process. The upper part of the figure shows color-coded matching points between model and data before the game, after 1 round of replicator dynamics and after 100 rounds (see the text for details). The last row shows respectively the payoff matrix (using a colored JET scale) and the (ordered) population vector after 1 and 100 iterations of the replicator dynamics. Notice that upon convergence most probability density is concentrated in few strategies: (a and d) Initial matches; (b and e) matches in 1 round; (c and f) matches in 100 rounds; (g) payoff matrix; (h) population in 1 round; (i) population in 100 rounds.

chose to include for each model point 5 candidates in the data which are mutually noncompatible as they share the same source point and we are looking for a one-to-one match. In the first and second row of Fig. 6(d) we can see respectively model and data feature points at the beginning of the matching process. After just one round of replicator dynamics we see that many outliers have been eliminated from the initial set S, but still some wrong matches are present. After 100 iterations only a few matches are retained, but it is easy to see that they are extremely coherent. Finally, in Fig. 6(h) and (i) we show the (sorted) population histogram respectively after 1 and 100 iterations. The first histogram shows that all the strategies are still played by a sizeable amount of the population, while after 100 iterations most of the consensus is held by the few surviving matches.

6. Experimental results

In this paper we introduce a Game-Theoretic Registration approach (GTR) that is based both on a feature detector/descriptor and on a matching technique. To better explore the role of both, we designed a wide range of experimental validations. First, we analyzed the sensitivity of the descriptor to several sources of noise and the influence of the number of scales (and thus of the size of the descriptor vector). Further, we studied the sensitivity of the matching algorithm to its parameters, with the goal of identifying an optimal parametrization (if any) and assess the stability of the method. A number of comparative tests were also made. Specifically, we analyzed the performance obtained by using our matcher with

different feature detectors and the overall comparison with respect to other well-known registration pipelines.

All the experiments were performed on a modern personal computer equipped with a Core i7 Intel processor and 8 GB of memory. The dataset used, where not differently stated, was built upon publicly available models; specifically the Bunny [41], the Armadillo [42] and the Dragon [43] from the Stanford 3D scanning repository. To further assess the shortcomings of the various approaches, we used two synthetic surfaces representative of as many difficult classes of objects: a wave surface and a fractal landscape (see Fig. 7). Since a ground truth was needed for an accurate quantitative comparison, we generated virtual range images from the models and then applied additive Gaussian noise to them. All the registration experiments adopted the payoff function (2) with the additional one-to-one constraint.

6.1. Implementation details

Several practical details should be defined and some parameters must be fixed to produce an actual implementation of the registration method presented. Most of such details are explored in the following experimental evaluation, where we study the role of parameters and the general feasibility of our approach. However, before describing in detail the tests performed and the results obtained, we are going to give an overview of the steps involved and an in-depth analysis of the role and behavior of each step. The scope of such coverage is two-fold: it is useful in order to better
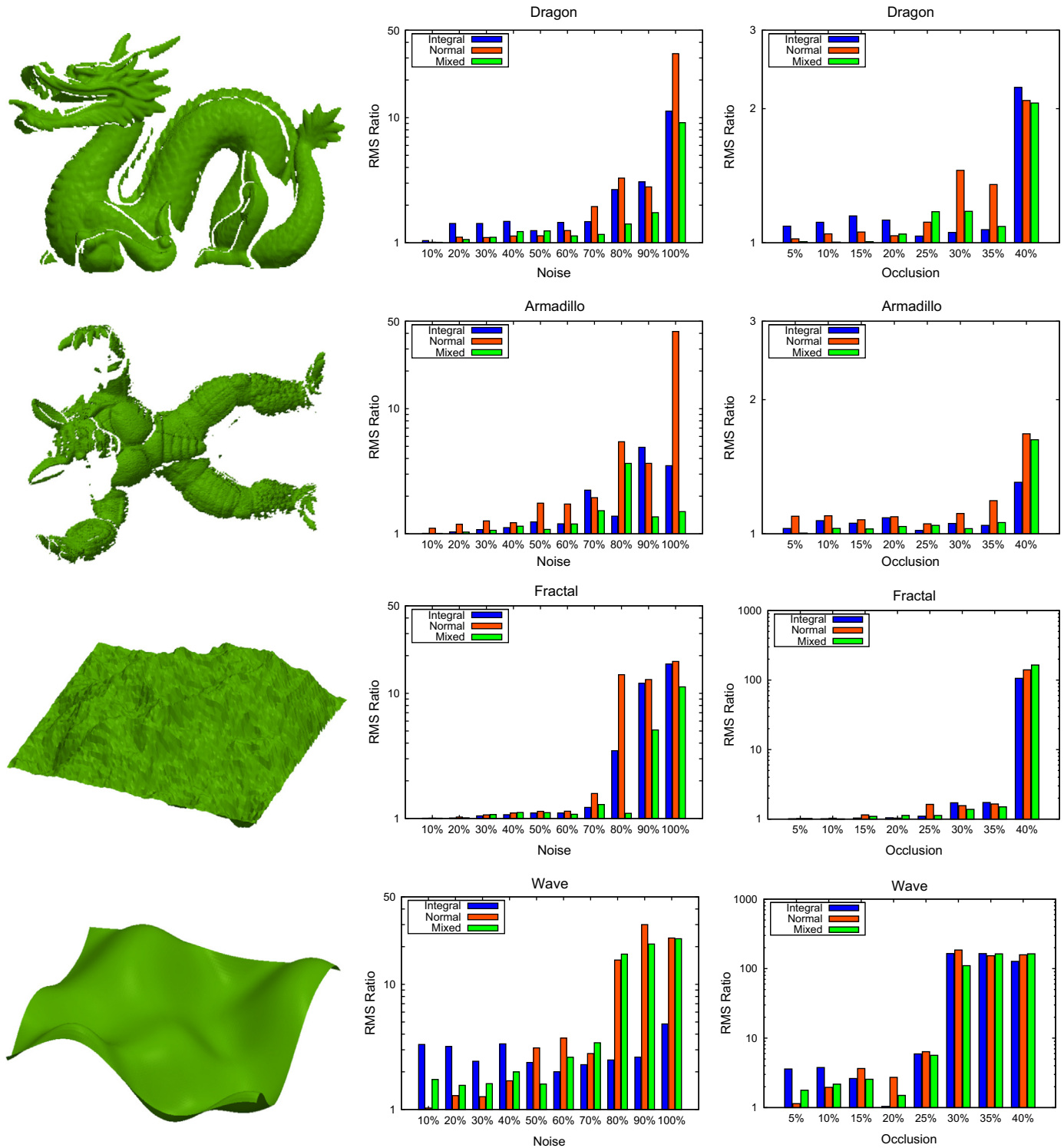
Fig. 7. Performance comparison of different descriptors using real and synthetic objects under different amounts of noise and occlusion. The error is expressed as a ratio between the obtained RMS and the RMS of the ground-truth alignment. Noise is expressed as percentage of the standard deviation of Gaussian additive noise over the average edge length. Occlusion is expressed as percentage of occluded surface area.

understand the experimental evaluation, and it can serve as an effective implementation guide.

The general method can be broken into a sequence of individual steps, each with a specific goal:

- Selection of the relevant points to be matched.
- Creation of the initial candidate matches between relevant points.
- Computation of a payoff matrix between candidates.
- Evolution of an initial population towards a stable state.
- Surface alignment by means of the surviving strategies.

The implementation details for each step, according to the configuration we used for our evaluation, are given in the following.

