CoSPAIR: Colored Histograms of Spatial Concentric Surflet-Pairs for 3D Object Recognition

K. Berker Logoglu[a], Sinan Kalkan[b], Alptekin Temizel[a]

[a] *Informatics Institute, Middle East Technical University, Ankara, Turkey*
[b] *Department of Computer Engineering, Middle East Technical University, Ankara, Turkey*

Abstract

Introduction of RGB-D sensors together with the efforts on open-source point-cloud processing tools boosted research in both computer vision and robotics. One of the key areas which have drawn particular attention is object recognition since it is one of the crucial steps for various applications. In this paper, two spatially enhanced local 3D descriptors are proposed for object recognition tasks: *Histograms of Spatial Concentric Surflet-Pairs* (SPAIR) and *Colored SPAIR* (CoSPAIR). The proposed descriptors are compared against the state-of-the-art local 3D descriptors that are available in Point Cloud Library (PCL) and their object recognition performances are evaluated on several publicly available datasets. The experiments demonstrate that the proposed CoSPAIR descriptor outperforms the state-of-the-art descriptors in both category-level and instance-level recognition tasks. The performance gains are observed to be up to 9.9 percentage points for category-level recognition and 16.49 percentage points for instance-level recognition over the second-best performing descriptor.

Keywords: 3D descriptors, 3D object recognition, Point clouds, RGB-D

1. Introduction

Object recognition is an important problem in computer vision and robotics. With the introduction of RGB-D sensors, there have been a boost in the performance of many applications in these fields. The main catalyst of this boost is that the RGB-D sensors, among which Microsoft's Kinect and Asus's Xtion are very popular, allow capturing depth and color information at the same time. Combined together, these two types of complementary information provide the necessary rich source of 2D and 3D information for recognizing objects, or activities.

The performance of object recognition is directly dependent on the descriptors used, and there have been tremendous effort in developing 3D descriptors. Many global and local descriptors, based on the size of the support with respect to a key point, have been proposed in the literature [1]. The *global* descriptors in the literature are either histogram-based (e.g., [2, 3, 4, 5]), transform-based (e.g., [6]), 2D view-based (e.g., [7, 8]) or graph-based [9], whereas local descriptors are histogram-based (e.g., [10, 11, 12]), signature-based (e.g., [13]) or a hybrid of these (e.g., [14, 15, 16]).

Among these aforementioned descriptors, only a few utilize shape and texture/color information together to take advantage of the data obtained from the RGB-D sensors; the MeshHOG proposed by Zaharescu et al. [14], the colored version of the Point Feature Histograms (PFH), called PFHRGB [11, 17], and the color/texture enhanced version of Signature of Histograms of Orientations (SHOT), called CSHOT, proposed by Tombari et al. [16].

Object recognition can be performed at two different levels: at the category level or the instance level. In *category-level* object recognition, an object is classified into pre-defined categories such as *cereal box* or *soda can*, whereas in *instance-level* recognition, specific instances of the objects such as "Cheerios" or "Pepsi can" are recognized. While promising results have been reported for category-level object recognition, instance-level recognition remains a more challenging problem [18, 19, 20].

In this article, we propose two novel local 3D descriptors: The first one is *Histograms of Spatial Concentric Surflet-Pairs* (SPAIR), which is a shape-only descriptor suitable for category-level recognition. The second descriptor, *Color-SPAIR* (CoSPAIR), extends and enhances SPAIR with the color information. By taking advantage of color information from the RGB-D sensor data, CoSPAIR is particularly suitable for instance-level recognition. In both descriptors, the support-radius is divided into regions from which histograms of 3D relations are accumulated. We have compared SPAIR and CoSPAIR against the state-of-the-art descriptors available in the Point Cloud Library on three publicly available datasets: namely, the RGB-D Object dataset [18], the recently introduced BigBIRD dataset [20] and the object scans used in the Amazon Picking Challenge at ICRA 2015 [21].

Email addresses: berkerlogoglu@gmail.com (K. Berker Logoglu), skalkan@ceng.metu.edu.tr (Sinan Kalkan), atemizel@metu.edu.tr (Alptekin Temizel)

http://dx.doi.org/10.1016/j.robot.2015.09.027 - Accepted for publication in Robotics and Autonomous Systems October 9, 2015

2. Related Work

A comparative evaluation of 3D descriptors available in Point Cloud Library (PCL) [17] was presented by Alexandre [19]. According to this analysis, CSHOT [16] and PFHRGB [17] which use color information in addition to shape, are the best performing descriptors, followed by the shape-only SHOT [15, 22], PFH [11] and FPFH [19]. It was also shown that PFHRGB and CSHOT are the best performing descriptors for object recognition using RGB-D data [22]. Due to their prominence, these local 3D descriptors are detailed and compared with the descriptors proposed in this article. Furthermore, it should be noted that the proposed descriptors are based on surflet-pair relations [23] similar to PFH, PFHRGB and FPFH; therefore, these descriptors are further detailed for the sake of completeness and clarity.

2.1. Point Feature Histograms (PFH)

Point Feature Histograms (PFH) was introduced by Rusu et al. in 2008 as a local descriptor for searching correspondences in 3D point clouds [11]. It is a pose-invariant feature based on geometrical relations of a point's nearest k-neighbours. The geometrical relations are computed from relative orientations of surface normals between point pairs. The main steps for computing a PFH descriptor are:

- For each point $\mathbf{p}$ at which a descriptor is to be extracted, the k-neighbouring points within a sphere of a radius r are selected.
- For every pair of points in the sphere, 3 surflet-pair-relation features [23] are calculated (although there are 4 features defined in [23], the fourth feature, the distance between the pairs, is not used since it changes with the viewpoint).
- Histograms of the relations are calculated. Each of the 3-relations is summarized into a 5-bin histogram, and their joint-histogramming yields 5^3 bins in total.

