A repeatable and efficient canonical reference for surface matching

Alioscia Petrelli, Luigi Di Stefano
Computer Vision Lab, DEIS, University of Bologna, Italy
{ *alioscia.petrelli, luigi.distefano*} *@unibo.it*

***Abstract*—The paper investigates on canonical references used for local surface description and matching. We formulate a novel proposal and carry out an extensive experimental evaluation addressing two major surface matching scenarios, namely shape registration and object recognition. We provide also a methodological contribution as, unlike previous work in the field, we propose a repeatability metric that captures the actual impact of the adopted local reference frame algorithm within surface matching tasks based on local 3D descriptors. Our proposal outperforms existing algorithms by a wide margin on several datasets acquired with different devices, such as laser scanners, stereo cameras and the Kinect, and in experiments relying on randomly extracted feature as well as state-of-the art keypoints.**

***Keywords*-local reference frame; surface matching; 3D descriptor**

I. Introduction

Object recognition and surface registration are two fundamental tasks in 3D computer vision. The former concerns determining the presence of 3D objects and estimating their poses in scenes acquired by 3D sensors. The latter is the task of aligning partial 3D views of a given object acquired from different vantage points. Both ground on surface matching, a challenging problem involving the search for corresponding surface elements. As global approaches can hardly deal with nuisances such as clutter, occlusions and missing regions, recent research has been mainly focused on the definition of descriptors that encode the neighborhood of 3D points into distinctive and robust representations to discover correspondences between surfaces. An essential trait of such local 3D descriptors is the invariance to 3D rotation. In the last fifteen years a remarkable number of local 3D descriptors have been designed, most of them [1]–[11] achieving invariance to rotation by establishing a *Local Reference Frame* (LRF) and describing the neighborhood (also *support*) according to LRF coordinates.

Both [8] as well as, more extensively, [12] have recently highlighted the importance of the algorithm used to establish the LRF, showing how its repeatability significantly impacts the effectiveness of a surface matching process based on local 3D descriptors. In particular, [12] proposes an evaluation of LRF algorithms carried out on a large 3D data corpus acquired by different laser scanning devices and focused on the surface registration task. Besides, [12] proposes a novel approach (hereinafter referred to as *Board*[1]) that delivers state-of-the-art performance.

In this paper we improve the investigation and evaluation on LRF algorithms proposed in [12]. As a first contribution, we extend the collection of datasets and methods considered in the experiments. In particular, in registration experiments we add seven datasets acquired by means of two different low-cost acquisition system, namely a *Spacetime Stereo* set-up and a *Kinect* device. To fill up the lack of object recognition experiments in [12], we evaluate the performance of the methods on three 3D versus 2.5D object recognition datasets characterized by different levels of clutter, occlusion, point density and noise. As for methods, besides those compared in [12], we consider also a recent proposal by Dos Santos *et al*. [14]. As a second contribution, we propose a novel metric to characterize the performance of LRF algorithms. Indeed, based on experimental analysis, we observed that, in practical surface matching tasks, local descriptors completely lose their distinctiveness (i.e. yield wrong matches) as soon as the LRFs at corresponding features get misaligned, which means that the actual misalignment above a certain degree turns out irrelevant for the specific purpose of surface matching. This effect can be perceived clearly in Fig.1 of [12], wherein each descriptor loses completely its matching power as the misalignment gets above a certain degree (and, overall, all are useless as the misalignment gets over about 25 degrees). In other words, as far as surface matching is concerned, two LRFs at corresponding features are simply aligned (so that the features hold the potential to be matched) or not (and so the features will hardly be matched). Therefore, unlike [12] which quantifies repeatability based upon the mean rotation error between LRFs at corresponding feature pairs, we propose a metric which aims at estimating the percentage of aligned LRFs (i.e. of potentially matchable features) yielded by the considered algorithms. Finally, as third contribution, we introduce a novel LRF algorithm that synthesizes the key strengths of the other considered methods. The experimental results, within both the object recognition and registration scenarios, show coherently that the proposed algorithm significantly outperforms all known methods, according to both the repeatability index adopted in [12] as well as the new metric introduced in this paper.

[1]According to the name given to this method in the available implementation within the PCL library [13]

II. Considered Methods

In this section, we describe how to compute the LRFs considered in our study. Most of them are based on the computation of the eigenvectors of a covariance matrix of the 3D coordinates of the points, $\mathbf{p}_i$, lying within a spherical support of radius R centered at the feature point $\mathbf{p}$.

Mian [15]: the unit vectors of the LRF are given by the normalized eigenvectors of the covariance matrix:

$$\mathbf{\Sigma}_{\hat{\mathbf{p}}} = \frac{1}{k}\sum_{i=0}^{k}(\mathbf{p}_i - \hat{\mathbf{p}})(\mathbf{p}_i - \hat{\mathbf{p}})^T \quad (1)$$

where $\hat{\mathbf{p}}$ denotes the barycenter of the points lying within the support:

$$\hat{\mathbf{p}} = \frac{1}{k}\sum_{i=0}^{k}\mathbf{p}_i \quad (2)$$

However, while the eigenvectors of (1) define the principal directions of the data, their sign is not defined unambiguously. To partially solve this problem, the z axis is disambiguated by computing the inner product between $\mathbf{n}$ and the two possible unit vectors z^+ and z^-, so as to choose the unit vector yielding a positive product [2].