Selection of relevant points: In principle the evolutionary selection process could be performed between all the possible pairs of

source and destination points between the data and model surfaces. Unfortunately, this would result in a huge number of candidates and an even bigger payoff matrix (give that its size grows with the square of the candidates). For this reason, it is important to operate a subsampling over surface points. Of course, different subsamplings will result in different performance levels, depending on the quality of the sampled points with respect to the registration task [47]. In this paper we introduce the Surface Hashes descriptor that can be coupled with a game-theoretical selector in order to operate subsampling using the approach described in Section 4.1. Specifically, in our experiments we set the selectivity parameter α to 1 and, after evolving an initially random population to a stable state, we selected the k points exhibiting lower densities in the final population vector, i.e. the k points that have less common descriptors. In Section 6.3 we analyze the behavior of the method for different amounts of sampled points and we conclude that as few as 1000 samples are enough to obtain a satisfying performance level. In Section 6.2 we study the effect of using different Surface Hashes flavors, and we conclude that Mixed Hashes performs better, hence being the recommended choice for implementation. Of course, nothing prevents to adopt completely different subsampling schemes and point descriptors that can be dropped-in as a substitute for Surface Hashes within this first step. In Section 6.5 we do exactly this, by studying what happens by using alternative descriptors, leaving all the other aspects of the pipeline untouched.

Creation of the initial candidate set: After selecting relevant points from each surface, a set of putative matches (i.e. strategies in our selection game) must be created. Again, in principle, we could make this set the Cartesian product between the points just sampled from source and data surfaces. Still, it would be more reasonable to perform an initial selection of the matches that are most likely to be correct. To this end, we took the most naive approach: each model point generates a candidate match with the n data points with the most similar descriptor vector. Clearly, n should be large enough in order to include most of the correct matches, at the same time it should be kept reasonably small to avoid the excessive growth of the payoff matrix. In Section 6.3 we investigate the optimal value for n and we find that setting it to 6 gives good results that are not significantly improved by raising it further. Thus, we recommend to use this value in practice.

Computation of the payoff matrix: Once the candidates have been created, the compatibility between each pair of them must be computed and used to populate the payoff matrix that will drive the evolutionary process. This is done by means of Eq. (2), which depends on parameter λ. As already described such parameter can be modulated to obtain different levels of selectivity. The optimal value for λ, which we use and recommend, has been determined to be about 1 through an apt experiment described within Section 6.3.

Evolution toward a stable state: The game-theoretical selection process is performed by evolving an initial population through Eq. (1). For an actual implementation to be feasible, two practical choices are to be made. The first one is the initial state of the population. In practice we observed that a well-performing initial position for the population vector is a slightly perturbed barycenter, that we obtain by adding 5% random noise to each element of a uniform distribution which is subsequently normalized again. The second choice to be made is when to stop the iterations of Eq. (1). In fact, reaching a truly stable state would require an infinite number of iterations, but in practice a near enough population configuration is reached quite soon. To this end, in our implementation we considered a stable state to be obtained when the higher change among all the strategies in the population vector is below a given threshold, that we fixed at 0.1% for all the experiments.

Extraction of surviving strategies and surface alignment: The actual registration takes place using Horn's quaternion-based

method [20] between the model and data points available in the surviving strategies. However, since a true stable state has not been reached, in practice all the strategies exhibit non-zero values in the population vector. For this reason, a criterion to tell a surviving strategy from an extinguished one must be chosen. In our implementation we set an adaptive threshold expressed as a fraction of the maximum value in the final population. As described in the experimental evaluation of Section 6.3, any value below 0.8 seems to be suitable for a proper culling.

6.2. Sensitivity analysis of the descriptor

The performance of different descriptors was tested for various levels of noise and occlusion applied to two surfaces obtained from real range scans ("armadillo" and "dragon" from Stanford) and two synthetic surfaces designed to be challenging for coarse and fine registration techniques ("fractal" and "wave").

The noise is a positional Gaussian perturbation on the point coordinates with its level (σ) expressed in terms of the percentage of the average edge length, while occlusion denotes the percentage of data and model surfaces removed. Notice that expressing the noise as a percentage over the average edge length makes sense for several reasons. First, our method is scale independent since replicator dynamics will yield the same solution for any multiplicative constant applied to the payoff matrix. For this reason, using a percentage rather than an absolute value allows to make the comparison fair among objects of different size in the adopted database. Moreover, the average edge length, especially for scanned data, can be regarded as a measure of density of the signal, thus adding a noise level related to it is a more fair measure of the noise-to-signal ratio. The net effect arising from the combination of the scale independence of our approach and the definition of an error level proportional to the average edge length is indeed a sort of normalization of the models in the database with respect to the edge length.

The occlusion is measured as a percentage of the surface and is applied to the range images in opposite directions of the overlapped area. That means that with an occlusion of 10% the actual overlap is reduced by 20%. Notice that, since we are applying occlusions up to 40%, in the more extreme tested conditions the overlap is very low.

The RMS Ratio in the charts is the ratio of the root mean square error (RMS) obtained after registration and the RMS of ground truth alignment.

The Normal and Integral Hashes were calculated over 3 levels of scale and the "Mixed" Hash is simply the concatenation of the previous two.

In Fig. 7 we see that all the descriptors obtain good results with real range images and the registration "breaks" only with very high levels of noise (on the same order of magnitude of the edge length). It is interesting to observe that the Mixed Hash always obtains the best performance, even with high level of noise: This higher robustness is probably due to the orthogonality between the Normal and Integral Hashes. The behavior with the "fractal" synthetic surface is quite similar, by contrast all the descriptors seem to perform less well with the "wave" surface. This is due to the lack of distinctive features on the model itself, which indeed represents a challenge for any feature based registration technique [7]. The performance obtained with respect to occlusion is similar: all the descriptors achieve fairly good results and are resilient to high levels of occlusion (note that 40 percent occlusion is applied both to data and model). Overall the Mixed Hash appears to be consistently more robust. Since we found that the descriptors calculated over 3 levels of scale break at a certain level of noise, we were interested in evaluating if their performance can be improved by increasing their dimension.
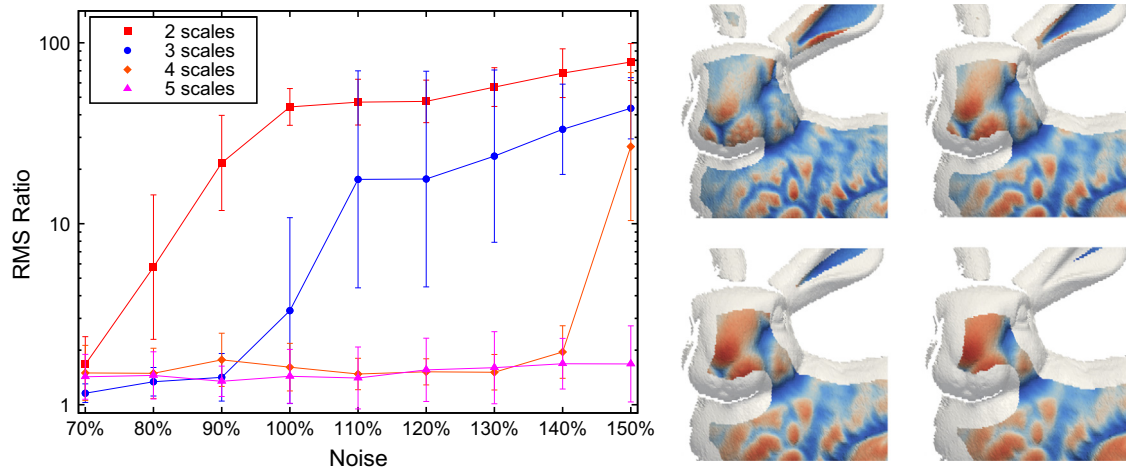
Fig. 8. Effect of scale on the matching accuracy computed over all the 5 test surfaces (plot on the left). The error is expressed as a ratio between the obtained RMS and the RMS of the ground-truth alignment. Noise is expressed as percentage of the standard deviation of Gaussian additive noise over the average edge length. Scalar fields for scale levels from 2 to 5 presented as a color map over the Bunny model (on the right).

In Fig. 8 we present the results obtained with different levels of scale for the Mixed Hash. The graphs show the average over all the surfaces and the associated RMS. It is interesting to observe that by reducing the scale level the technique becomes less robust, whereas its performance increases dramatically when the number of scales increases. With a scale level of 5 our approach can deal even with surfaces subject to Gaussian positional noise of σ greater than the edge length. Unfortunately, this enhanced reliability comes with a drawback: by using larger levels of scale the portion of boundary that cannot be characterized grows. In the right half of Fig. 8 the shrinking effect is shown for scale levels from 2 to 5.