Since PFH considers surflet-pair-relations for every pair of points inside a sphere with radius r, the computational complexity is $O(k^2)$. In other words, for dense point clouds, the time required for extracting PFH descriptors is prohibitively high for practical applications [12, 19, 22].

2.2. Colored Point Feature Histogram (PFHRGB)

PFHRGB is an extension of PFH with color information. It includes three more histograms in addition to those in PFH. These additional histograms represent the ratio between color channels of point pairs, thus bringing the total size of the descriptor to 250 [17]. Adding color information has been shown to increase the performance of PFH [19] but PFHRGB suffers from the same drawback as PFH, i.e., being computationally expensive.

2.3. Fast Point Feature Histograms (FPFH)

Fast Point Feature Histograms is an improvement over PFH for reducing the computational complexity down to $O(k)$ from $O(k^2)$ [12]. This is achieved by generating the histograms from the relations between only a point and its k-neighbouring points inside the support radius r, instead of analyzing relations between all pairs in the sphere. This is called *Simplified Point Feature Histogram* (SPFH). To re-compensate for the missing connections (compared to PFH where all the point-pairs contribute to the descriptor), the SPFHs extracted at the neighbours of a point $\mathbf{p}$ are weighted and summed according to their spatial distance:

$$FPFH(\mathbf{p}) = SPFH(\mathbf{p}) + \frac{1}{k}\sum_{i=1}^{k}\frac{1}{w_i}\cdot SPFH(\mathbf{p}_i), \quad (1)$$

where the weight w_i represents the distance between source / query point $\mathbf{p}$ and a neighbour point $\mathbf{p}_i$. It should be noted that SPFH values should be calculated for all the points in the dataset and the effective radius implicitly becomes $2r$ since additional point pairs outside the r radius are also included in FPFH. Although being significantly faster than PFH and PFHRGB [19], FPFH was shown to be an order of magnitude slower than its alternatives, e.g., SHOT [22]. Moreover, FPFH lacks color information.

2.4. Signature of Histograms of Orientations (SHOT)

Signature of Histograms of Orientations (SHOT) was introduced by Tombari et al. [15, 22]. For extracting a SHOT descriptor, first, a robust, unique and repeatable 3D Local Reference Frame (LRF) is calculated around the source/query point. Then, a spherical grid that consists 32 volume segments (8 divisions along the azimuth, 2 along the elevation, and 2 along the radius) is centered at the point. For each of these volume segments, histogram of the angle between the normal of the source/query point and the points inside the segment is calculated. Finally, all the 32 histograms are concatenated to create the descriptor. SHOT descriptors have been shown to be rotation invariant and robust to noise [15, 22].

2.5. Color-SHOT (CSHOT)

Color-SHOT (CSHOT) combines shape information extracted by SHOT with a texture signature [16] in order to incorporate the color information. To extract texture, the $L_1 - norm$ of the color triplets are binned into histograms. For this, CIELab color space was chosen over RGB since it is perceptually more uniform than the RGB space. CSHOT has been reported to perform better than SHOT due to the extra color information [19, 22].

3. The Proposed Descriptors: SPAIR and CoSPAIR

In this paper, two new descriptors are proposed. The first one utilizes only shape information and is called *Histograms of Spatial Concentric Surflet-Pairs*, whereas the

second one utilizes shape and color information together and is called *Colored Histograms of Spatial Concentric Surflet-Pairs*.

3.1. Histograms of Spatial Concentric Surflet-Pairs (SPAIR)

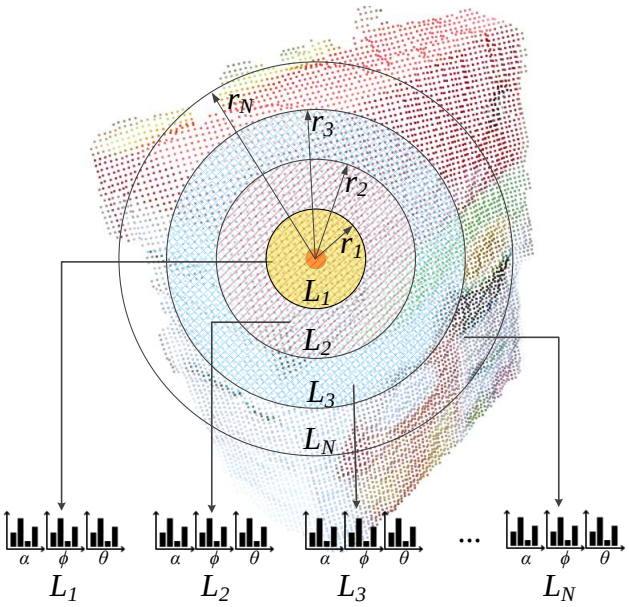


Figure 1: Concentric spherical regions and stitching of the histograms to construct SPAIR descriptor. [Best viewed in color]

Histograms of Spatial Concentric Surflet-Pairs (SPAIR) is based on *surflet-pair-relations* similar to PFH and FPFH where a *surflet* is defined as an oriented surface point, and *surflet-pair-relations* as geometric relations between two *surflets* by Wahl et al. [23].

As described in Section 2.3, Rusu et al. used a method called *Simplified Point Feature Histogram* (SPFH) that relies on the comparison of source/query point/surflet with only the direct k-neighbours inside a sphere (not all the pairs). Furthermore, in order to add spatial information, a special weighting scheme was used in FPFH as formulated in Equation 1.