SHOT [8]: to avoid computation of (2), the barycenter appearing in (1) is replaced with the feature point. Moreover, to improve repeatability in presence of clutter in object recognition scenarios, a weighted covariance matrix is computed by assigning smaller weights to more distant points:

$$\mathbf{\Sigma}_{\mathbf{pw}} = \frac{1}{\sum_{i:d_i\leq R}(R-d_i)}\sum_{i:d_i\leq R}(R-d_i)(\mathbf{p}_i-\mathbf{p})(\mathbf{p}_i-\mathbf{p})^T \quad (3)$$

with $d_i = \|\mathbf{p}_i - \mathbf{p}\|_2$. To achieve true rotation invariance, a sign disambiguation technique inspired by [16] is applied to the eigenvectors of (3). In particular, the sign of an eigenvector is chosen so as to render it coherent with the majority of the vectors it is representing. To deal with the case of an even number of vectors, the implementation available in the PCL library [13] relies on the following procedure: points $\mathbf{p}_i$ are sorted by their distance to $\mathbf{p}$ and the median point is found. Hence, in case the initial set of points is even, only the median, the 2 preceding and the 2 following points are used to apply the disambiguation. The disambiguation is applied to the eigenvectors associated with the largest and smallest eigenvalues, in order to attain the unit vectors defining, respectively, the *x* and *z* axes. The third unit vector is computed via the cross-product $z \times x$.

PS [2]: the LRF associated with the Point Signatures descriptor is defined as follows. The intersection of the spherical support with the surface generates a 3D curve, $\mathbf{C}$, whose points are used to fit a plane. The z axis is directed along the normal to the fitted plane. In order to disambiguate z axis, the same method adopted by *Mian* is applied. The x axis is attained by defining a signed distance from the points belonging to $\mathbf{C}$ to the fitted plane. Points that lie in the same half space as the normal to the fitted plane are given a positive distance, those lying in the opposite half-space a negative distance. The point with the highest positive distance is then selected, and the projection on the fitted plane of the vector from this point to the feature point $\mathbf{p}$ defines the x axis. As usual, the third axis is computed via cross-product.

MeshHog [11]: support points, $\mathbf{p}_i$, are determined based on the geodesic rather than Euclidean distance. The z axis is robustly estimated by the mean of the normals of the 5-ring neighbourhood of point $\mathbf{p}$. The identification of the x axis is inspired by *SIFT* [17]. At each $\mathbf{p}_i$, the discrete gradient $\nabla_S f(p_i)$ is computed, $f(p_i)$ being the mean surface curvature. Gradient magnitudes are added to a polar histogram of 36 bins (covering 360°) and weighed by a Gaussian function of the geodesic distance from $\mathbf{p}$, with σ equal to half of support radius. To deal with aliasing and quantization, votes are interpolated tri-linearly between neighboring bins. While in *SIFT* histogram bins are filled according to gradient orientation, in [11] points $\mathbf{p}_i$ are projected onto the tangent plane defined by $\mathbf{n}$ and the orientation with respect to a random axis lying on such a plane is considered. Then, the chosen x axis orientation is given by the dominant bin in the polar histogram. At last, y is computed as $z \times x$.

DosSantos [14]: as pointed out in [12] and [14], the *SHOT* LRF is negatively affected by the uneven distribution of points within the support due to acquisitions from angularly distant vantage points. To reduce this effect, dos Santos *et al*. build an approximation of the covariance matrix by using point normals, instead of coordinates, weighted by influence areas of points (implemented as voronoi areas). The x and z axes disambiguation is obtained by orienting the two unit vectors toward the biggest influence area of the support. Finally, y is attained as $z \times x$.

Board [12]: in order to robustly estimate the z direction, the method fits a plane to the points within a small support of radius 5 $\times$ the average mesh resolution (hereinafter mr). To disambiguate the sign, the method computes the average normal, $\tilde{\mathbf{n}}$, over support points and chooses between z^+ and z^- so as to get a positive inner product with $\tilde{\mathbf{n}}$. The x axis estimation relies again on surface normals. For each point $\mathbf{p}_i$ with distance to $\mathbf{p}$ larger than 0.85 $\times$ R, the angle between its normal $\mathbf{n}_i$ and the z axis is computed. The x axis is directed from $\mathbf{p}$ towards the point $\mathbf{p}_{min}$ that reveals the largest angle. Finally the axis is projected onto tangent plane. For the sake of robustness, instead of strictly considering $\mathbf{n}_i$, the average normal over the 2-ring neighbourhood of $\mathbf{p}_i$ is computed. Board tries also to overcome a peculiar problem of partial shape matching that affects feature points extracted near the borders of the views. These points show missing regions within the support that significantly deteriorate the

[2]We knew of the presence of the z axis sign disambiguation step, not specified in the paper, by a personal communication with the author.

repeatability of the LRF computation. To deal with this issue, the method deploys a heuristic, consisting of three stages, aimed at assessing whether $\mathbf{p}_{min}$ would lie in a region actually missing in the case the support were complete. In the first stage, missing regions are identified. The second estimates whether one missing region could contain $\mathbf{p}_{min}$ by evaluating the angle of normals $\mathbf{n}_a$ and $\mathbf{n}_b$ of the two points, $\mathbf{p}_a$ and $\mathbf{p}_b$, at the boundaries of the considered region. If $\mathbf{n}_a$ and $\mathbf{n}_b$ are sufficiently inclined, the last stage estimates the position of $\mathbf{p}_{min}$ in the missing region based again on the angles of $\mathbf{n}_a$ and $\mathbf{n}_b$ and on the consideration that, intuitively, $\mathbf{p}_{min}$ is closer to $\mathbf{p}_a$ if the angle of $\mathbf{n}_a$ is greater that the angle of $\mathbf{n}_b$ and vice versa.