6.3. Sensitivity to the parameters of the matcher

The game-theoretic matching technique presented basically depends on four parameters:

- The number of points sampled from the model object.
- The number of neighbors considered when building the initial set of candidates.
- The selectivity λ for the rigidity-enforcing payoff (2).
- The quality threshold used to deem a strategy as non-extinct upon convergence.

The first two parameters are related to the creation of the set of strategies S. From a performance point of view, adopting a strong subsampling will produce a smaller set of strategies and a smaller payoff matrix, thus each occurrence of the replicator dynamics (1) will be faster. However, a too sparse sampling can lead to groups of mutually compatible matches that are too small and are unable to thrive during the evolutionary process. In a similar manner, a small number of neighbors will reduce the number of strategies. However, this will give fewer chances of capturing the correct pairings since such tightening would require the descriptor to always give a high rank to the correct pairing. In Fig. 9 it can be seen that optimal results can be achieved with less than 1000 samples and that there is virtually no gain in using more than 6 neighbors. Later in this chapter we will show that on the test system used these conditions allow to perform an alignment in less than one second.

The third parameter (λ) is related to the level of strictness with respect to the enforcement of the rigidity constraint: Higher values for λ will make the payoff function more steep, thus making the selection process more picky. By contrast, lowering λ will yield a payoff matrix with smaller variance, up to the limit value of 0,

when the matrix assumes value 1.0 for all the strategies pairs that do not break the one-to-one constraint and 0 otherwise. As expected, our experiments show that very low or very high values for λ deliver poor results and, while there is clearly a larger variance that what has been captured by the experiments, the optimal value seems to be around 1.

Finally, the fourth parameter sets the ratio (with respect to the most successful match) used to classify a strategy as surviving or extinct. The last experiment of Fig. 9 shows that all the tested values below 0.8 give similarly good results. This simply means that there is good separability between extinct and non-extinct strategies, the former being very close to 0.

Overall we can assess that the matching method has a very limited dependency on its parameters, which can easily be fixed at values that are both safe and efficient. The most influential parameter is probably λ, however a value of 1.0 (that indeed simplifies equation (2) to a simple ratio) appears to be optimal for our test set.

6.4. Comparison with full pipelines

The whole registration algorithm introduced with this paper can be classified as a coarse method, since it does not require initialization. For this reason we compared it with several other coarse techniques. Specifically, we implemented the whole Spin Images pipeline [29], the RANSAC-Based and PROSAC-Based Darces methods [16], and we adopted the implementation supplied by the authors respectively for the MeshHOG/MeshDOG [31] and the Four Points Congruent Sets [17] methods. The latter method was initialized both with the parameters suggested by the authors and also with values for δ (agreement threshold) and s (number of samples) that we manually optimized to get the best possible results from our dataset. In addition, we also included in the comparison the scale-invariant game theoretical method proposed in [36] as a technique for object recognition in cluttered scenes.

In the first row of Fig. 10 we present the results of this comparison. The noise is the usual additive Gaussian noise with a standard deviation expressed as a percentage over the average edge length and the occlusion is measured with respect to each range image and is applied in opposite directions of the overlapped area. The occlusion test has been made with noise at level 10% and the noise test was performed with no occlusion. From the tests our method exhibits better results in both scenarios and breaks only with high levels of occlusion and noise. Note that the 4PCS method with parameters $\delta = 0.9$ and $s = 500$ does not always give a feasible solution with any occlusion greater than 10%. With extreme levels of noise the 4PCS
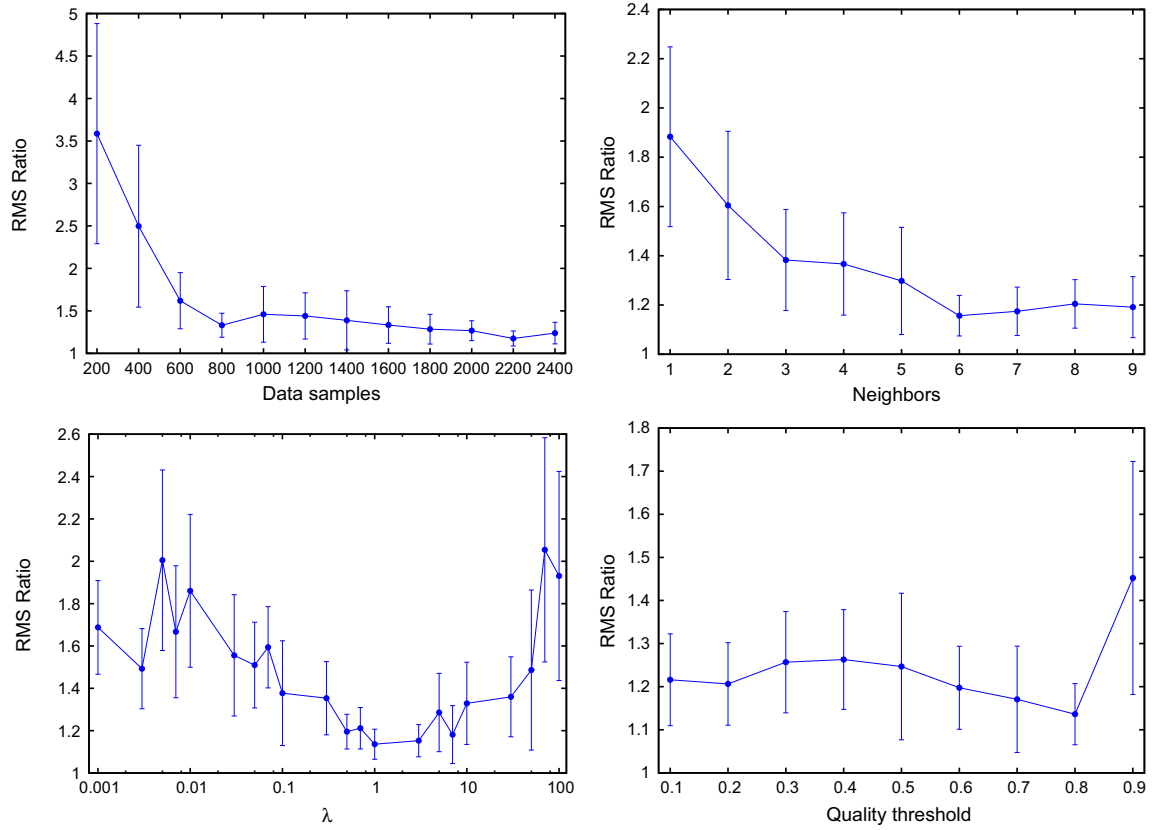
Fig. 9. Analysis of the sensitivity of the Game-Theoretic Matcher with respect to the parameters of the algorithm. The test has been performed using all the 5 test surfaces.
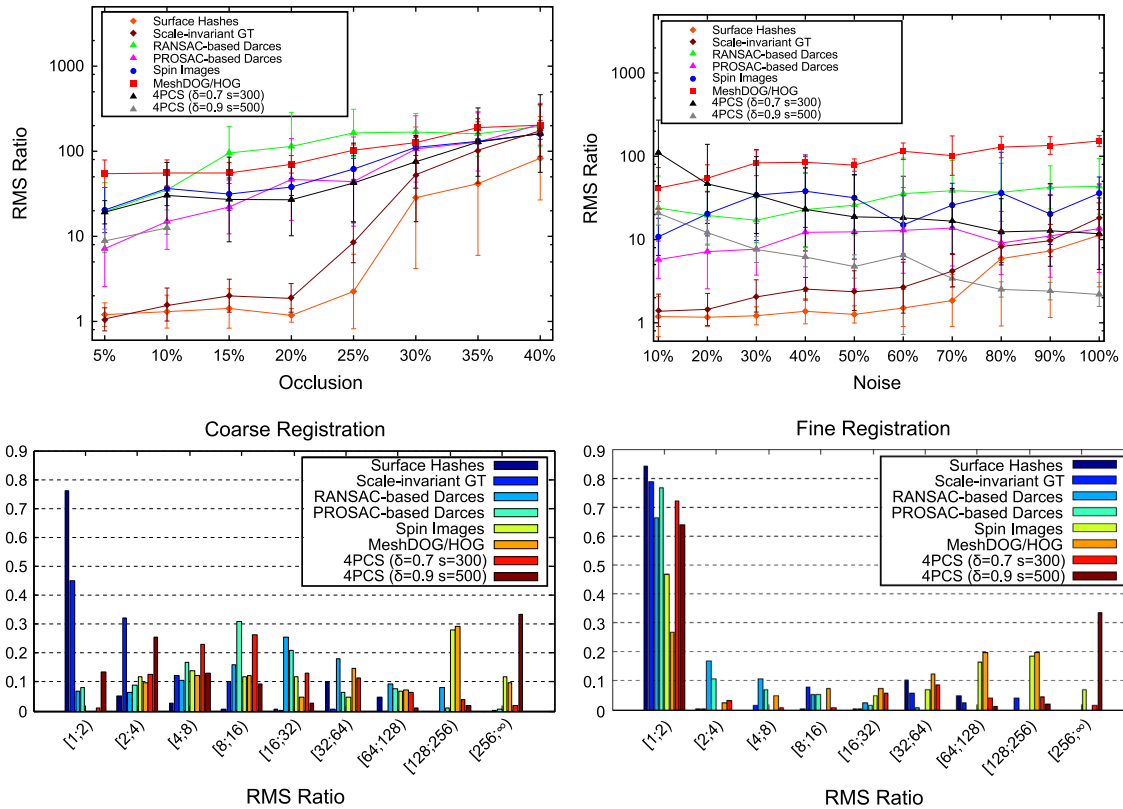