With SPAIR, we aimed for a simpler thus faster method which requires less point-pair comparisons while adding more spatial information by encoding the geometrical properties of a point's neighbourhood according to distance from the point.

As shown in Figure 1, in our approach, the support radius r is divided into N equal size $(r_1, r_2, ..., r_N)$ regions. The resulting 3D grid can be visualized as N concentric spheres. For each distinct spherical shell (i.e., the region between two adjacent spheres), which we name as a *level* $(L_1, L_2, ..., L_N)$, the surflet-pair-relations between the points inside a level and the source/query point (see Figure 2) are calculated as follows [12, 23]:

- Let $\mathbf{p}_s$ be the source/query point that SPAIR is to be extracted for, $\mathbf{p}_t$ be one of the target points inside a *level* and $\vec{n}_s$, $\vec{n}_t$ the corresponding normals.

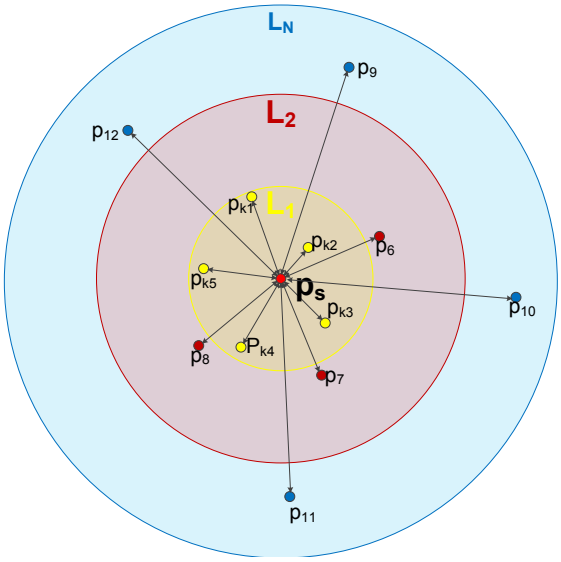


Figure 2: Influence region diagram for SPAIR/CoSPAIR. [Best viewed in color]

- A fixed reference coordinate uvw frame is defined as shown in Figure 3, following [3]:

$$\vec{u} = \vec{n}_s, \tag{2}$$
$$\vec{v} = (\mathbf{p}_t - \mathbf{p}_s) \times \vec{u}, \tag{3}$$
$$\vec{w} = \vec{u} \times \vec{v}. \tag{4}$$

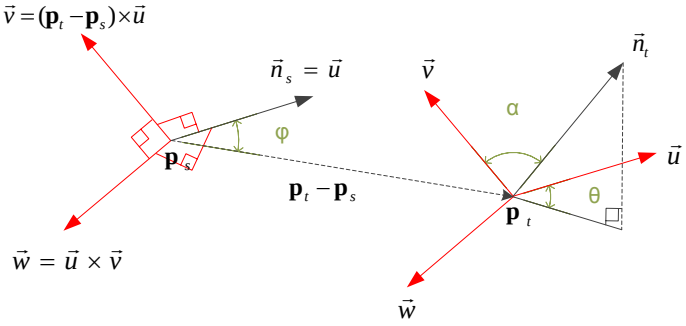


Figure 3: The reference coordinate uvw frame and the angular relations between surflets (adapted from [3])

- Using the reference frame defined above, the angular relations between surflets are calculated as follows:

$$\alpha = \vec{v} \cdot \vec{n}_t, \tag{5}$$
$$\phi = \frac{\vec{u} \cdot (\mathbf{p}_t - \mathbf{p}_s)}{\|\mathbf{p}_t - \mathbf{p}_s\|}, \tag{6}$$
$$\theta = \arctan\left(\vec{w} \cdot \vec{n}_t, \vec{u} \cdot \vec{n}_t\right), \tag{7}$$

where $\alpha \in [-1, 1]$ represents $\vec{n}_t$ as the cosine of a polar angle, $\phi \in [-1, 1]$ is the direction of the translation from $\mathbf{p}_s$ to $\mathbf{p}_t$, $\theta \in [-\pi, \pi]$ corresponds to $\vec{n}_t$ as an azimuthal angle.

Then, the three values for the angles (α, ϕ, θ) in Equations 5, 6, 7 are binned into separate histograms:

$$H^l_\alpha(b) = \sum_{\mathbf{p}_t} \delta\left(\left\lfloor \frac{1}{2}\alpha(\mathbf{p}_t, \mathbf{p}_s)B \right\rfloor - b\right), \quad (8)$$

$$H^l_\phi(b) = \sum_{\mathbf{p}_t} \delta\left(\left\lfloor \frac{1}{2}\phi(\mathbf{p}_t, \mathbf{p}_s)B \right\rfloor - b\right), \quad (9)$$

$$H^l_\theta(b) = \sum_{\mathbf{p}_t} \delta\left(\left\lfloor \frac{1}{2\pi}\theta(\mathbf{p}_t, \mathbf{p}_s)B \right\rfloor - b\right), \quad (10)$$

where l is the level for which the histogram is being computed, $\delta()$ is the Kronecker delta function, b is the bin index of a histogram, and B is the total number of bins. When calculations are finalized for all the defined surflet-pairs, the histograms H^l_α, H^l_ϕ and H^l_θ are normalized using the number of distinct points in each *level*:

$$\hat{H}^l_\alpha(b) = \frac{1}{C^l} H^l_\alpha(b), \quad (11)$$

$$\hat{H}^l_\phi(b) = \frac{1}{C^l} H^l_\phi(b), \quad (12)$$

$$\hat{H}^l_\theta(b) = \frac{1}{C^l} H^l_\theta(b), \quad (13)$$

where C^l is the number of points in *level* l.