III. Proposed Method

In our study we introduce a novel method (hereinafter *P*) aimed at exploiting those traits of the other approaches that turn out more beneficial in the definition of a repeatable LRF. As shown in [12], the way *Board* computes the z axis is the most repeatable, so we rely on the same procedure. Unlike methods based on the covariance matrix, which exploit all the support for the computation of the z axis, *Board* exploits only a small subset of points (depicted in blue in Fig.1) centered at the considered feature point. This enables better handling of nuisances such as clutter and occlusions that may alter the shape of the whole support and render unstable the estimation of the z axis. This approach makes the estimation of the z axis robust to noise as well as stable regardless of the wider extension of the support used for estimation of the x axis. This feature is crucial, since, likewise other methods such as *PS*, *Board* and *MeshHog*, with our proposal the computation of the x axis depends on that of the z axis.

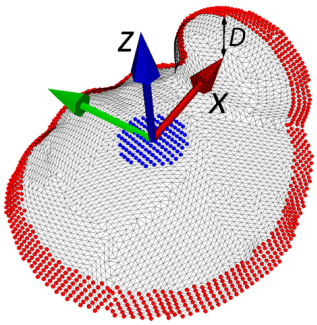


Figure 1. An example helping to describe the proposed method.

Regarding the x axis, again similarly to *Board*, we consider only a subset of points $\mathbf{p}_i$ lying at the periphery of the support (depicted as red points in Fig.1. This, together with the small support used for the z axis, provides also good scalability with respect to the size of the support radius. Whilst in approaches such as *MeshHog*, and methods based on covariance matrixes alike, all support points contribute

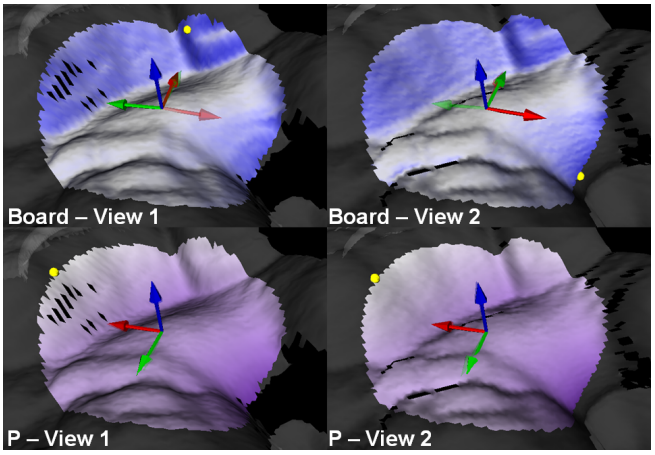


Figure 2. Comparison between *Board* (top) and the proposed method *P* (bottom) on the same two corresponding points extracted from two partial views. x axes are shown as red arrows, y axes are in green whilst z axes in blue. In the figures, the LRF found at the point is opaque whereas the LRF found in the other view is overlaid semi-transparently for comparison. In the figures concerning *Board*, the behaviour of surface normals is illustrated by assigning a darker blue to points having normals more inclined with respect to the z axis, whereas for the figures regarding *P* points with higher signed distances are denoted by brighter purple. Yellow spheres indicate the support points chosen by the methods to define the x axis. Whilst *Board* selects two different points (due to the similar degree of the normal inclinations), *P* coherently identifies the same characteristic point.

to establish the direction and sign of the x axis, in *Board* and *PS* only a single highly distinctive point is identified to define where to point at the x axis. Accordingly, these approaches tend to be more robust to point density variations and missing regions. Unlike *Board*, which identifies the characteristic point based on point normals, our proposal grounds its choice on the signed distance to the tangent plane defined by the z axis and $\mathbf{p}$. Precisely, the x axis is given by the normalized projection onto the tangent plane of the vector from $\mathbf{p}$ to $\mathbf{p}_{max}$, the latter being the point within the periphery of the support showing the largest signed distance ($\mathbf{D}$ in Fig.1) to the tangent plane. Our experimental analysis has shown that the signed distance is a more distinctive cue than the normal and hence provides higher repeatability especially on flatter supports, i.e. at surface patches where the detection of a repeatable tangent direction is more challenging. Furthermore, signed distances are direct measurements related to the shape of the support, whereas normals represent their derivatives and as such tend to be more sensitive to noise. *Board* deploys a heuristic to cope with the issue of missing regions in proximity of the borders of a partial view, while our algorithm inherently handles this nuisance as points close to the borders of a view tend to exhibit lower signed distances than points closer to the center of the view. Fig.2 shows how the proposed method can capture better than *Board* the peculiar shape of the surface and, hence, better identify the characteristic point to establish the x axis.

IV. Evaluation Methodology

To compare the methods described in previous sections, we extend and enrich the evaluation presented in [12]. In particular, we formulate a novel proposal for the metric adopted to assess the performance of LRFs and consider also a 3D object recognition scenario, together with further datasets for partial shape registration experiments.

In [12] only datasets acquired by different type of laser scanners were considered. However, the broad use of low-cost acquisition systems in the last years calls for evaluating algorithms also on datasets acquired by such devises. So, we add to the collection of publicly available datasets used in [12] for registration experiments, 3 datasets acquired by a *Spacetime Stereo* set-up (*MarioStereo*, *DuckStereo*, *FrogStereo*) and 4 datasets acquired by a *Kinect* device (*MarioKinect*, *DuckKinect*, *FrogKinect* and *SquirrelKinect*). For each new dataset we obtained the ground truth by carrying out a manual coarse registration followed by a global refinement by means of *Scanalyze*[3].

