

# On the repeatability of the local reference frame for partial shape matching

Alioscia Petrelli and Luigi Di Stefano  
CVLab - DEIS, University of Bologna  
Viale Risorgimento, 2 - 40135 Bologna, Italy

alioscia.petrelli@unibo.it, luigi.distefano@unibo.it

## Abstract

We investigate on local reference frames (LRF) deployed with 3D descriptors to achieve invariance to objects' pose. We address the task of matching together partial views of surfaces and propose an experimental study on a large corpus of real data which allows for clearly ranking existing LRF proposals based on their repeatability. Then, drawing inspiration from analysis of the experimental findings, we formulate a new proposal which, in particular, peculiarly includes a procedure aimed at estimating a repeatable LRF also at border features, which is very important when matching partial views of surfaces. Experiments show that the new proposal neatly outperforms existing methods in terms of repeatability, is computationally very efficient and provide relevant benefits in practical applications.

## 1. Introduction

Surface matching deals with the ability of finding similarities between 3D surfaces, often described by triangulated meshes, and is a key task in scenarios such as 3D object recognition and surface registration. Last decade research on surface matching has been mainly focused on local rather than global approaches, for the former being able to withstand nuisances such as clutter and occlusions. Hence, research efforts have addressed the definition of local 3D descriptors, that is compact representations of surface points based on the characteristics of their neighborhood (hereinafter *support*). These representations should be as distinctive and robust as possible, so as to allow for matching corresponding points between surfaces and then estimate aligning rigid body transformations in surface registration or establish upon the presence and pose of a model sought for in 3D object recognition. Invariance to objects' pose is an indispensable trait of every 3D descriptor. Some authors achieve it by using descriptions based on a Reference Axis only. This is the case e.g. of *Spin Images*[6], which builds an histogram by using two cylindrical coordinates, radial distance and elevation, de-

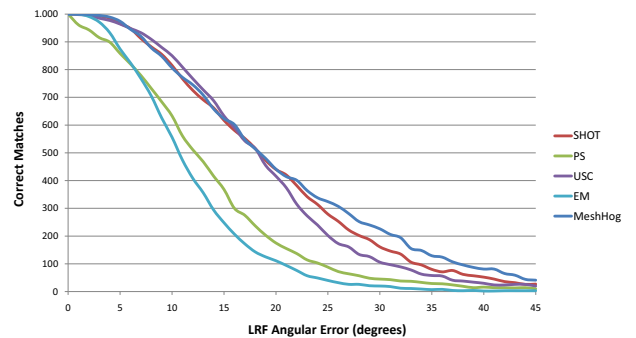


Figure 1. Number of correct matches vs. repeatability of LRF. First, 1000 points are randomly extracted from a 3D surface and described. Then, these features are matched to a second set, which is obtained by considering again the previous surface points and introducing an increasing angular perturbation into the LRFs before computation of descriptors. Perturbations consist of uniformly distributed random rotations around the  $z$  axis (the  $z$  axis is kept fixed since, as discussed in Sec.4, it tends to be highly repeatable). The experiment concerns 5 LRF-descriptor pairs (*SHOT*[15], *PS*[2], *USC*[14], *EM*[11], *MeshHog*[16]).

finned with respect to the surface normal at the point to be described. By disregarding the information on the angular coordinate, *Spin Images* trades distinctiveness for robustness. Differently, other proposals rely on the definition of an invariant *Local Reference Frame* and on the description of the support with respect to such LRF coordinates [12, 2, 13, 11, 4, 17, 9, 15, 14, 7, 16]. As long as the LRF turns out repeatable and robust to noise, the descriptor holds the potential for higher distinctiveness since it can encode all the shape information within the support.

However, descriptors tend to be very sensitive to the degree of misalignment between LRFs at corresponding feature points. Accordingly, as highlighted by Fig.1, the effectiveness of the surface matching process tend to decrease rapidly as the repeatability of the LRF decreases. Since defining a repeatable LRF is a challenging task, several proposals, such as e.g. *Point Signature* [2] and *3D Shape Context* [5], resort to compute multiple descriptions of each

surface point in order to account for different possible rotations of the object. Yet, this approach implies a growth of the computational cost associated with the description process in terms of both execution time as well as memory occupancy. Moreover, the matching stage becomes more ambiguous and significantly slower.

In a recent paper [15], the importance of a unique (i.e. not bound to a multi-description approach) and repeatable LRF for effective and computationally efficient local surface description has been pointed out. The paper presents the first experimental study specifically aimed at assessing the repeatability of the LRFs proposed in literature for local surface description, together with a state-of-the-art proposal for the definition of a unique and repeatable LRF. The experimental evaluation focuses on 3D object recognition, with clutter, occlusion and noise being the addressed nuisances. However, in [15] the repeatability of LRFs is not evaluated in partial shape matching scenarios, which concerns finding corresponding features between different partial 3D views of a given subject. Partial shape matching is required in surface registration, whereby a full 360° reconstruction of an object is built by fusing together partial 3D views taken from different vantage points. Additionally, 3D object recognition may involve matching a full 3D model to a 3D view of a scene or also one or more partial 3D views of a model to a 3D view of the scene.