The resulting SPAIR descriptor $\mathbf{v}_{SPAIR}$ is the concatenation of all the histograms in an order based on their distances to the center point:

$$\mathbf{v}_{SPAIR} = \hat{H}^0_\alpha \oplus \hat{H}^0_\phi \oplus \hat{H}^0_\theta \oplus ... \hat{H}^N_\alpha \oplus \hat{H}^N_\phi \oplus \hat{H}^N_\theta, \quad (14)$$

where $\oplus$ denotes concatenation. Figure 1 illustrates the levels inside the concentric sphere borders and stitching of the corresponding histograms.

3.2. Colored Histograms of Spatial Concentric Surflet-Pairs (CoSPAIR)

It has been reported that adding color/texture information improves the performance of various descriptors considerably [18, 19, 22, 24].With this motivation, we update SPAIR such that it encodes color as well as shape, and call it *Colored Histograms of Spatial Concentric Surflet-Pairs* (CoSPAIR).

In CoSPAIR, color/texture and shape information is encoded at each level of the SPAIR descriptor as shown in Figure 4. In our experiments, three different color spaces; RGB, HSV and CIELab have been tested. Additionally, for each color space, two variants have been evaluated: (i) Using simple color histogram of each color channel. (ii) Using histogram of $L_1 - norm$ of point pairs for each color channel. Our experiments (see Table 1) indicated that the best results are obtained by using simple color histograms in the CIELab color space for each channel at each level. This resulted in a descriptor that has 3 sub-features for both shape and color for each *level*:

$$\mathbf{v}_{CoSPAIR} = \hat{H}^0_\alpha \oplus \hat{H}^0_\phi \oplus \hat{H}^0_\theta \oplus \hat{H}^0_{\mathbf{L}} \oplus \hat{H}^0_{\mathbf{a}} \oplus \hat{H}^0_{\mathbf{b}} \oplus ... \hat{H}^N_\alpha \oplus \hat{H}^N_\phi \oplus \hat{H}^N_\theta \oplus \hat{H}^N_{\mathbf{L}} \oplus \hat{H}^N_{\mathbf{a}} \oplus \hat{H}^N_{\mathbf{b}}. \quad (15)$$

where $\oplus$ denotes concatenation and **L**, **a**, **b** denotes the CIELab color components.

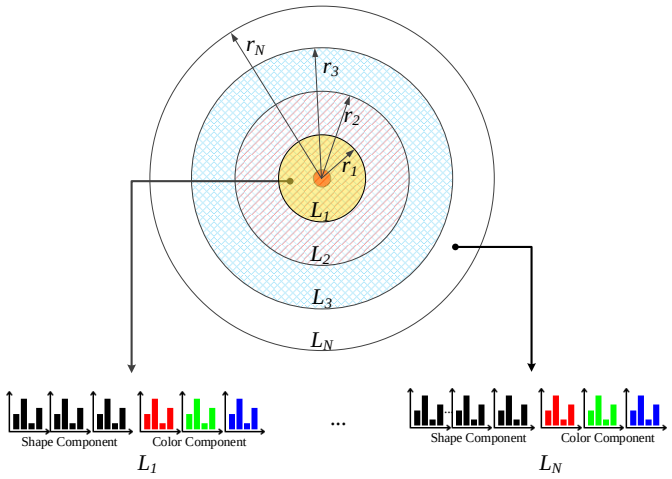


Figure 4: Concentric spherical regions and the stitching of shape and color histograms for the extraction of CoSPAIR. [Best viewed in color]

Table 1: Average accuracy results for the evaluated color contributions. The tests are conducted in Dataset 1 (see Section 4.1.1).

	Category Level	Instance Level
RGB	93.63	81.76
RGB-L_1	91.74	82.64
HSV	91.40	76.31
HSV-L_1	86.46	64.61
CIELab	**94.34**	**83.10**
CIELab-L_1	86.25	64.23

4. Experiments and Results

We compare the proposed descriptors against the state-of-the-art local 3D descriptors that are publicly available in the Point Cloud Library (PCL) [17]: PFH [11], PFHRGB [17], FPFH [12], SHOT [15, 22] and CSHOT [16]. The same testing procedure, which is summarized in Figure 5, is used for evaluating the descriptors.

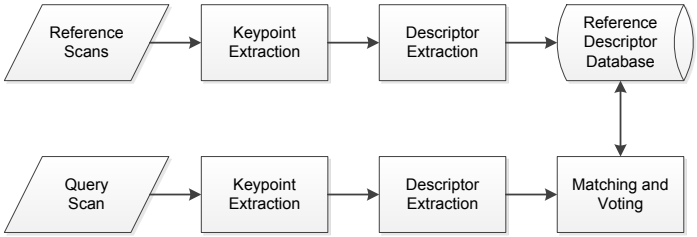


Figure 5: Test pipeline.

For all the conducted tests/experiments, the surface normals are estimated with a search radius of 1 cm as in [19]. Then, the datasets used in the tests are split into a

query set and a reference set depending on the test scenario. In this paper, two different scenarios that are proposed in [18] are used:

1. *Leave-sequence-out*: Test and train sets are chosen to be from scans with different camera heights.
2. *Alternating contiguous frames*: The video sequences from different heights are divided into 3 contiguous sequences of equal length. Since there are 3 heights (videos) for each object in the datasets used, this gives 9 video sequences for each object. 7 of these are randomly selected for training and the remaining 2 for test. 10 trials are performed and the results are averaged.