Fig. 10. Comparisons between our Game-Theoretic Registration technique and other widely used surface registration pipelines.

seems to get better and obtains lower RMS ratios than our method. The reduction in performance of our method is related to the breaking of the descriptors, that at such high levels of noise do not carry sufficient information any more. A clarification should finally be made about the apparent improvement that 4PCS seems to exhibit as noise increases. In fact, at high noise levels the RMS associated to ground-truth motion is also high. In such conditions the additional error due to misalignment becomes less relevant in terms of contribution to the overall RMS ratio, which is dominated by random noise. Since 4PCS explores thoroughly the set of feasible motions until a solution with RMS low enough is found (depending on the stopping criterion), it is expected to test more alignments when surfaces are noisier and thus yield lower RMS ratio values. However, it is easy to build simple examples where a solution can obtain a low RMS ratio (even lower than one) and still be far from the correct alignment. Finally, we also note that this newly introduced method consistently outperforms the method presented in [36]. This is due to the task-specific optimizations and design choices described in Section 2.

These results only indicate that GTR gives a better coarse registration, however to seek a perfectly fair comparison it is also needed to measure how much enhancement can be obtained by performing a fine registration step starting from the obtained coarse initialization. To this end we applied the ICP algorithm starting from the initial motion estimated with the different methods with no occlusion and random noise values below 60%. The results are shown in the bottom row of Fig. 10 with histograms obtained by binning the distance between model points and data surface along the normal vector. Normals that do not intersect the data surface are discarded. The size of the bins grows exponentially. The first histogram shows the distribution obtained from the coarse registration and the second reports the enhancement obtained by applying ICP. Again, the results are favorable to our method, with very few points exhibiting large errors after refinement.

6.5. Influence of different feature descriptors

In principle, there is nothing that binds the proposed method to the Surface Hashes descriptors. Actually, the game-theoretic step does not use the descriptor at all and any other interest point characterization could be used as a drop-in replacement in order to build the initial matching strategies. To show the generality of the technique and to investigate the robustness of GTR we swapped the Surface Hashes descriptor with a dense variation of Spin Images [44] and the more recent SHOT 3D feature [32].

In Fig. 11 the results obtained performing the same experiments designed to compare different methods are shown. It is apparent that Spin Images do not work very well with our method. By contrast, the SHOT descriptor behaves well with respect to high

noise levels and is even a little more tolerant to occlusion than Surface Hashes. This is mainly due to the large support needed by the latter. On the other hand, Surface Hashes offer better alignment under noise, and are also much faster to compute.

6.6. Quality of fine registration

In addition to the full pipeline comparisons we also investigated how reliable the proposed approach would be if directly used as a fine registration technique. The goal of this test is two-fold: we want to evaluate our quality as a complete alignment tool and, at the same time, find the breaking point of traditional fine registration techniques.

The method we used for comparison is a best-of-breed ICP variant, similar to the one proposed in [41]. Point selection is based on Normal Space Sampling [7], and point-surface normal shooting is adopted for finding correspondences; distant mates or candidates with back-facing normals are rejected. To minimize the influence of incorrect normal estimates, matings established on the boundary of the mesh are also removed. The resulting pairings are weighted with a coefficient based on compatibility of normals, and finally a 5%-trimming is used. Each test was performed by applying a random rotation and translation to different range images selected from the Stanford 3D scanning repository. Additionally, each range image was perturbed with a constant level of Gaussian noise with standard deviation equal to 12% of the average edge length. We completed 100 independent tests and for each of them we measured the initial RMS error between the ground-truth corresponding points and the resulting error after performing a full round of ICP (ICP) and a single run of our registration method (GTR). In addition, we applied a step of ICP to the registration obtained with our method (GTR + ICP) in order to assess how much the solution extracted using our approach was further refinable.

A scatter plot of the obtained errors before and after registration is shown in Fig. 12. The final error is on a log scale, so the dotted curve represents the points with identical initial and final error. We observe that ICP reaches its breaking point quite early; in fact with an initial error above the threshold of about 20 mm it is unable to find a correct registration. By contrast, GTR is able to obtain excellent alignment regardless of the initial motion perturbation. Finally, applying ICP to GTR decreases the RMS only by a very small amount.

6.7. Sensitivity to sampling

One of the main applications of the proposed method is the registration of range images acquired using 3D scanners. Within this context, both model and data surfaces will exhibit a similar sampling density, which is usually quite high even when using
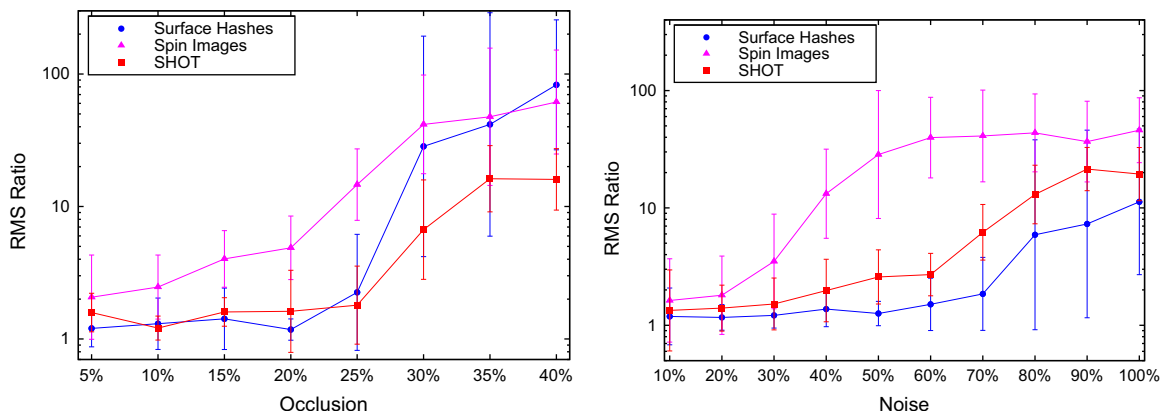

Fig. 11. Performance of the Game-Theoretic Registration method using different feature descriptors.

consume-grade off-the-shelf 3D capturing devices. Still, it is possible to describe usage scenarios where the model contains a very low number of vertices. This could be the case, for instance, if the registration between an acquired object and the corresponding CAD model is sought in order to measure the quality of a manufactured part. In fact, CAD models (or mechanical parts in general) are usually characterized by large regular surfaces and few key vertices. Under these conditions, a direct application of our method would fail, since not enough correct matching candidates could be established between the dense representation of the captured surface and the sparse model.