Therefore, the first contribution of this paper is an extensive benchmark study whose purpose is to analyze and compare the repeatability of LRFs in a partial shape matching scenario, so as to elucidate on the methods most appropriate to this important task which was not addressed by the authors in [15]. Unlike [15], we do not consider synthetic data with injected noise but instead evaluate LRFs on publicly available datasets of partial views acquired with real scanning systems. Moreover, as detailed in Sec.2, we consider a larger set of methods than [15], including, in particular, the LRFs used in [10], that proposed in conjunction with the *MeshHog* descriptor [16] and a variant of the state-of-the-art method presented in [15].

As reported in Sec.4, our experimental evaluation show that LRFs based on principal directions, such as e.g. those proposed in [15] and [10], which prove highly repeatable against the nuisances considered in [15], turn out unsuitable to partial shape matching mainly due to the local point density variations induced by the changes of the vantage point. On the other hand, the LRF associated with the Point Signatures descriptor [2] reveal itself to be the most repeatable.

The second contribution of this paper stems from analysis of the experimental findings reported in Sec.4. More precisely, since our results prove that, for every LRF, the axis directed along the surface normal is always more repeatable than the reference tangent direction, we conceive and propose a novel LRF which requires the computation of surface

normals only. Furthermore, we improve our novel method in order to deal with a problem that is peculiar of partial shape matching and occurs at the borders of 3D views. In fact, supports that are too close to the borders of a view hold missing regions that drastically decrease LRF repeatability.

Experiments run on the same dataset as in Sec.4 allow us to prove that the proposed LRF exhibits the highest repeatability and that its computation time is comparable to that of the fastest -and less repeatable- existing methods.

## 2. Local Reference Frames Overview

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**[10]: the unit vectors of the LRF are given by the normalized eigenvectors of the covariance matrix:

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

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.

**SHOT**[15]: 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:

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

with  $d_i = \|\mathbf{p}_i - \mathbf{p}\|_2$ . To achieve true rotation invariance, a sign disambiguation technique inspired by [1] 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. This procedure is applied to the eigenvectors associated with the largest and smallest eigenvalues, in order to attain the unit vectors defining, respectively, with the  $x$  and  $z$  axes. The third unit vector is computed via the cross-product  $z \times x$ .

**SHOTb**: we replace  $\mathbf{p}$  with  $\hat{\mathbf{p}}$  in (3) and apply the aforementioned sign disambiguation procedure to eigenvectors. This allows us to investigate on whether the use of the barycenter may improve repeatability with respect to the original proposal in [15].

**EM**[11]: the  $z$  axis is given by the surface normal,  $\mathbf{n}$ , at the feature point  $\mathbf{p}$ . To obtain the  $x$  axis, the eigenvector of (1) associated with the largest eigenvalue is projected onto the tangent plane defined by  $\mathbf{n}$ . Then, the third axis is given by  $z \times x$ .

**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 between the two possible unit vectors,  $z^+$  and  $z^-$ , the inner product with  $\mathbf{n}$  is computed for both, so as to chose the unit vector yielding a positive product. 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**[16]<sup>1</sup>: support points,  $\mathbf{p}_i$ , are determined based on the geodesic rather than euclidean distance. The  $z$  axis is given by the surface normal  $\mathbf{n}$ , whereas identification of the  $x$  axis is inspired by *SIFT* [8]. At each  $\mathbf{p}_i$ , the discrete gradient  $\nabla_S f(p_i)$  is computed, function  $f(p_i)$  being the mean surface curvature. Gradient magnitudes are added to a polar histogram of 36 bins (covering  $360^\circ$ ) and weighed by a Gaussian function of the geodesic distance from  $\mathbf{p}$ , with  $\sigma$  equal to half of the average mesh resolution (hereinafter  $mr$ ). To deal with aliasing and quantization, votes are interpolated bilinearly between neighboring bins. While in *SIFT* histogram bins are filled according to gradient orientation, in [16] 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  $z \times x$ .

### 3. Evaluation Methodology

To assess and compare the repeatability of the considered LRFs for the task of partial shape matching, we selected nine datasets taken from two popular repositories: the Stanford 3D Scanning Repository [3] (*Bunny*, *Dragon*, *Armadillo*) and the AIM@SHAPE Repository<sup>2</sup> (*Amphora*, *Buste*, *Dancing Children*, *Glock*, *Neptune*, *Fish*). Since the philosophy of our work is to evaluate LRFs under a variety of real working conditions, datasets are selected so as to address different characteristics of 3D mesh acquisition sys-

tems. First of all, the chosen datasets are acquired with different laser scanners, e.g. the high quality *Cyberware 3030 MS* for the Stanford datasets and the low quality *Minolta vi700* for the *Glock* dataset. That involves different point densities, with very detailed views, such as *Neptune*, *Glock*, *Dancing Children*, as well as significantly coarser acquisitions, i.e. *Bunny*, *Buste* and *Amphora*. Noise is out there in our experiments, for it consists of the real noise injected into the data by the employed acquisition systems. Those considered in our study include very noisy datasets, such as *Glock*, as well as much more accurate and cleaner data, such as *Neptune* and *Amphora*. Finally, we have considered datasets comprising different number of views, e.g. *Glock*, *Bunny* and *Fish* with, respectively, 8, 10 and 10 views, as well as *Dragon* and *Armadillo* with, respectively, 61 and 91 views.