As for the general methodology, we mainly follow the procedure suggested in [12], in which, given a view pair, a set of corresponding points are extracted by relying on ground truth rototranslation and, then, the related LRFs are computed. In order to evaluate the repeatability of a method on a dataset, misalignment errors are computed for every pair of corresponding LRFs so as to obtain a repeatability index by averaging misalignment errors: first across all corresponding LRFs of a view pair, then over all view pairs. In our present evaluation we have introduced just a few adjustments. First, we discard view pairs that show an overlapping area lower than 10%, instead of relying on the number of extracted feature correspondences. Furthermore, while for experiments on randomly extracted features we adopt the same procedure as in [12], in the experiments relying on keypoints we make use of the *ISS* detector [6] rather than *MeshDog*, as the former has provided superior performance and a significantly faster speed in a recent evaluation [18]. Moreover, we extract features at 4 different scales, $(5, 10, 20, 30) \times mr$, from all dataset views, then, for each view pair (V_1, V_2), we apply the ground truth rigid motion to each feature point $p_{i,2}$ of V_2 so as to check if a feature point $p_{i,1}$ in V_1 is closer than $8 \times mr$ from $p_{i,2}$ [4]. We consider as corresponding those feature pairs $(p_{i,1}, p_{i,2})$ satisfying such condition. We choose this procedure so as to account also for a possible imprecise localization of feature points when assessing the performance of LRF algorithms. Both on random features and ISS keypoints, the LRFs are computed by using a large set of radii R, i.e. $(5, 10, 20, 30, 40, 50, 60) \times mr$.

[3]http://graphics.stanford.edu/software/scanalyze/

[4]This stems from experiments carried in our Lab showing that in many diverse scenarios $8 \times mr$ turns out a good inlier threshold to estimate 3D rigid body transformations within a RANSAC paradigm

The most remarkable variation with respect to the methodology proposed in [12] concerns the metric to compute the performance, i.e. the repeatability, of the LRF algorithms at corresponding points. In [12], given a pair of views VP_n, for each point correspondence i, the index $MeanCos'_{i,n}$ is computed as the average between the alignment error of the x and z axes. This index properly represents the rotation error between two LRFs computed by an algorithm at corresponding points. Nevertheless, as already pointed out in Sec.I, we found that the degree of misalignment between corresponding LRFs does not capture effectively the kind of "on-off" behavior of LRF algorithms with respect to the descriptor matching process. Indeed, corresponding descriptors keep their distinctiveness and thus can be matched effectively only in case the established LRFs are very well aligned. Conversely, if corresponding LRFs are misaligned, it does not really matter in practice how large is the actual rotation error, as description is so corrupted that features can no longer be matched. Hence, for each point correspondence i of view pair VP_n, we attempt to evaluate whether the computed LRFs can be considered aligned or not. Purposely, we calculate $MeanCos'_{i,n}$ as in [12] and then define a novel performance index as follows:

$$A_{i,n} = \begin{cases} 1, \ MeanCos'_{i,n} \geq T_A \\ 0, \ MeanCos'_{i,n} < T_A \end{cases} \tag{4}$$

where T_A is a threshold that discriminates between aligned and misaligned LRFs. In principle such a threshold is somehow descriptor-dependent, given that different descriptors may tolerate different degrees of misalignment, owing to their own nature as well as the setting of parameters. For example, following the taxonomy in [8], *signatures* are inherently more sensitive than *histogram*-based methods to the degree of alignment of LRFs (as also vouched by the plot of Fig. 1 in [12])), and the chosen bin size might impact notably the ability of the latter to tolerate a given amount of misalignment between corresponding LRFs. Nonetheless, as it will be shown in Sec.V, the threshold chosen to distinguish between aligned and misaligned LRFs does not affect the relative ranking between LRF algorithms, so that here we can arbitrarily set $T_A = 0.97$.

To define a global figure of merit concerning a specific dataset, we follow the same approach as proposed in [12] to get the global index $MeanCos'$. First, $A_{i,n}$ measurements are aggregated by averaging across all point correspondences i to attain the percentage, $\bar{A}_n$, of aligned LRFs for view pair VP_n. Then, $\bar{A}_n$ figures are aggregated again by averaging over all view pairs to get the final index $\bar{A}$.

In object recognition experiments we use 3 different kinds of datasets. To evaluate the methods on detailed shapes, we run the experiments on the well-known *Mian* dataset [15], which was acquired by the Minolta Vivid 910 laser scanner. This dataset consists of 5 models and 50 cluttered

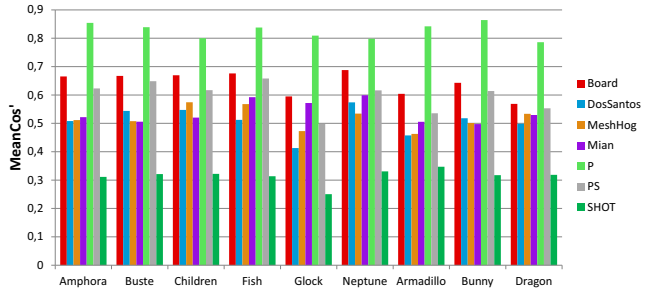
and highly occluded scenes. To compare the methods also on less accurate and noisier data, we rely on the *Kinect* dataset [18], that is made up of 6 models and 27 scenes, again with significant clutter and occlusions. Finally, we consider a synthetic dataset (*Virtual Stanford*), which presents a lower amount of clutter. We created 50 different scenes by placing at random (but avoiding surface intersections) 3, 4, or 5 models belonging from the Stanford 3D Scanner Repository [19] and acquiring, for each scene, 6 views from different vantage points by way of a software tool that simulates the Kinect device according to the guidelines provided in [20]. Given a vantage point, a 640×480 pixels depth-map is generated by ray casting, then Gaussian noise is added and z-coordinates are quantized, with both the noise variance and the quantization step increasing with distance. Bilateral filtering is then applied to the depth maps before the actual processing to smooth out noise and quantization artifacts. As for the methodology, we introduce two differences with respect to object recognition experiments. Firstly, as clutter disrupts repeatability with wide supports, we run experiments up to a smaller maximum radius (i.e. $50 \times mr$). Secondly, while in shape registration we compute $\bar{A}_n$ by aggregating $\bar{A}_{i,n}$ over all the corresponding features of a view pair VP_n, in object recognition experiments we aggregate $A_{i,s}$ across all corresponding features extracted from scene s, so that then the global index $\bar{A}$ is achieved by aggregation of the $\bar{A}_s$ across all dataset scenes.