A quite straightforward solution is to resample the sparse object obtaining additional vertices distributed over its surface. To evaluate the effect of the introduction of artificial vertices and to study how many should be added for a satisfying alignment we designed a specific experiment. We selected 6 models of technical objects, 3 of them being synthetic CAD models and the remaining 3 watertight reconstructions of actual mechanical parts (see Fig. 13). For each part we produced synthetic range images using a virtual 3D imaging device with a sensing grid of 1000×1000 points. Each part was captured 10 times with random orientations, thus producing a total of 60 virtual ranges. Finally, each range image was perturbed by additive

Gaussian noise with standard deviation equal to 12% of the average edge length of the surface. In Fig. 13 we plot, for several levels of resampling, the average ratio between the ground-truth RMS error and the RMS error obtained after applying the registration process. Note that we were not able to get stable registrations with fewer than 250 points per model. We can observe that the quality of the final registration increases with the number of resampled points (which is of course expected) and that the additional improvement that can be obtained with more than 1000 artificial vertices is almost negligible.

6.8. Some qualitative results

In addition to the quantitative experiments presented, we also performed some qualitative tests. While these tests do not offer a measurable comparison between the results obtained by different methods, they certainly help in putting the number presented so far in perspective. It is in fact sometimes harder to tell how much the RMS ratio affects the registration than judging some anecdotal alignment.

In Fig. 14 we show some coarse alignments obtained with the four methods under comparison. We are using surfaces with 100% overlap and moderate noise just for illustrative purposes. In fact, this way, it is easier to recognize the non-perfectly aligned areas all over the surfaces. The reader should refer to the previous section for a complete quantitative analysis covering different levels of overlap and noise.

In this particular example we can see that Spin Images fail on the Dragon, MeshDOG does not perform well with any of the tested meshes, while the Four Points Congruent Sets method obtains good results in all cases. On the other hand, GTR exhibits by far the best alignment in every example.

Fig. 15 illustrates qualitative differences with the registration obtained by using different descriptors with and without ICP refinement. The goal of this test is to visually illustrate that our method works very well also without applying the additional ICP step, thus it can be effectively used as a proper fine alignment approach. Since it is very difficult to spot defects on registration after the refinement we decided to perform a full registration of ranges acquired from a laser scanner contained in a publicly available database [2] and to build a closed surface using the Poisson Surface Reconstruction technique [45].
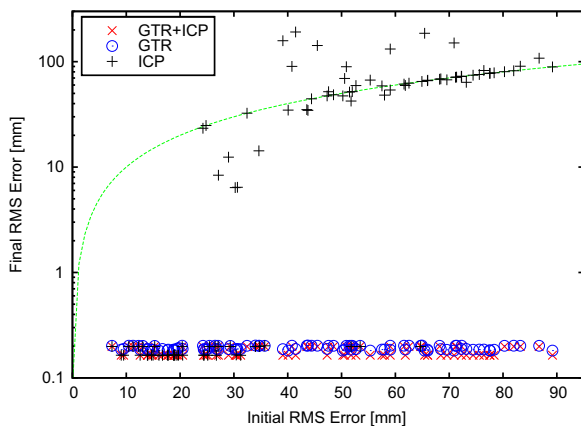

Fig. 12. Comparison of fine registration accuracies (the green dashed line represents y=x). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)
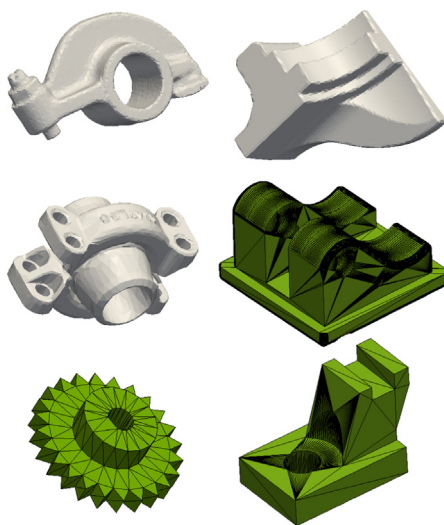

Fig. 13. Performance evaluation of the registration method when applied between range images and CAD models. Since one-to-one point correspondences are needed for the method to work, the CAD model has to be resampled for the process to work correctly. In this experiment we use both mechanical components acquired with a scanner (white), as well as models produced with a CAD software (green). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)
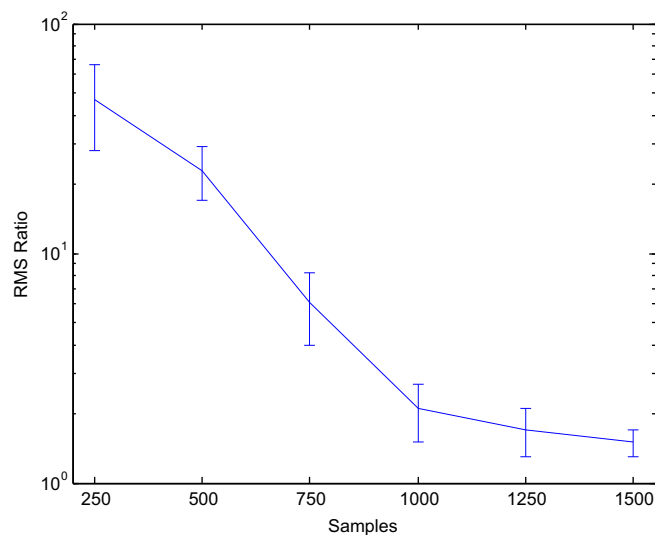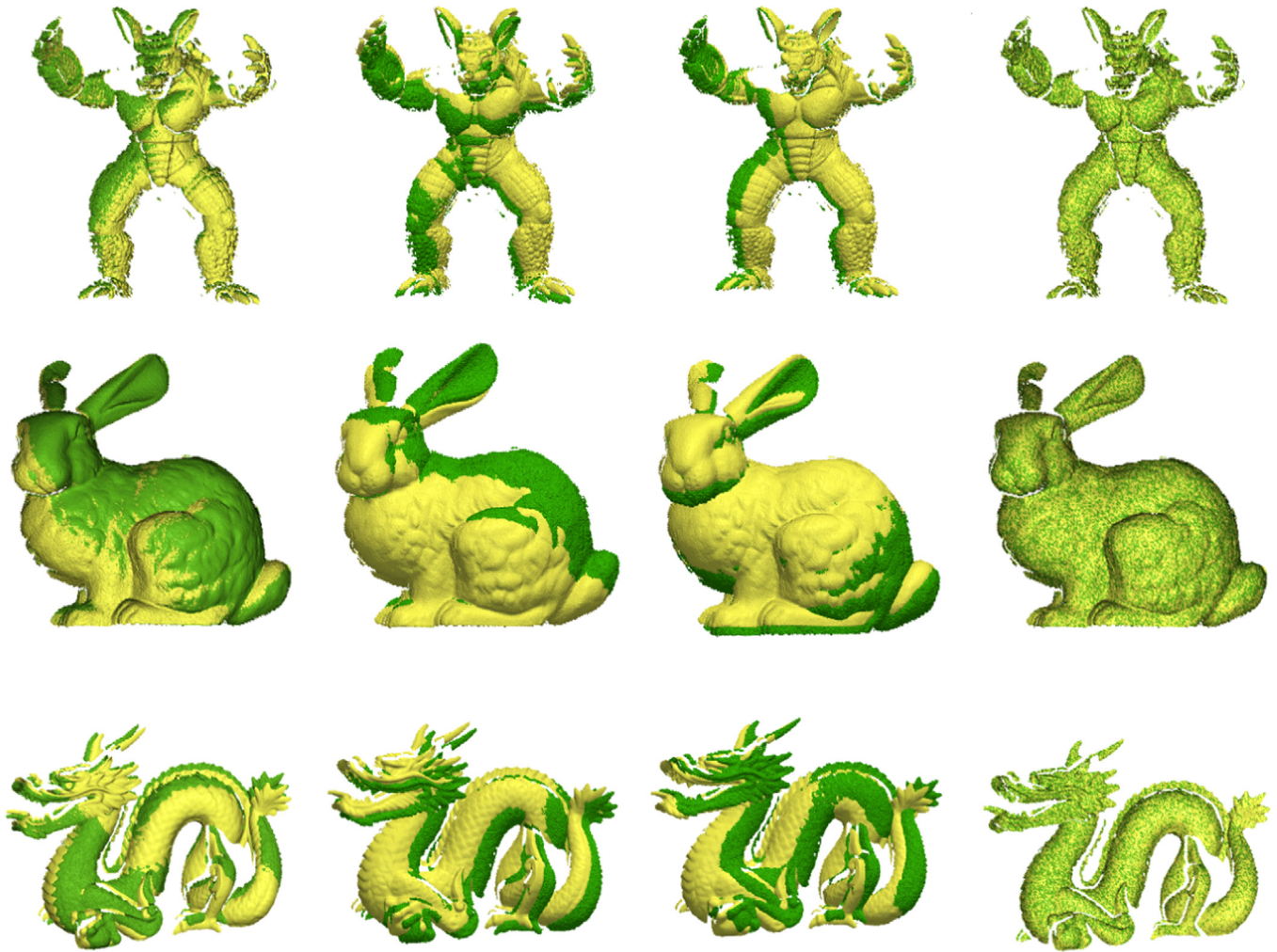
Fig. 14. Examples of surface registration obtained respectively with Spin Images (first column), MeshDOG (second column), 4PCS (third column) and our Game-Theoretic Registration technique (last column).