At the matching phase, the query descriptors are brute-force matched to the nearest descriptor in the reference descriptor database (see Figure 5) using Euclidean norm ($L^2 - norm$) and the final decision is made via a majority rule [25] as follows:

$$D(X) = \arg\max_{C} \sum_{i=1}^{K} I(f_i(X) = C), \tag{16}$$

where C is the class label, X is the object to be classified, f is a keypoint, K is the total number of keypoints on the query object and D is the final decision. For the *Matching and Voting* stage, OpenCV library [26] is used whereas for all the remaining stages, Point Cloud Library [17] is used. The performance of the descriptors are calculated as *average accuracy*, the average per-class effectiveness [27]:

$$\frac{1}{L} \sum_{i=1}^{L} \frac{TP_i + TN_i}{TP_i + FP_i + FN_i + TN_i}, \tag{17}$$

where L is the total number of class labels and TP, TN, FP, FN are *true positives*, *true negatives*, *false positives* and *false negatives*, respectively. For extracting/detecting the keypoints, we have chosen the Intrinsic Shape Signatures 3D (ISS3D) method [28], which is available in the PCL library. ISS3D has recently been shown to be among the top performing methods and it was reported to stand out for its performance, repeatability and efficiency [29, 30]. Our experiments also confirm these findings as detailed in Section 4.3.

4.1. The Datasets

The experiments were conducted on three different object recognition datasets in four configurations. The first dataset is the well known RGB-D Object Dataset introduced by Lai et al. in 2011 [18]. This dataset was used in two different configurations: the first configuration is a subset that had been used by Luis A. Alexandre [19]. This subset is used for optimization and comprehensive analysis. The second configuration of this dataset consists of all the objects and were used for complementary analysis. The second dataset is the recently introduced BigBIRD ((Big) Berkeley Instance Recognition Dataset) by Singh et al. [20]. The third dataset is the object scans used in the Amazon Picking Challenge at ICRA 2015 [21].

4.1.1. Dataset 1: Subset of the RGB-D Object Dataset

The RGB-D Object Dataset [18] consists of 300 common household objects in 51 categories. The objects were scanned with an RGB-D camera with 640×480 resolution from different angles and the total number of RGB-D images are around $250,000$.

As a first step in our experiments, a subset of this large dataset which contains 48 objects in 10 categories is chosen. The chosen subset was used by Luis A. Alexandre in a comprehensive evaluation of various descriptors that are available in PCL [19] and it contains the following categories: apple, ball, banana, bell pepper, binder, bowl, calculator, camera, cap and cell phone. Examples of segmented scans for each category are given in Figure 6.

In this subset, a total of 1421 point clouds are chosen as in [19]. The *leave-sequence-out* and *alternating contiguous frames* scenarios are applied for both category and instance-level recognition experiments. As in [18] and [19], for *leave-sequence-out*, in the query set, the camera is mounted 45° above the horizontal axis relative to the turntable whereas in the reference set it is mounted 30° and 60° above. We refer to [31] for more details on the setup and query scans.

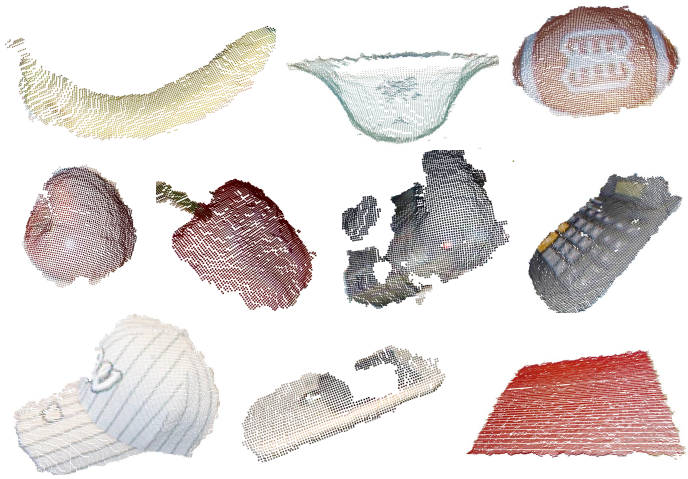


Figure 6: Examples of point clouds from the chosen 10 category subset of the RGB-D Object Dataset [18].

4.1.2. Dataset 2: RGB-D Object Dataset - All Objects

As our second dataset, the RGB-D Object Dataset with all 300 objects in 51 categories is used. Since the total number of images in the dataset as well as the number of scans per object is high, the scans in azimuth are subsampled by taking every twentieth sample. This yielded an average of 10 scans for each object for each video sequence (whole rotation on the turntable) from different

camera heights resulting a total of 9944 point clouds for test and training in total.

As in Dataset 1, for the *leave-sequence-out* scenario, the camera positions are chosen as 45° for the query set and 30° and 60° above the horizontal axis of the turntable for the reference set.

4.1.3. Dataset 3: BigBIRD Dataset

BigBIRD is a recently introduced instance-level object recognition dataset introduced by Singh et al. [20] which is publicly available [32]. The RGB-D data was collected by a Carmine 1.09 sensor. The resolution of the RGB-D scans is the same as in Dataset 1, i.e., 640×480. The initial version of the dataset contains 100 objects and the dataset is being updated. At the time the tests were being performed, the dataset included a total of 123 objects. However, in our tests, we excluded the transparent objects[1] due to their poor quality point clouds, as also stated in [20]. With the removal of the transparent objects, the resulting dataset contains 105 different objects.

BigBIRD is a very challenging dataset due to the extreme similarity between object instances. Not only many objects are similar in shape and size, but also product varieties of the same brand are labeled as different object instances - see Figure 7 for some samples.