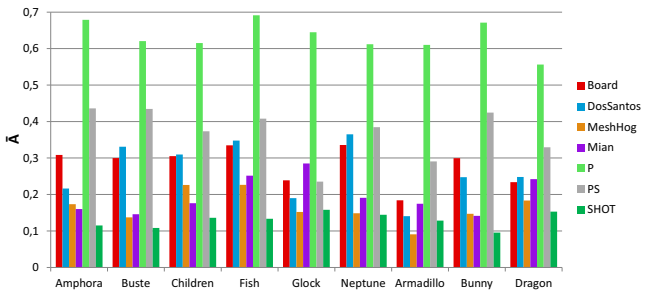
V. Experimental Results

This section reports the experimental results obtained by the considered methods on all datasets. Both in the registration and object recognition scenarios, for each method the experiments have been run across all the considered support radii and the radius yielding the highest repeatability has been chosen to define the score associated with the method.

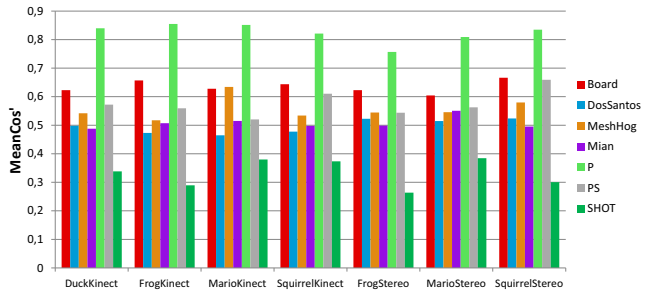
Fig.3 shows the repeatability of LRF algorithms in the task of partial shape registration by using randomly detected feature points. The Figure reports the scores with respect to the index $MeanCos'$ adopted in [12], which measures the mean rotation error between corresponding features, as well as those achieved by the metric denoted as $\bar{A}$ proposed in this paper, which instead estimates the percentage of aligned LRFs and disregards the actual rotation error found in misaligned ones. First of all, the charts coherently highlight that the method proposed in this paper (referred to as *P*) neatly overcomes the other proposals on every dataset, with, in particular, a percentage of aligned LRFs most often above 60 % in the laser scanner datasets and above 50 % in the less detailed and nosier *Space time stereo* and *Kinect* datasets.

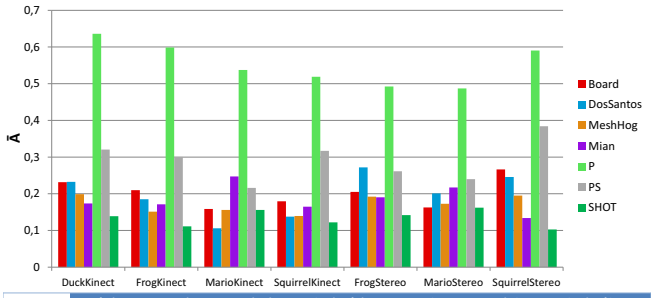
It is worth pointing out that the proposed algorithm provides consistently an increase between 20 to 30 % of aligned LRF (i.e. matchable features) with respect to the second best method in each experiment. As found also in [12], *PS* and *Board* exhibit good performance, also in the





	Amphora	Buste	Children	Fish	Glock	Neptune	Armadillo	Bunny	Dragon
Board	10	10	10	10	20	10	10	10	5
DosSantos	5	5	10	10	5	10	5	5	5
MeshHog	30	30	60	60	50	20	60	20	60
Mian	40	10	10	50	20	20	10	10	10
P	50	20	50	60	60	30	40	50	60
PS	10	10	10	20	10	10	5	10	5
SHOT	10	10	10	60	10	20	10	10	10





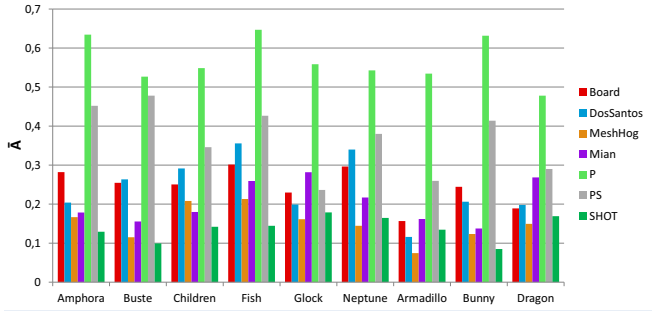
	DuckKinect	FrogKinect	MarioKinect	SquirrelKinect	FrogStereo	MarioStereo	SquirrelStereo
Board	20	20	50	10	10	20	20
DosSantos	20	10	10	10	20	20	10
MeshHog	60	40	60	30	60	60	60
Mian	20	20	50	60	20	20	20
P	60	60	60	40	60	60	60
PS	20	20	30	20	20	20	30
SHOT	20	30	50	60	30	30	30

Figure 3. Results of registration experiments with randomly extracted features. From top to bottom: repeatability scores according to $MeanCos'$ [12], according to the metric denoted as $\bar{A}$ and proposed in this paper, a table reporting for each method and dataset the support radius (in mr units) that maximizes $\bar{A}$. The 3 topmost figures refer to the laser scanner dataset used in [12], the next 3 figures to the *Space time stereo* and *Kinect* datasets proposed in this paper.