It is easy to note that for the same detail Surface Hashes and SHOT allow to build a smoother surface with no artifacts even before applying the refinement step. However a small defect can be spotted on the third finger on the foot reconstructed using SHOT without refinement. By contrast, Spin Images cannot be used to obtain reliable alignment even after applying ICP (the big artifacts that can be observed are mainly due to a few grossly misaligned ranges).

In Fig. 16 we show the result obtained using the GTR pipeline as a fully automatic tool for aligning 12 range images acquired by a laser scanner. Again, it can be observed that the effect of ICP refinement is very limited (if any).

Finally, in Fig. 17, we show a couple of examples of alignment between two range images [32,18] acquired using a Microsoft Kinect sensor. The first example shows the registration between two views of a man-made object characterized by an irregular surface and without any background or supporting surface that could help the registration process. The second example shows the alignment between two views of a complex scene captured from an office room containing several different objects.

6.9. Memory and execution time

Finally, we analyze the memory and cpu time requirements for our method.

The memory needed depends on the number of strategies in the initial set S. Since the payoff matrix sets a compatibility between each pair of strategies, the memory required is quadratic with $|S|$. To give a rough figure, with 1000 points sampled from the model and 6 neighbors for the initial matches, a single-precision matrix would require a little more than 64 MB. Of course, it is not really needed to materialize the payoff matrix. In fact, if the payoff function is simple enough, it could be advantageous to compute it on the fly during the iteration of the replicator dynamics. Actually, given that memory access is often the bottleneck on modern architectures, this could even speed-up computation (especially for GPU-based implementations of the replicator dynamics).

Regarding the execution time, each iteration of the replicator dynamics is quadratic with $|S|$. In principle, convergence time could be estimated using stochastic modelling [48,49]. However, it is not easy to state how many iterations are required to converge as it depends on many factors and parameters. In Fig. 18 we plotted a point cloud that relates the number of strategies with the convergence time (on our setup) for a large number of trials generated with the database of meshes adopted in the previous experiments. The convergence time shown includes all the iterations needed to reach a stable state according with the stop criteria defined in Section 6.1. When dealing with a few thousands of strategies (which is the common case) the evolution happens in about one second, which is reasonable for most non-real time applications.
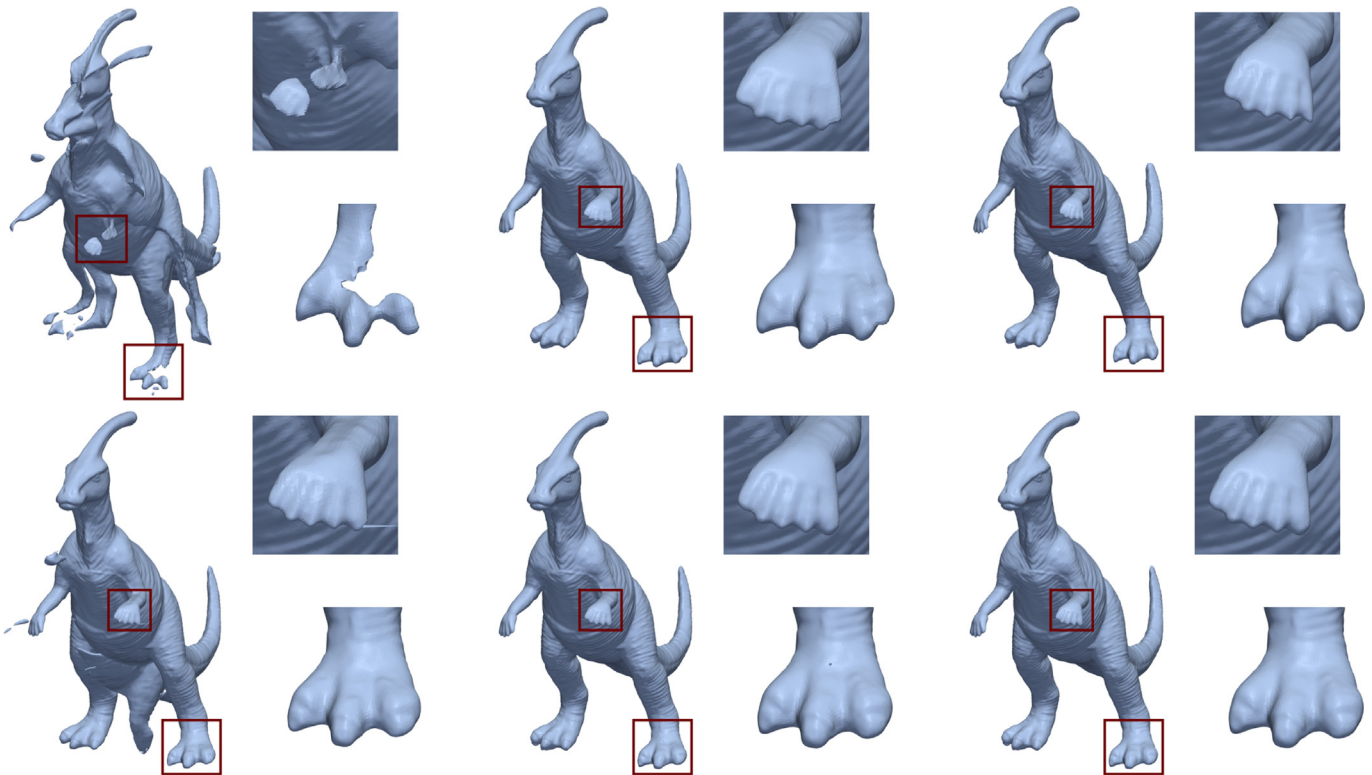
Fig. 15. Comparison of surface reconstruction using different descriptors before (first row) and after (second row) ICP enhancement of the coarse registration obtained respectively using Spin Images (first column), SHOT (second column) and Surface Hashes (third column). Notice that Surface Hashes allows to obtain a defect-free reconstruction even before the ICP step.