Figure 7: Sample RGB images from the BigBIRD dataset [20], each from different object instance.

In the BigBIRD dataset, the objects were scanned from 5 different polar angles and 120 azimuthal angles with a total of 600 images and point clouds per instance. The polar angles are named as `NP1`, `NP2`,...,`NP5` where `NP1` corresponds to a position where the sensors are located 0° with respect to the horizontal axis of the turntable, `NP5` corresponds to 90° and `NP2`, `NP3`, `NP4` located on a quarter circular arc in between [21]. In our experiments, for both test scenarios, we have used the poses similar to the experiments in the previous datasets. We have chosen the data obtained from positions `NP2`,`NP3` and `NP4` and for *leave-sequence-out* scenario, we have used `NP3` for the query and `NP2` and `NP4` for the reference sets. Additionally, not all azimuthal scans are used. The scans are sub-sampled by taking every tenth, resulting in approximately 12 scans per object. With the chosen views and sub-sampling of scans, a total of 3746 point clouds are used in experiments.

4.1.4. Dataset 4: The Amazon Picking Challenge Dataset

The dataset was collected for the first Amazon Picking Challenge at ICRA 2015 using the same system setup as in the BigBIRD Dataset [20], [33] and is publicly available [21]. The dataset is composed of 26 different objects. Although some of the objects such as `safety works safety glasses`, `munchkin white hot duck bath toy` and `first years take and toss straw cups` have significantly below-average quality models, they are not excluded from the tests since they are not high in number. Some of the objects from the dataset including the challenging ones that have transparent parts are given in Figure 8. The same procedure used for the BigBIRD dataset (Section 4.1.3) is followed for choosing the scans for the experiments. This resulted a total of 949 point clouds to be used in the experiments.

4.2. Tuning SPAIR/CoSPAIR: Choosing Number of Bins and Concentric Levels

There are two parameters in our descriptors: the number of concentric levels and the number of bins used for each sub-feature (angular relations given in Equations 5, 6, 7 for SPAIR; both angular relations and additional color histograms for CoSPAIR). To set these parameters, various experiments were conducted on Dataset 1: Subset of the RGB-D Object Dataset.

As the first step, we tested the performance of the SPAIR and CoSPAIR descriptor for various bin numbers. For 7 levels and a support radius of 10 cm, accuracy results are given in Figure 9. We see that 9 bins for each sub-feature provides the best accuracy considering instance-level recognition and second best with a minimal margin for category-level recognition. A similar analysis for CoSPAIR also yields similar results. Therefore, the number of bins is set to 9 for both SPAIR and CoSPAIR.

The second parameter is the number of the concentric levels. As our aim was to have a fixed the number of levels regardless of the chosen support radius, experiments were conducted for various support radius sizes. The results are given in Figure 10a for category-level recognition and in Figure 10b for instance-level recognition. As can

[1]The transparent objects are: `aunt jemima original syrup`, `bai5 sumatra dragonfruit`, `coca cola glass bottle`, `listerine`, `palmolive` (two instances), `softsoap` (five instances), `vo5` (three instances), `whiterain` (three instances) and `windex`.

Figure 8: Sample RGB images from the Amazon Picking Challenge dataset [21], each from different object.

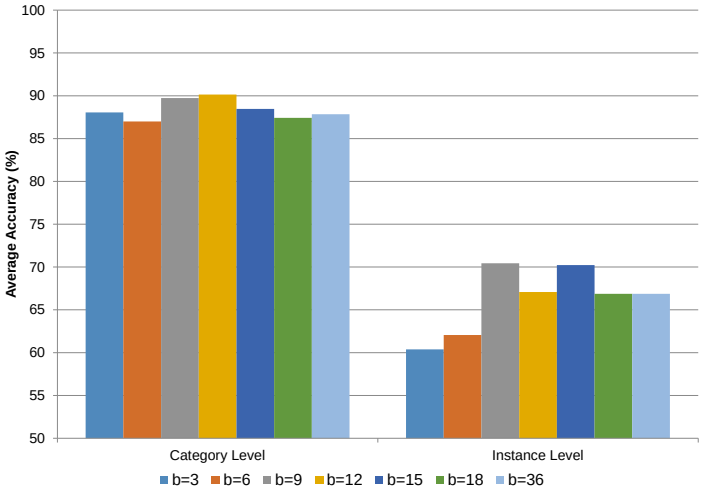


Figure 9: *Leave-sequence-out* average accuracy of SPAIR versus number of bins used in each level for each sub-feature where support radius is 10 cm and the number of levels is 7.

be observed from these figures, there is not a single particular number of levels where the accuracy is the highest for all support radius sizes. The performance is fairly stable after 4 levels with peak performances at around 7 and 8 levels. A similar analysis for CoSPAIR also reveals the same results. Therefore, the number of concentric levels was chosen to be 7 for all support radius sizes for both SPAIR and CoSPAIR.

Based on these choices, the size of the SPAIR descriptor becomes 189 due to 7 levels where each level consists of 3 histograms with 9 bins each. On the other hand, the size of the CoSPAIR descriptor is 378, i.e., double the size of the SPAIR descriptor due to the color histograms. In the remainder of the paper, the parameters of SPAIR and CoSPAIR are fixed and no further optimization is performed for Datasets 2, 3 and 4.

It should be noted that the parameters of the other compared descriptors are fixed in the Point Cloud Library at their best values and cannot be directly modified. Therefore, we used them as they are provided in the Point Cloud Library.