To achieve quantitative evaluation of repeatability, we define a set of indexes. Given a dataset  $D = \{V_1, V_2 \dots V_M\}$  consisting of  $M$  views, we denote as  $VP_n = (V_h, V_k)$  the  $n$ -th view pair attained by considering two different views, with  $N = \frac{M(M-1)}{2}$  being the total number of view pairs within  $D$ . For every  $VP_n = (V_h, V_k)$ , we randomly pick up  $N_f$  (set to 1000 in the experiments) random points  $\mathbf{p}_{i,h}$  from view  $V_h$  and, to each of these features, apply the ground-truth rigid body transformation<sup>3</sup> from  $V_h$  to  $V_k$ . If the distance between the transformed point and the closest point  $\mathbf{p}_{i,k}$  in  $V_k$  is less than  $2.5 * mr$ , we select  $FP_{i,n} = (\mathbf{p}_{i,h}, \mathbf{p}_{i,k})$  as a pair of corresponding features in view pair  $VP_n$ . Should the number of corresponding feature pairs,  $N_{fp}$ , be less than  $0.05 \cdot N_f$ , view pair  $VP_n$  would not be considered further in the experiment. Instead, given enough feature pairs  $FP_{i,n}$ , for each of them we compute the local reference frame at the corresponding points according to the method under evaluation, so as to come up with the local reference frame pair

$$LRF_{i,n} = (LRF(\mathbf{p}_{i,h}), LRF(\mathbf{p}_{i,k})) \quad (4)$$

Then, we calculate a set of indexes based upon the angles between corresponding axes of the two LRFs appearing in (4). In particular, denoted as  $\mathbf{z}(\mathbf{p}_{i,h})$  and  $\mathbf{z}(\mathbf{p}_{i,k})$  the unit vectors defining the  $z$  axis of the two LRFs, we measure the repeatability of the  $z$  axis by the cosine of the angle between unit vectors:

$$Cos(Z)_{i,n} = \mathbf{z}(\mathbf{p}_{i,h}) \cdot \mathbf{z}(\mathbf{p}_{i,k}) \quad (5)$$

Analogously, given the two corresponding LRFs in (4), we measure the repeatability of the  $x$  axis based on unit vectors  $\mathbf{x}(\mathbf{p}_{i,h})$ ,  $\mathbf{x}(\mathbf{p}_{i,k})$ .

Since the third axis can always be computed from the former two and, therefore, explicitly accounting for it is not

<sup>1</sup>The algorithm explained here, and used to run our tests, is publicly available on Zaharescu's website. Indeed, it is the latest version of their method, which is slightly different from the formulation reported in [16].

<sup>2</sup><http://shapes.aim-at-shape.net/>.

<sup>3</sup>Ground-truth transformations are available for all the considered datasets.

necessary, [15] gets one single repeatability index associated with the pair  $LRF_{i,n}$  by averaging the cosine measurements taken for the  $x$  and  $z$  axes. This index, however, may be biased by the correlation between cosine measurements due to the  $x$  and  $z$  axes being orthogonal. To significantly decorrelate  $x$  and  $z$  measurements, we compute the rotation that aligns  $\mathbf{z}(\mathbf{p}_{i,k})$  to  $\mathbf{z}(\mathbf{p}_{i,h})$  and rotate  $\mathbf{x}(\mathbf{p}_{i,k})$  accordingly, so as to get  $\mathbf{x}'(\mathbf{p}_{i,k})$ , which lies in the same plane orthogonal to  $\mathbf{z}(\mathbf{p}_{i,h})$  as  $\mathbf{x}(\mathbf{p}_{i,h})$ . Thus, we can define:

$$\text{Cos}'(X)_{i,n} = \mathbf{x}'(\mathbf{p}_{i,h}) \cdot \mathbf{x}(\mathbf{p}_{i,k}) \quad (6)$$

and then the repeatability index as:

$$\text{MeanCos}'_{i,n} = \frac{\text{Cos}'(X)_{i,n} + \text{Cos}(Z)_{i,n}}{2} \quad (7)$$

The previous quantity is first aggregated over all the local reference pairs belonging to a given view pair

$$\text{MeanCos}'_n = \frac{1}{N_{fp}} \sum_{i=1}^{N_{fp}} \text{MeanCos}'_{i,n} \quad (8)$$

and then averaged across all view pairs, so as to get the final repeatability score associated with the given dataset:

$$\text{MeanCos}' = \frac{1}{N} \sum_{n=1}^N \text{MeanCos}'_n \quad (9)$$

Since, as highlighted in Sec.2, some methods do not deal with the inherent sign ambiguity of principal directions, we also define two indexes aimed at quantifying separately the repeatability of direction and sign. Considering again a LRF pair and, e.g., the  $x$  axis, the two additional indexes are defined as follows:

$$\text{AbsCos}(X)_{i,n} = \left| \text{Cos}(X)_{i,n} \right| \quad (10)$$

$$\text{Sign}(X)_{i,n} = \begin{cases} 1, & \mathbf{x}(\mathbf{p}_{i,h}) \cdot \mathbf{x}(\mathbf{p}_{i,k}) \geq 0 \\ 0, & \mathbf{x}(\mathbf{p}_{i,h}) \cdot \mathbf{x}(\mathbf{p}_{i,k}) < 0 \end{cases} \quad (11)$$