Fig. 16. Fully automatic registration of 12 views of the t-rex model from [2] using the GTR pipeline with Surface Hashes. A set of range images (first column) with unknown initial positions is given; coarse alignment is then performed with our technique and refined with a few steps of ICP (second and third column). The last column shows the final model obtained by Poisson reconstruction [45].
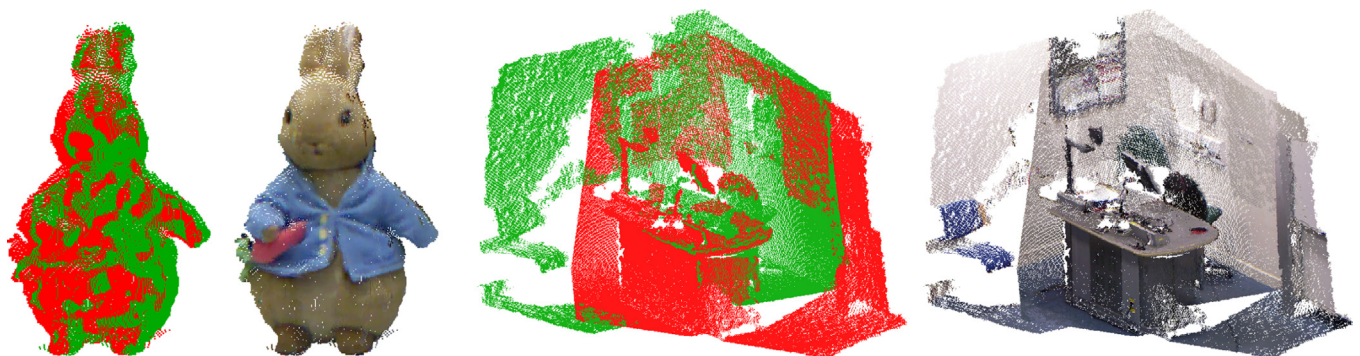

Fig. 17. Examples of alignment between surfaces acquired with a Microsoft Kinect sensor.
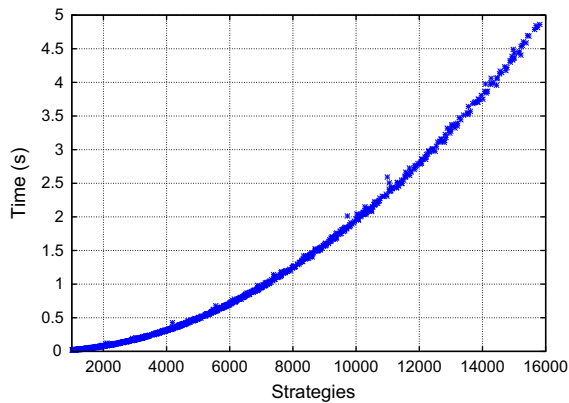
Fig. 18. Execution time of the proposed method with respect to the number of initial strategies. This scatter plot has been produced over several surfaces with different levels of occlusion. Note that the execution time depends only on the number of strategies and not on the surface.

Table 1

Comparison of the execution times of our method and two parametrizations of Super4PCS. The *tuned* variant makes use of optimally hand-tuned parameters for each model.

Method	Armadillo (27k)	Bunny (40k)	Dragon (43k)
Our method (s)	5.01	6.52	6.78
Super4PCS (s)	3.32	2.85	3.34
Super4PCS (tuned) (s)	0.89	1.70	1.85

Finally, in Table 1 we report a comparison of the execution times of our method and a fast variant of the four-points congruent sets method, named Super4PCS [18]. The latter method produces the same solutions as 4PCS, although in linear time. It should be noted that the timing performance and effectiveness of Super4PCS depend on the specific choice of parameters for the data at hand: estimates for vertex noise, surface overlap, and sampling quality constitute input for the method and directly affect the final matching time. For these reasons, we show the results obtained with a slower, but more robust parametrization as suggested by the authors (second row), as well as an optimal parametrization specifically tuned for each model (third row). In order to provide a fair comparison, the execution times reported for our method include *both* the construction of the similarity matrix, and the optimization step via replicator dynamics.

7. Conclusions

In this paper we introduced a novel game-theoretic technique that solves both the coarse and fine surface registration problems at once. Our approach has several advantages over the state-of-the-art: it does not require any kind of initial motion estimation, as it does not rely on spatial relationships between model and data points, and, unlike most inlier selection techniques, it is not affected by a large number of outliers since it operates an explicit selection of good inliers rather than using random selection or vote counting for validation. The approach has also shown to be general enough to accept different feature descriptors. From a theoretical point of view, a sound correspondence between optimal alignments and evolutionary equilibria has been presented and a wide range of experiments validated both the robustness of the approach with respect to noise and its performance in comparison with other well-known techniques.

Conflict of interest

None declared.

References

[1] T.S. Newman, A.K. Jain, A system for 3d cad-based inspection using range images, Pattern Recognit. 28 (10) (1995) 1555–1574.

[2] A.S. Mian, M. Bennamoun, R. Owens, Three-dimensional model-based object recognition and segmentation in cluttered scenes, IEEE Trans. Pattern Anal. Mach. Intell. 28 (October) (2006) 1584–1601.

[3] Y.-K. Ahn, Y.-C. Park, K.-S. Choi, W.-C. Park, H.-M. Seo, K.-M. Jung, 3d spatial touch system based on time-of-flight camera, WSEAS Trans. Info. Sci. Appl. 6 (2009) 1433–1442.

[4] D. Borrmann, J. Elseberg, K. Lingemann, A. Nüchter, J. Hertzberg, Globally consistent 3d mapping with scan matching, Robot. Auton. Syst. 56 (February) (2008) 130–142.

[5] Z. Zhang, Iterative point matching for registration of free-form curves and surfaces, Intl. J. Comput. Vis. 13 (2) (1994) 119–152.

[6] P.J. Besl, N.D. McKay, A method for registration of 3-d shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (2) (1992) 239–256.

[7] S. Rusinkiewicz, M. Levoy, Efficient variants of the icp algorithm, in: Proceedings of the Third International Conference on 3D Digital Imaging and Modeling, 2001, pp. 145–152.

[8] Y. Liu, Replicator dynamics in the iterative process for accurate range image matching, Int. J. Comput. Vis. 83 (1) (2009) 30–56.

[9] S. Granger, X. Pennec, A. Roche, Rigid point-surface registration using an em variant of icp for computer guided oral implantology, in: Proceedings of the 4th International Conference on Medical Image Comput. and Computer-Assisted Intervention, Springer-Verlag, London, UK, 2001, pp. 752–761.

[10] Y. Chen, G. Medioni, Object modeling by registration of multiple range images, in: Proceedings of 1991 IEEE International Conference on Robotics and Automation, vol. 3, 1991, pp. 2724–2729.

[11] T. Masuda, Registration and integration of multiple range images by matching signed distance fields for object shape modeling, Comput. Vis. Image Underst. 87 (1–3) (2002) 51–65.

[12] M.A. Fischler, R.C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.

[13] O. Chum, J. Matas, Matching with PROSAC—progressive sample consensus, in: CVPR, IEEE Computer Society, Washington, DC, USA, 2005, pp. 220–226.

[14] D.H. Chung, I.D. Yun, S.U. Lee, Registration of multiple-range views using the reverse-calibration technique, Pattern Recognit. 31 (4) (1998) 457–464.

[15] J.-P. Tarel, H. Civi, D.B. Cooper, Pose estimation of free-form 3d objects without point matching using algebraic surface models, in: Proceedings of IEEE Workshop Model Based 3D Image Analysis, Mumbai, India, 1998, pp. 13–21.

[16] C.-S. Chen, Y.-P. Hung, J.-B. Cheng, Ransac-based darces: a new approach to fast automatic registration of partially overlapping range images, IEEE Trans. Pattern Anal. Mach. Intell. 21 (11) (1999) 1229–1234.

[17] D. Aiger, N.J. Mitra, D. Cohen-Or, 4-points congruent sets for robust surface registration, ACM Trans. Graph. 27 (3) (2008) 85, 1–10.