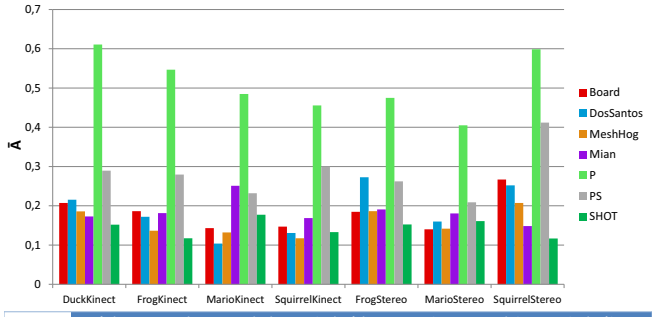
new datasets. However, in both types of datasets *PS* tends to outperform *Board* according to the new repeatability metric. The recent method by *DosSantos*, conceived to deal with the issue of local point density variations discussed in [12], turns out quite effective and achieves performance that, according to the new metric $\bar{A}$, are overall comparable to *Board*.

Compared to the results achieved on laser scanner datasets, the results on the datasets acquired by the *Space time stereo* and *Kinect* devices show degraded performance, which prove that, in general, all LRF methods are notably affected by the quality of the scanning devices.

Fig.4 reports the results in case of *ISS* keypoints. The charts basically confirm the ranking attained on randomly extracted features and prove the neat superiority of the novel proposal. Interestingly, although one might guess that keypoints would identify more distinctive surface patches than random features, so that, accordingly, LRF algorithms should inherently exhibit higher repeatability, for all methods the scores in Fig.4 turns out always slightly worse than in Fig.3. This is due to the peculiar nuisance of imprecise



	Amphora	Buste	Children	Fish	Glock	Neptune	Armadillo	Bunny	Dragon
Board	10	10	10	10	60	10	10	20	5
DosSantos	5	5	10	10	5	10	5	5	5
MeshHog	30	10	60	60	40	20	50	10	60
Mian	30	10	10	50	20	20	10	20	10
P	60	20	50	60	60	40	60	50	60
PS	10	10	10	20	20	10	10	10	10
SHOT	20	10	10	60	10	20	60	10	10



	DuckKinect	FrogKinect	MarioKinect	SquirrelKinect	FrogStereo	MarioStereo	SquirrelStereo
Board	20	30	20	20	40	50	20
DosSantos	20	10	10	10	20	20	20
MeshHog	60	40	60	30	60	60	60
Mian	20	20	50	50	20	20	30
P	60	60	60	50	60	60	60
PS	20	20	30	20	20	20	30
SHOT	60	40	50	60	30	40	30

Figure 4. Results of registration experiments on *ISS* keypoints. Repeatability scores are given according to $\bar{A}$.

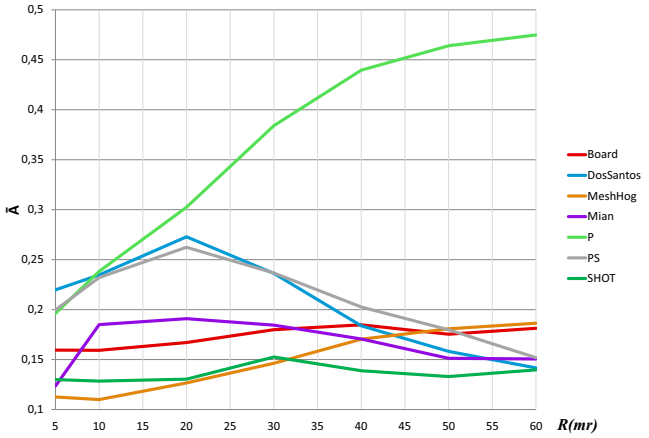
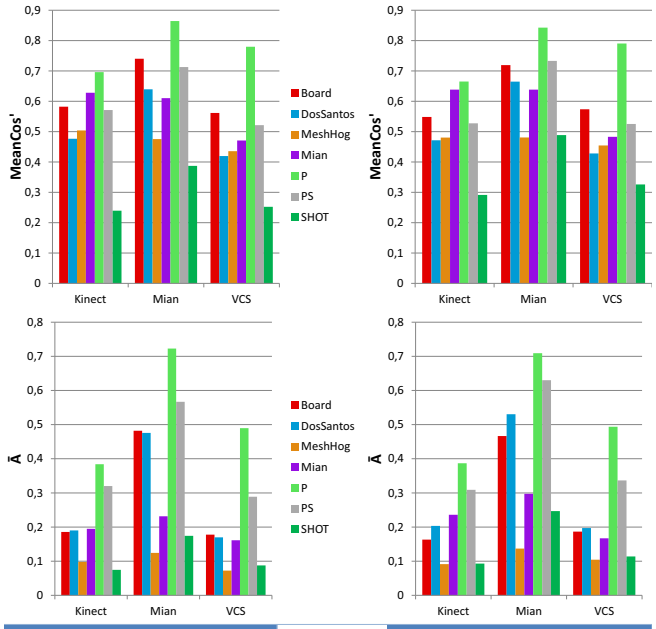


Figure 5. Repeatability vs. support radius (*ISS* keypoints, *FrogStereo* dataset).

localization injected in the experiments with keypoints (as discussed in Sec.IV), which shows a higher impact than the saliency of surface patches and overall lowers repeatability with respect to the case of randomly extracted (but precisely localized) features.