As before, indexes are averaged through LRF pairs and then view pairs to attain global figures of merit associated with the considered dataset. These additional figures will be referred to hereinafter as  $\text{AbsCos}X$ ,  $\text{AbsCos}Z$ ,  $\text{Sign}X$  and  $\text{Sign}Z$ .

#### 4. Evaluation of Existing Methods

All the methods presented in Sec.2 have been evaluated on the nine chosen datasets. For every method and dataset, we have run tests with different values of the support radius  $R$  (i.e.  $5 \cdot mr$ ,  $10 \cdot mr$ ,  $20 \cdot mr$ ). The chart in Fig.2 shows the highest repeatability score yielded by each method on each dataset, the table beneath reporting the corresponding optimum values of  $R$ .

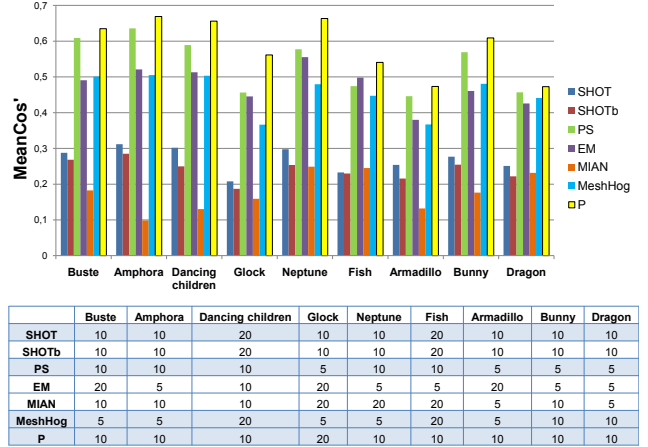


Figure 2. Repeatability scores and support radii (in  $mr$ ) units.

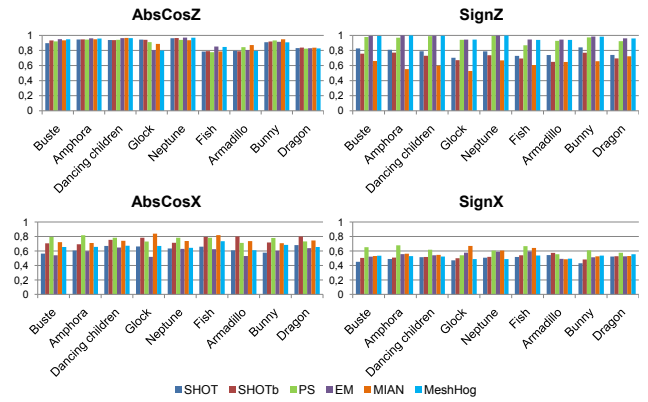


Figure 3. Direction and sign repeatability indexes for the  $z$  and  $x$  axes. Radii are the same as in the table of Fig. 2.

PS exhibits neatly the highest repeatability across the considered datasets, while the methods that fully rely on principal directions, i.e. MIAN, SHOT and SHOTb, turn out notably less effective. Overall, EM and MeshHog rank somewhat in between PS and principal directions methods, though EM even outperform PS on *Fish* and yields comparable performance on *Glock* and *Neptune*.

To better comprehend and compare the behavior of the different methods, we report in Fig.3 the direction and sign repeatability indexes corresponding to the results shown in Fig.2 (i.e. the radii are those maximizing the global repeatability score  $\text{MeanCos}'$ ).

Fig.3 shows how, consistently across the considered methods and as for both direction and sign, estimation of the  $z$  axis is significantly more repeatable than the  $x$  axis. Let us comment on  $z$  direction first: we ascribe the high repeatability shown by the  $\text{AbsCos}Z$  chart to estimation of the  $z$  direction representing, for every LRF, a sort of estimation of the surface normal  $\mathbf{n}$ , which is intrinsically a well defined direction given the nature of surfaces. In fact, it is

well-known that in principal direction methods the eigenvector corresponding to the smallest eigenvalue represents a TLS estimation of the surface normal. As regards PS, the normal to the plane that fits curve  $C$  can be seen as a sort of surface normal, to which, in turn, usually tends to be aligned. EM and MeshHog use directly the surface normal  $\mathbf{n}$  as  $z$  axis of their LRF.