[18] N. Mellado, D. Aiger, N.J. Mitra, Super 4PCS: fast global pointcloud registration via smart indexing, Comput. Graph. Forum 33 (5) (2014) 205–215.

[19] J. Salvi, C. Matabosch, D. Fofi, J. Forest, A review of recent range image registration methods with accuracy evaluation, Image Vis. Comput. 25 (5) (2007) 578–596.

[20] B.K.P. Horn, Closed-form solution of absolute orientation using unit quaternions, J. Opt. Soc. Am. A 4 (April (4)) (1987) 629–642.

[21] C. Harris, M. Stephens, A combined corner and edge detector, in: Proceedings of Fourth Alvey Vision Conference, 1988, pp. 147–151.

[22] D. Marr, E. Hildreth, Theory of edge detection, Royal Soc. Lond. Proc. Ser. B 207 (February) (1980) 187–217.

[23] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust wide-baseline stereo from maximally stable extremal regions, Image Vis. Comput. 22 (10) (2004) 761–767, British Machine Vision Computing 2002.

[24] K. Mikolajczyk, C. Schmid, An affine invariant interest point detector, in: ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I, Springer-Verlag, London, UK, 2002, pp. 128–142.

[25] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 20 (2003) 91–110.

[26] T.T. Herbert Bay, L. Van Gool, Surf: speeded up robust features, in: 9th European Conference on Computer Vision, vol. 3951, 2006, pp. 404–417.

[27] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Patt. Anal. Mach. Intell. 27 (10) (2005) 1615–1630.

[28] C.S. Chua, R. Jarvis, Point signatures: a new representation for 3d object recognition, Int. J. Comput. Vis. 25 (October (1)) (1997) 63–85.

[29] A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3d scenes, IEEE Trans. Pattern Anal. Mach. Intell. 21 (5) (1999) 433–449.

[30] H. Pottmann, J. Wallner, Q.-X. Huang, Y.-L. Yang, Integral invariants for robust geometry processing, Comput. Aided Geom. Des. 26 (1) (2009) 37–60.

[31] A. Zaharescu, E. Boyer, K. Varanasi, R.P. Horaud, Surface feature detection and description with applications to mesh matching, in: Proceedings of the IEEE

Conference on Computing Visual and Pattern Recognition, Miami Beach, Florida, June 2009.

[32] F. Tombari, S. Salti, L. Di Stefano, Unique signatures of histograms for local surface description, in: ECCV 2010—11th European Conference on Computer Vision, 2010, pp. 356–369.

[33] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, 3D object recognition in cluttered scenes with local surface features: a survey, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, pp. 2270–2287.

[34] A. Albarelli, S. Rota Bulò, A. Torsello, M. Pelillo, Matching as a non-cooperative game, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE Computer Society, 2009.

[35] A. Albarelli, E. Rodolà, A. Torsello, Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: a game-theoretic perspective, Int. J. Comput. Vis. (March) (2012) 36–53.

[36] E. Rodolà, A. Albarelli, F. Bergamasco, A. Torsello, A scale independent selection process for 3D object recognition in cluttered scenes, Int. J. Comput. Vis. (March) (2013) 129–145.

[37] A. Albarelli, E. Rodolà, A. Cavallarin, A. Torsello, Robust figure extraction on textured background: a game-theoretic approach, in: Proceedings—International Conference on Pattern Recognition, ser. ICPR '10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 360–363.

[38] E. Rodolà, A.M. Bronstein, A. Albarelli, F. Bergamasco, A. Torsello, A game-theoretic approach to deformable shape matching, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2012.

[39] J. Weibull, Evolutionary Game Theory, MIT Press, Cambridge, Massachusetts, 1995.

[40] P. Taylor, L. Jonker, Evolutionarily stable strategies and game dynamics, Math. Biosci. 40 (1978) 145–156.

[41] G. Turk, M. Levoy, Zippered polygon meshes from range images, in: Proceedings of SIGGRAPH 94, ACM, New York, NY, USA, 1994, pp. 311–318.

[42] V. Krishnamurthy, M. Levoy, Fitting smooth surfaces to dense polygon meshes, in: Proceedings of SIGGRAPH 96, 1996, pp. 313–324.

[43] B. Curless, M. Levoy, A volumetric method for building complex models from range images, in: Proceedings of SIGGRAPH 96. ACM, New York, NY, USA, 1996, pp. 303–312.

[44] O.T. Carmichael, D.F. Huber, M. Hebert, Large data sets and confusing scenes in 3-d surface matching and recognition, in: 3DIM, 1999, pp. 358–367.

[45] M. Kazhdan, M. Bolitho, H. Hoppe, Poisson surface reconstruction, in: Proceedings of the Fourth Eurographics Symposium on Geometry Processing, ser. SGP '06, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2006, pp. 61–70.

[46] A. Albarelli, E. Rodolà, F. Bergamasco, A. Torsello, 2011. A non-cooperative game for 3D object recognition in cluttered scenes. In: Proceedings of the 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2011, art. no. 5955368, pp. 252–259. http://dx.doi.org/10.1109/3DIMPVT.2011.39.

[47] A. Torsello, E. Rodolà, A. Albarelli, 2011. Sampling Relevant Points for Surface Registration. In: Proceedings of the 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), vol., no., pp. 290–295, 16–19 May 2011. http://dx.doi.org/10.1109/3DIMPVT.2011.43.

[48] S. Balsamo, 2007. A. Marin: queueing networks, LNCS 4486. In: Proceedings of the 7th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, pp. 34–82, 2007. http://dx.doi.org/10.1007/978-3-540-72522-0_2.

[49] Simonetta Balsamo, Andrea Marin. 2011. Performance engineering with product-form models: efficient solutions and applications. In: Proceedings of the 2nd ACM/SPEC International Conference on Performance engineering (ICPE '11). ACM, New York, NY, USA, pp. 437-448. http://dx.doi.org/10.1145/1958746.1958812, http://doi.acm.org/10.1145/1958746.1958812.

Andrea Albarelli worked in the industry for 10 years before receiving his Ph.D. in Computer Science in from the University Ca' Foscari Venice in 2010. He has taught Computer Vision, Computer Architecture and Information Theory. He co-authored more than 50 peer-reviewer technical papers, mainly in the field of Computer Vision, with particular attention to issues of representation and processing of 3D data and the adoption of Game Theory in the context of Pattern Recognition problems, ranging from point-pattern matching to surface registration. He has participated in several research and technology transfer projects funded by both public and private partners. Since 2010, he is a founding member of an academic spinoff acting as a bridge between industry and Computer Vision research. He held visiting researcher positions at the Department of Computer Science, Technische Universitat Munchen (TUM) and at the Research Centre for Humanities, University of Nova Gorica. In 2010 he was winner of the tender "IMPRESA", sponsored by the Ministry of Economic Development, the prize "Working Capital" for innovative young researchers and the "NVIDIA Best Paper Award". In 2011 he won the "Award for Research" established by Venice University to reward young researchers exhibiting the highest scientific impact.

Emanuele Rodolà received the B.S./M.S. degree in computer engineering from the University of Rome "Tor Vergata", Italy, in 2008 and the Ph.D. degree in computer science from Ca' Foscari University of Venice in 2012. In 2012 he was a JSPS Research Fellow at the University of Tokyo, Intelligent Systems and Informatics Lab, and since 2013 he is an Alexander von Humboldt Postdoctoral Fellow at the Technical University of Munich, Computer Vision group. He has been in the program committees of various international conferences and journals, and was recognized as Outstanding Reviewer at CVPR 2013 and ECCV 2014. His research interests include 3D shape analysis, matching, reconstruction and has around 30 publications on these topics in refereed conferences and journals.

Andrea Torsello received his Ph.D. in computer science at the University of York, UK, and is currently an Associate Professor at Ca' Foscari University of Venice, Italy. His research interests are in the areas of computer vision and pattern recognition, in particular, the interplay between stochastic and structural approaches as well as game-theoretic models. Dr. Torsello has published over 100 technical papers in refereed journals and conference proceedings and has been in the program committees of various international conferences and workshops.