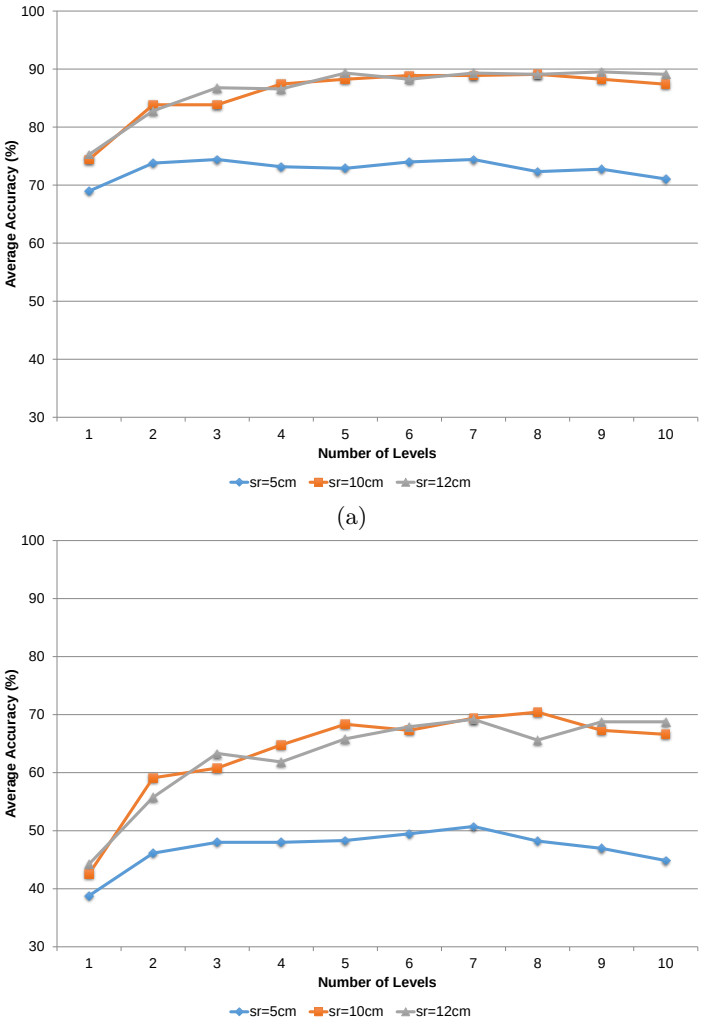


Figure 10: *Leave-sequence-out* average accuracy of SPAIR vs number of concentric levels used to extract the descriptor: a) Category-level, b) Instance-level.

4.3. Effect of Keypoint Detection Methods

The performances of all the descriptors were also evaluated for various keypoint detection methods; ISS3D [28], Harris3D [17] and uniform sampling using a 3D voxel grid with a leaf size of 1 cm. The average accuracy results are given in Table 2. It can be observed that the keypoint detection methods affect all the tested descriptors similarly.

Therefore, it is possible to choose a single extractor for all the descriptors. According to our evaluation, ISS3D performs better than Harris3D and its performance is very close to uniform sampling. Since ISS3D has been reported to stand out for its performance, repeatability and efficiency [29, 30] we used it as the keypoint detection method in our experiments.

Table 2: *Leave-sequence-out* average accuracy results of descriptors for different keypoint extraction methods where support radius is 10 cm.

	Category Level			Instance Level		
	ISS3D	H3D	Uni.	ISS3D	H3D	Uni.
SPAIR	89.73	68.76	89.94	70.44	38.99	68.55
FPFH [12]	80.29	66.88	81.93	51.36	37.53	51.05
SHOT [22]	90.15	80.92	90.97	61.84	50.31	65.55
CoSPAIR	95.39	87.00	96.23	84.91	72.75	86.16
CSHOT [16]	92.03	85.95	94.54	79.66	68.76	82.35

4.4. Results on Dataset 1: RGB-D Subset

The average accuracy, average per class recall and precision results are given in Table 3 for category-level recognition and Table 4 for instance-level recognition. In addition, *leave-sequence-out* average accuracies are shown in Figure 11 to visualize the performance trend with respect to the support radius size.

Results show that, in this small dataset, CoSPAIR slightly outperforms the second best performer CSHOT in category-level recognition, except for the *Alternating contiguous frames* methodology for low support radius sizes. CoSPAIR outperforms CSHOT with a higher margin in instance-level recognition using both methodologies (*leave-sequence-out* and *alternating-contiguous-frames*). In the *leave-sequence-out* methodology, CoSPAIR achieves 85.53% average accuracy at 12 cm whereas CSHOT achieves 79.66% at 10 cm; in the *alternating-contiguous-frames* methodology, CoSPAIR achieves 91.96% average accuracy at 10 cm compared to CSHOT's 87.20% at 8 cm.

Among the shape only descriptors, SPAIR performs slightly better for larger support radius sizes whereas SHOT performs better for smaller support radius sizes.

4.5. Results on Dataset 2: RGB-D All Objects

Next, we evaluate the methods on the whole RGB-D Object Dataset (with all the available 300 objects in 51 categories), which is much more challenging than Dataset 1. The average accuracy, average per class recall and precision results are given in Table 5 for category-level recognition and Table 6 for instance-level recognition. In addition, *leave-sequence-out* average accuracies are shown in Figure 12 to visualize the performance trend with respect to the support radius size. It should be noted that PFH and PFHRGB are excluded from this experiment because of these descriptors' prohibitively long extraction times on such a big dataset (see Section 4.8).