However, as vouched by the *SignZ* chart, between principal direction methods, MIAN exhibit the lowest sign repeatability since it does not attempt any disambiguation of the sign of the eigenvector estimating the normal. This largely accounts for the poorest performance provided by MIAN in terms of overall repeatability (Fig.2), for the good repeatability of the  $z$  direction being wasted due to instability of the sign. The SHOT and SHOTb bars in the *SignZ* chart prove the benefits brought in by sign disambiguation in the estimation of  $z$  based on principal directions, which determines the higher overall repeatability of these two methods with respect to MIAN (Fig.2). However, the best  $z$ -sign disambiguation approach is employed by PS, which is grounded on the normal  $\mathbf{n}$ , and the highest repeatability in terms of both direction and sign of the  $z$ -axis is definitely achieved by those methods, such as EM and MeshHog, that align it to the surface normal  $\mathbf{n}$ .

On the other hand, defining the  $x$  axis deals with finding a well defined direction on the tangent plane, which turns out harder since many surfaces exhibit nearly flat or symmetric regions. Hence, intrinsically, estimating a repeatable direction on the tangent plane is more difficult than estimating the plane itself (i.e. the surface normal). This explains the significantly lower repeatability indexes for the  $x$  axis with respect to the  $z$  axis reported in Fig.3.

As far as the  $x$  direction is concerned, the *AbsCosX* chart indicates that higher repeatability is achieved by PS and those principal directions methods that compute covariances with respect to barycenter, i.e. SHOTb and MIAN. However, PS seems less robust to noise than the latter methods, as suggested by the results on *Glock*, which is particularly noisy within the considered datasets. In SHOT, the choice of computing covariances with respect to the feature point implies a relevant decrease of the repeatability of the  $x$  direction. Similarly, in EM the projection onto the tangent plane of the largest eigenvector renders notably less stable the estimated  $x$  direction.

As for  $x$ -sign repeatability, the *SignX* chart points out that, overall, PS provides for the most effective approach, although, again, the method seems quite sensitive to noise, as vouched by the indexes measured on *Glock*. For principal components methods, it is clear that sign disambiguation on the tangent direction is not as effective as along the surface normal, with MIAN and EM performing now somewhat better than SHOT and SHOTb.

A peculiar issue of partial shape matching which was

not addressed in [15] consists in surfaces being seen by angularly offset vantage points. As a result, between two angularly distant views of a given feature the distribution of points within the support can change significantly, since in each view the portion of the support closer to the camera turns out denser than the farther one. This asymmetry has a significant impact on the repeatability of LRF methods based on principal directions, since these tend to point towards the denser portion of the support. We found that this occurs particularly with non-barycentric methods (i.e. SHOT), while use of the barycenter (i.e. MIAN, SHOTb) tends to mitigate the effect of the asymmetric density variations induced by viewpoint changes. However, such variations have a detrimental impact on sign-disambiguation with both SHOT and SHOTb, since the choice of the sign is guided by the denser portion of the support. Finally, we believe that also MeshHog is affected by the local point density variations issue, since the  $x$  axis orientation relies on a polar histograms which is populated based on the spatial position of the points within the support.

## 5. A Novel LRF Proposal

As shown in Sec. 4, the normal  $\mathbf{n}$  at  $\mathbf{p}$  is quite repeatable and thus represents a good starting point for the direction and sign of the  $z$  axis, even though it may turn out not enough robust with very noisy datasets (see EM and MeshHog on *Glock* in the *AbsCosZ* chart of Fig.3). Conversely, it is difficult to define a repeatable direction on the tangent plane. We also argue that not necessarily the most appropriate support radius for estimation of the  $z$  axis is the same as for the  $x$  axis. Starting from these considerations, we assemble a novel LRF that outperforms PS in terms of repeatability and robustness and requires a significantly less computation time.

In order to estimate robustly the  $z$  direction, we fit a plane to the support points  $\mathbf{p}_i$ . To disambiguate the sign, we compute the average normal,  $\tilde{\mathbf{n}}$ , over support points and then take the inner product between  $\tilde{\mathbf{n}}$  and the two possible unit vectors  $z^+$  and  $z^-$ , so as to chose the unit vector yielding a positive product. Tuning experiments have shown that a support radius  $R_z$  as small as  $5 \cdot mr$  provides the best estimation of the  $z$  axis throughout all the considered datasets.

The basic intuition for estimation of the  $x$  axis consists in trying to rely again on surface normals, since normals prove repeatable. Hence, we find the point  $\mathbf{p}_i$  within a support of radius  $R_x$  showing the largest angle between its normal  $\mathbf{n}_i$  and the previously defined  $z$  axis, then point the  $x$  axis towards such a point. For the sake of robustness, we do not strictly consider  $\mathbf{n}_i$  but instead compute the average normal over a neighborhood of  $\mathbf{p}_i$ . More precisely, given a point  $\mathbf{p}_i$  and denoted as  $\mathcal{A}(\mathbf{p}_i)$  the set of the adjacent points of  $\mathbf{p}_i$

on the mesh, we define

$$\begin{aligned}
 ring_0(\mathbf{p}_i) &= \mathbf{p}_i \\
 ring_1(\mathbf{p}_i) &= ring_0(\mathbf{p}_i) \cup \mathcal{A}(\mathbf{p}_i) \\
 &\vdots \\
 ring_r(\mathbf{p}_i) &= ring_{r-1}(\mathbf{p}_i) \cup \{\mathcal{A}(\mathbf{p}_r) : \mathbf{p}_r \in ring_{r-1}(\mathbf{p}_i)\}
 \end{aligned} \tag{12}$$

Then,  $\mathbf{n}_{r,i}$  denotes the normal at  $\mathbf{p}_i$  obtained by averaging normals over all the points belonging to  $ring_r(\mathbf{p}_i)$ .