Concerning the radii that maximize repeatability, it emerges that, for the majority of datasets, *P* tends to use the widest supports, similarly to *MeshHog* and *Board* and, once in a while, *SHOT*. Conversely, *PS*, *Mian* and *DosSantos* rely on smaller radii. To better understand this behavior, Fig.5 plots the relation between the repeatability scores of the methods and the radius. Clearly, *P* strongly improves as the radius increases, as it is also the case of *Board* and *MeshHog* - though according to a milder trend. This is due to the way these methods compute the z axis. In fact, all of them rely on a smaller support, whereas the others use the entire supports that, in particular for large radii, can vary widely due to the presence of missing regions and, hence, rendering unstable the computation of the axis. *SHOT* suffers less this effect because of the weighting of covariance matrix with respect to point distances. It is worth noting the difference in performance between *Board* and our proposal even if they share the computation of z axis. Evidently, as the radius increases, the distance from the tangent plane turns out a more distinctive cue for a point than its normal direction.

The results related to the object recognition scenario, depicted in Fig.6, are coherent with the findings provided by registration experiments. In fact, *P* turns out consistently the most repeatable method (according to both the considered metrics), followed again by *Board*, *PS* and *DosSantos*. Comparing the results between the three datasets, the methods exhibit overall higher repeatability on *Mian*. This is to be ascribed to the different accuracy of the considered datasets. In fact, consistently with the shape registration scenario, the low-quality acquisition systems used for *Kinect* and *Virtual Stanford* datasets (made up, respectively, of real



Kinect	Mian	VCS		Kinect	Mian	VCS
20	10	10	**Board**	20	10	10
10	10	5	**DosSantos**	10	10	5
50	5	5	**MeshHog**	50	5	5
20	20	30	**Mian**	20	20	30
40	30	50	**P**	40	30	50
20	10	10	**PS**	20	10	10
10	10	10	**SHOT**	10	10	10

Figure 6. Results related to object recognition experiments. Randomly extracted features (left) and *ISS* keypoints (right).

and synthetic data) render the estimation of LRF a more challenging task.

For *P*, and the other methods that take advantage of the estimation of z axis on a smaller support alike, it is worth noticing that the radii that yield the highest repeatability are strongly correlated with the level of clutter and occlusions of the datasets. Indeed, whereas with *Virtual Stanford* these methods can exploit larger radii, the higher degree of clutter and occlusions in *Mian* and *Kinect* limits the extension of the support that can be deployed by the methods.

As discussed in Sec.IV, the metric $\bar{A}$ we use for the evaluation of the repeatability depends on the threshold T_A chosen to establish aligned upon misaligned LRFs, with such a threshold being in principle related to the descriptor used to match the features, given that different descriptors may tolerate different degrees of misalignment. However, Fig.7 plots, in the case of the *Amphora* dataset, the trend of $\bar{A}$ vs T_A. Such curves indeed turn out all very similar for every dataset and clearly show that, even though, obviously, repeatability scores increase as the threshold decreases, the ranking between the different LRF algorithms holds completely steady and hence is independent of the actual value chosen for the threshold T_A. Consequently, Fig.7 suggest that the proposed method is the best LRF to be deployed with any descriptor, as it always tends to provide the higher

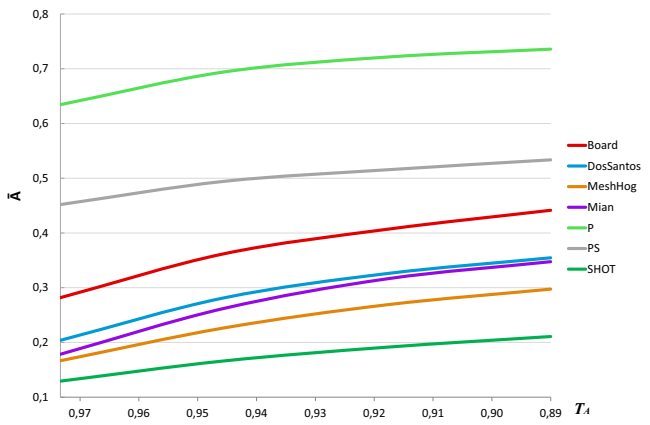


Figure 7. Repeatability scores on *Amphora* dataset as the threshold T_A changes.

percentage of aligned canonical references irrespective to the degree of misalignment tolerated by a specific descriptor.

With regards to computational efficiency, Fig.8 shows the average computation time (in ms) to estimate a LRF as a function of the support radius. *Mian* proves to be the fastest method, even if the differences with *P* and *SHOT* are minimal. Whereas *PS* ranks in the middle, *DosSantos*, *Board* and *MeshHog* prove to be, orderly, the slowest methods. *DosSantos* and *Board* pay for the searching of adjacent points that are used to, respectively, compute the voronoi areas and robustly estimate the normals. PS, instead, is penalized by the estimation of the curve used to fit the plane, whilst *MeshHog* is mostly slowed down by the definition of the support based on the geodesic rather than Euclidean distance. Unlike the results in [12], *Board* turns out slower than *PS*. This is mainly due to the poor scalability of *Board* with respect to the wide radii we consider in our evaluation[5]. As previously discussed, *P* outperforms significantly all the other methods as it can better exploit the distinctiveness emerging from wider supports. Nonetheless, the employment of large radii does not harm its efficiency as the use of a small percentage of points lying at the periphery of the support yields good scalability with respect to the radius size and renders its computation effort comparable to that of the fastest (but dramatically less repeatable) method.