Thus, for every point  $\mathbf{p}_i$  within the support we compute the cosine of the angle between  $\mathbf{n}_{r,i}$ <sup>4</sup> and the unit vector defining the  $z$  axis, denoted as  $cos_i$ , in order to select the point  $\mathbf{p}_{min}$  yielding the lowest cosine. The  $x$  axis is then given by the normalized projection onto the tangent plane of the vector from  $\mathbf{p}$  to the selected  $\mathbf{p}_{min}$ . Since  $\mathbf{p}_{min}$  usually lies close to the margin of the support, to speed up the computation we consider only the points  $\mathbf{p}_i$  having distance to  $\mathbf{p}$  greater than  $T_m \times R_x$  (we choose  $T_m = 0.85$ ).

The proposed method, and all those introduced in sec. 2 alike, suffers from a problem peculiar of partial shape matching and highlighted by Fig.4: the support of a feature point located close to the border of a view shows missing surface points that dramatically deteriorate the repeatability of LRF estimation. In particular, Fig.4 depicts two corresponding points extracted from two different views of the surface together with the LRFs yielded by the method described so far herein. For the support in the left view, the method finds  $\mathbf{p}_{min}$  (and therefore the  $x$  axis) in a region that is missing in the support shown in the right view, because in the latter the point is too close to the border. Hence, in the right view the method selects another point, so that two different  $x$  directions are estimated in the two views.

A trivial solution to this problem may consist in discarding feature points close to borders, for instance by rejecting those with border distance lower than the support radius. Unfortunately, in surface registration, as the two views to be matched together are acquired from increasingly offset viewpoints, the deployable features tend to diminish and lie close to the -opposite- borders of the views. Consequently, relatively distant views can hardly be registered unless border features are effectively described and matched.

For this reason, we have improved our method so as to try to estimate a repeatable LRF at border points too. The improvement stems from the observation, which might perhaps seem paradoxical, that the presence of missing points brings in useful information about the surface shape within the support. Indeed, besides self-occlusions, a missing region is due to surface normals being too inclined with respect to the line of sight of the acquisition system, for an acquisition system being able to acquire a surface patch only if the normals are sufficiently parallel to its line of sight.

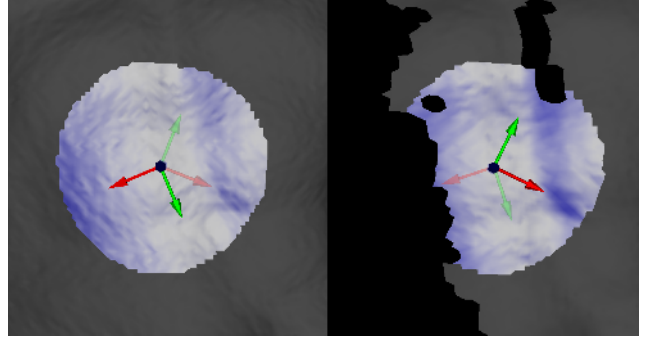


Figure 4. LRFs of two corresponding points extracted from two partial views.  $x$  axes are shown as red arrows,  $y$  axes are in green,  $z$  axes point outward from the image. The behavior of surface normals is illustrated by assigning a darker blue to points having normals more inclined with respect to the  $z$  axis. In both views, the LRF found at the point is opaque whereas the LRF found in the other view is overlaid semi-transparently for comparison.

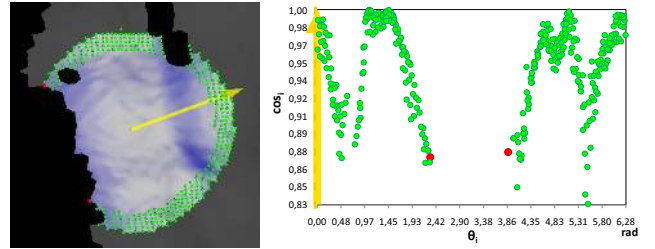


Figure 5. Missing region identification.

Therefore, while the normals of non-missing regions point in the direction of the line of sight, the normals of missing regions do not. Hence it is likely that, had missing regions been acquired, their normals would have been the most inclined within the support with respect to normal at the acquired feature point (e.g., as in Fig.4). Since we define the  $x$  axis based on the point,  $\mathbf{p}_{min}$ , showing the most inclined normal, when there is a missing region we try to assess whether  $\mathbf{p}_{min}$  should be more likely located within the missing region.

The improvement consists of three stages. In the first, we look for missing regions within the support. In the second, we evaluate whether a missing region may include  $\mathbf{p}_{min}$ . In the last one, we try to localize  $\mathbf{p}_{min}$ . More specifically, in the first stage, a random axis lying on the fitted plane is computed (the yellow arrow in Fig.5). For every considered  $\mathbf{p}_i$  (the green points in Fig.5), the vector from  $\mathbf{p}$  to  $\mathbf{p}_i$  is projected onto the fitted plane. The counterclockwise angle  $\theta_i$  between the random axis and this vector is computed. We identify a missing region when  $\Delta\theta$  between two consecutive points along the angular direction is larger than  $2\pi \times T_h$  (we use  $T_h = 0.2$ ).

For every found missing region, the second stage considers the points  $p_a$  and  $p_b$  at the boundaries of the region (the

<sup>4</sup> $r = 2$  in chosen our experiments

$R_z$	$r$	$T_m$	$T_h$	$T_S$
$5 \cdot mr$	2	0.85	0.2	0.1

Table 1. Parameters of the proposed method.

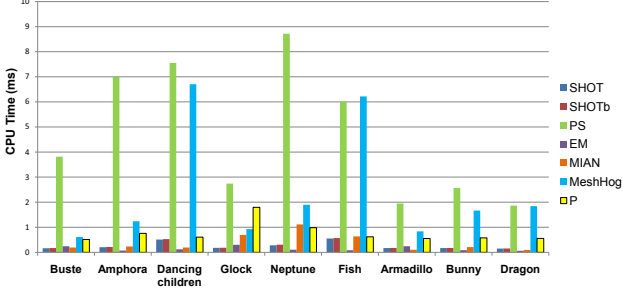


Figure 6. Computation times for all methods.

two red points in Fig.5) and uses them to quickly estimate if the region contains  $\mathbf{p}_{min}$ . We assume that a missing region contains  $\mathbf{p}_{min}$  if  $\mathbf{n}_{r,a}$  and  $\mathbf{n}_{r,b}$  are sufficiently inclined with respect to the others  $\mathbf{n}_{r,i}$ , and therefore if

$$S = (|\cos_a| + |\cos_b|)/2 \quad (13)$$

is greater than  $T_S$  (our tuning tests show that  $T_S = 0.1$  is the best choice), where  $\cos_a$  and  $\cos_b$  are normalized in the range  $[0,1]$  by:

$$|\cos_i| = 1.0 - \frac{\cos_i - \min(\cos_i)}{1.0 - \min(\cos_i)} \quad (14)$$

If a support contains several missing regions, we consider only that with the greatest  $S$ . The last stage aims at finding an angle  $\theta_t$  between  $\theta_a$  and  $\theta_b$  that estimates the position of  $\mathbf{p}_{min}$ . Intuitively,  $\theta_t$  is closer to  $\theta_a$  if  $|\cos_a|$  is greater than  $|\cos_b|$  and vice versa. Hence, formally:

$$t = \frac{(|\cos_b| - |\cos_a|) + 1.0}{2}; t \in [0, 1] \quad (15)$$

$$\theta_t = \theta_a + (\theta_b - \theta_a)t \quad (16)$$

Finally, the  $x$  axis is obtained by rotating the random axis by  $\theta_t$  around the  $z$  axis. Table 1 summarizes the parameters of our method together with the suggested default values resulting from the tuning process.

## 6. Evaluation of the Novel Proposal

The chart in Fig.2 shows that our proposal (P, yellow bar) consistently outperforms other methods throughout all datasets, including the very noisy *Glock* where, in particular, our approach turns out significantly more robust. As for the support radius, similarly to Sec. 4, we have tested our proposal with  $R_x$  equal to  $5 \cdot mr$ ,  $10 \cdot mr$  and  $20 \cdot mr$  and, for each dataset, chosen the value yielding the highest repeatability (see the table in Fig.2). It is worth pointing out, that,

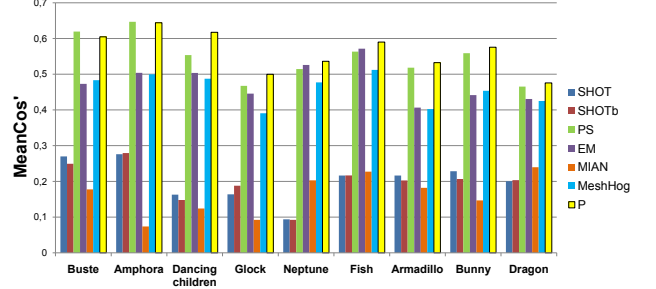


Figure 7. Repeatability scores on *MeshDog* keypoints.

compared to other methods, our proposal seem to be notably more stable with respect to the choice of the support radius across data gathered in different working conditions.

Fig.6 reports, for all methods and datasets, the measured average computation time (in *ms*) spent to estimate the LRF at a feature point. The chart indicates that PS is definitely, and significantly, the slowest method while our proposal yields computational times comparable to faster methods (i.e. SHOT, SHOTb, MIAN and EM).