VI. Conclusion and Future Work

Our study highlights that, among evaluated methods, the proposal described in this paper is neatly the state-of-the-art solution for 3D local descriptors deploying a local reference frame. As a matter of fact, our method outperforms other approaches by showing the best repeatability in registration and object recognition scenarios, both with randomly extracted

[5]Furthermore, our current implementation of *PS* is more optimized than that used in [12].

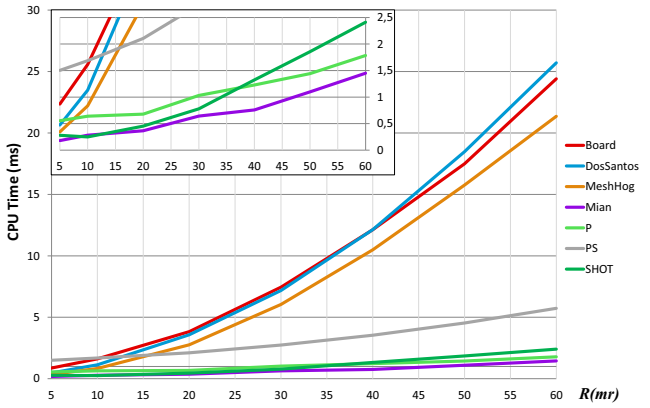


Figure 8. Computation time vs. support radius (*Amphora* dataset, *ISS* keypoints).

features as well as keypoints provided by a state-of-the-art 3D detector. Moreover, the effectiveness of our proposal has been assessed based on two distinct figures of merit. The first is the mean rotation error proposed in the paper that first provided an extensive experimental evaluation of LRF algorithms [12]. The second is a more "application oriented" metric introduced in this paper, which aims at estimating the percentage of correctly aligned LRFs and allows for ranking the algorithms independently of the actual degrees of alignment required by different descriptors.

The novel proposal grants a remarkable percentage of correctly aligned LRFs both on registration as well as object recognition experiments. Moreover, it is interesting to note that every single point correspondence, together with its aligned LRFs, defines the rigid-motion that would allow for aligning two views in a registration task as well as to estimate the pose in an object recognition task. All this consistent data may be sifted out by means of robust estimators, such as e.g. RANSAC or 3D Hough voting [21], in order to end up with the sought transformation. Hence, we are currently investigating on the definition of object recognition and shape registration pipelines that would not require the standard description stage to match features but instead leverage only on the fast and reliable local orientation information provided by the LRF algorithm described in this paper.

REFERENCES

[1] F. Stein and G. Medioni, "Structural indexing: Efficient 3-d object recognition," *PAMI*, vol. 14, no. 2, pp. 125–145, 1992.

[2] C. S. Chua and R. Jarvis, "Point signatures: A new representation for 3d object recognition," *IJCV*, vol. 25, no. 1, pp. 63–85, 1997.

[3] Y. Sun and M. A. Abidi, "Surface matching by 3d point's fingerprint," *ICCV*, vol. 2, pp. 263–269, 2001.

[4] J. Novatnack and K. Nishino, "Scale-dependent/invariant local 3d shape descriptors for fully automatic registration of multiple sets of range images," in *ECCV*, 2008.

[5] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *ECCV*, vol. 3, 2004, pp. 224–237.

[6] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3d object recognition," in *ICCV-WS: 3dRR*, 2009.

[7] A. Mian, M. Bennamoun, and R. Owens, "A novel representation and feature matching algorithm for automatic pairwise registration of range images," *IJCV*, vol. 66, no. 1, pp. 19–40, 2006.

[8] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *ECCV*, vol. 6313, 2010, pp. 356–369.

[9] ——, "Unique shape context for 3d data description," in *3DOR*, 2010, pp. 57–62.

[10] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool, "Hough transform and 3d surf for robust three dimensional classification," in *ECCV*, 2010, pp. 589–602.

[11] A. Zaharescu, E. Boyer, and R. Horaud, "Keypoints and Local Descriptors of Scalar Functions on 2D Manifolds," *IJCV*, 2012.

[12] A.Petrelli and L. Di Stefano, "On the repeatability of the local reference frame for partial shape matching," *ICCV*, pp. 2244–2251, 2011.

[13] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *ICRA*, 2011.

[14] T. R. dos Santos, A. Franz, H.-P. Meinzer, and L. Maier-Hein, "Robust multi-modal surface matching for intra-operative registration," *CBMS*, pp. 1–6, 2011.

[15] A. Mian, M. Bennamoun, and R. Owens, "On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes," *IJCV*, vol. 89, pp. 348–361, 2010.

[16] R. Bro, E. Acar, and T. G. Kolda, "Resolving the sign ambiguity in the singular value decomposition," *J. Chemometrics*, vol. 22, pp. 135–140, 2008.

[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[18] S. Salti, F. Tombari, and L. Di Stefano, "A performance evaluation of 3d keypoint detectors," *IJCV*, 2012.

[19] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *SIGGRAPH*, 1996, pp. 303–312.

[20] J. Smisek, M. Jancosek, and T. Pajdla, "3D with kinect," in *ICCV-WS*, 2011, pp. 1154 –1160.

[21] F. Tombari and L. Di Stefano, "Hough voting for 3d object recognition under occlusion and clutter," *IPSJ Trans, on CVA*, vol. 4, pp. 20–29, 2012.