In surface matching applications, the issue of nearly flat or symmetric regions may be dealt with to a certain extent by using a suitable feature detector, so as to describe distinctive points only. Yet, to avoid biasing the evaluation towards the features found by a particular type of detector, we have chosen to randomly pick-up feature points. This might, however, introduce a smoothness bias given that many features will likely be picked up within nearly flat or symmetric regions. Therefore, to complement the evaluation, we have run an additional set of experiments on *MeshDog* keypoints<sup>5</sup> [16]. The results in Fig.7 show that, overall, our proposal turns out again, neatly, the most repeatable LRF and that the evaluation on keypoints is consistent with the ranking between existing method discussed in Sec.4 (i.e. PS best method, then EM-MeshHog, principal directions approaches notably less effective).

Finally, to provide an indication of the solid practical benefits which can be brought in by the use of a good LRF estimation algorithm with respect to a less repeatable method, we propose a qualitative experiment. In particular, following the automatic 3D reconstruction pipeline detailed in [11], we try to register and fuse together 18 partial 3D views (Fig.8a) of an object acquired by a *Space-time Stereo* set-up. To this purpose, we use the SHOT LRF and the SHOT descriptor to match randomly extracted feature points between the views. This would provide a coarse registration, to be then handed over to a standard tool (i.e. *Scanalyze*) in order to end-up with the final 3D reconstruction. However, due to the difficulty of this registration task,

<sup>5</sup>Starting from  $5 * mr$ , we have used 3 scale space levels and determined the support radius  $R$  for LRFs based on the characteristic scale provided by the detector.



Figure 8. 3D reconstruction based on the SHOT descriptor: (a) initial views. (b) coarse registration with SHOT LRF (c) coarse registration and (d) reconstruction with the proposed LRF (e) coarse registration and (f) reconstruction with PS LRF. A 3D visualization of the results is provided in the supplementary material.

to be ascribed to the presence of large symmetric regions on object’s surface, the coarse registration turns out so inaccurate (Fig.8b) that *Scanalyze* cannot perform the final reconstruction. We then simply use the SHOT descriptor with the LRF proposed in this paper and, thanks to its much higher repeatability, succeed in achieving a very good coarse registration (Fig.8c) that allows *Scanalyze* to provide an accurate 3D reconstruction (Fig.8d). Moreover, we also try the SHOT descriptor with the PS LRF: the coarse registration (Fig.8e) is not as accurate as to permit a final reconstruction without artifacts (highlighted by green circles in Fig.8f).

## 7. Final Remarks

The proposed approach yields state-of-the-art performance among LRF estimation algorithms. It turns out very repeatable and robust to noise, and seems also more easily tunable as regards the main parameter, i.e. the radius of the support used to define the reference axis on the tangent plane. As for the other parameters, although we have not carried out so far a formal sensitivity analysis, we are confident that the suggested ones provide reasonable default values, since they proved effective across a large set of data acquired by different types of systems and under different working conditions. Another future direction of investiga-

tion concerns analysis of LRFs repeatability with respect to imprecise feature localization, which is an important practical issue not dealt with in our present work. Nonetheless, the proposed 3D reconstruction experiment provides indications of a remarkable robustness of the proposed method, since feature points are extracted by means of the -by far- less precise detection method, i.e. they are picked up randomly from each view.

## References

- [1] R. Bro, E. Acar, and T. Kolda. Resolving the sign ambiguity in the singular value decomposition. *J. Chemometrics*, 22:135–140, 2008.
- [2] C. S. Chua and R. Jarvis. Point signatures: A new representation for 3d object recognition. *IJCV*, 25(1):63–85, 1997.
- [3] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, pages 303–312, 1996.
- [4] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik. Recognizing objects in range data using regional point descriptors. In *ECCV*, volume 3, pages 224–237, 2004.
- [5] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik. Recognizing objects in range data using regional point descriptors. In *ECCV*, volume 3, pages 224–237, 2004.
- [6] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *PAMI*, 21(5):433–449, 1999.
- [7] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. V. Gool. Hough transform and 3d surf for robust three dimensional classification. In *ECCV*, pages 589–602, 2010.
- [8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.
- [9] A. Mian, M. Bennamoun, and R. Owens. A novel representation and feature matching algorithm for automatic pairwise registration of range images. *IJCV*, 66(1):19–40, 2006.
- [10] 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*, 89:348–361, 2010.
- [11] 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.
- [12] F. Stein and G. Medioni. Structural indexing: Efficient 3-d object recognition. *PAMI*, 14(2):125–145, 1992.
- [13] Y. Sun and M. A. Abidi. Surface matching by 3d point’s fingerprint. *ICCV*, 2:263–269, 2001.
- [14] F. Tombari, S. Salti, and L. D. Stefano. Unique shape context for 3d data description. In *3DOR*, pages 57–62, 2010.
- [15] F. Tombari, S. Salti, and L. D. Stefano. Unique signatures of histograms for local surface description. In *ECCV*, volume 6313, pages 356–369, 2010.
- [16] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud. Surface feature detection and description with applications to mesh matching. In *CVPR*, pages 373–380, 2009.
- [17] Y. Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *ICCV-WS: 3dRRR*, 2009.