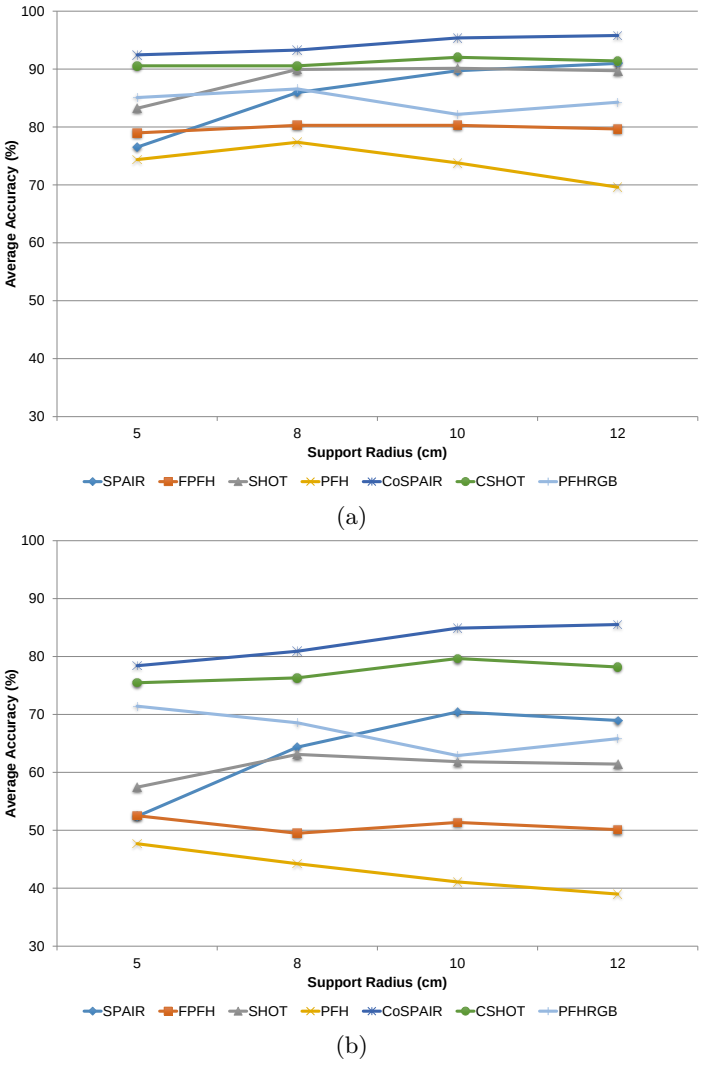


Figure 11: *Leave-sequence-out* average accuracy results for 10 category subset of RGB-D Object Dataset: a) Category-level, b) Instance-level.

In this dataset, for all support radius sizes and for both test scenarios, CoSPAIR outperforms all other descriptors in both category and instance-level recognition. For the *leave-sequence-out* scenario, CoSPAIR achieves an average accuracy of 86.21% for a support radius of 12 cm in category-level recognition and 74.46% in instance-level recognition for a support radius of 10 cm whereas the second top performer CSHOT achieves 76.31% in category-level recognition for a support radius of 10 cm and 57.97% in instance-level recognition for a support radius of 8 cm, leading to 16.49 percentage points (pp) performance difference. It is even higher if the same support radius is considered for all the descriptors; resulting up to 17.41 pp difference at 12 cm. For the *alternating-contiguous-frames* scenario, CoSPAIR outperforms competitors as well but with a slightly lower margin. CoSPAIR achieves an average accuracy of 96.15% for a support radius of 12 cm in category-level recognition and 90.09% in instance-level recognition for a support radius of 12 cm whereas the sec-

Table 3: Category-level average accuracy, average recall and average precision results for the 10 category subset of RGB-D Object Dataset

		sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
							Leave-sequence-out						
		Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR		76.52	74.85	78.95	85.95	84.58	84.12	89.73	88.90	88.87	90.99	90.26	90.72
FPFH	[12]	78.99	78.15	76.31	80.29	79.90	77.08	80.29	80.35	77.13	79.66	80.00	76.99
SHOT	[22]	83.23	81.07	82.17	89.94	88.63	87.53	90.15	88.66	87.92	89.73	88.63	88.18
PFH	[11]	74.37	74.31	71.33	77.36	77.08	74.89	73.79	74.18	73.28	69.60	70.48	73.07
CoSPAIR		**92.45**	**92.01**	**92.73**	**93.29**	**92.90**	**92.01**	**95.39**	**95.30**	**94.96**	**95.81**	**95.68**	**95.44**
CSHOT	[16]	90.57	89.89	90.89	90.57	89.15	89.77	92.03	91.48	92.29	91.40	90.95	91.45
PFHRGB	[17]	85.08	84.72	84.60	86.58	85.98	85.99	82.18	82.00	83.36	84.28	83.37	82.72
							Alternating contiguous frames						
SPAIR		78.54	75.67	78.20	88.25	86.17	86.76	90.27	88.57	88.79	91.26	89.66	90.29
FPFH	[12]	81.25	79.46	78.48	82.92	81.39	80.92	81.34	80.31	79.60	80.38	79.52	79.17
SHOT	[22]	87.89	85.74	86.97	92.52	91.01	91.11	93.39	91.78	92.13	93.36	92.21	92.68
PFH	[11]	78.58	76.92	76.81	78.57	77.10	77.97	74.78	74.47	77.48	72.76	72.59	76.57
CoSPAIR		96.45	95.46	96.23	97.09	96.29	96.84	**97.98**	**97.42**	**97.61**	**97.82**	**97.14**	**97.43**
CSHOT	[16]	**97.02**	**96.48**	**96.85**	**97.76**	**97.38**	**97.75**	97.44	97.07	97.22	97.14	96.69	96.89
PFHRGB	[17]	92.98	91.72	92.35	93.96	92.88	93.45	91.15	89.99	90.62	92.37	91.01	91.62

Table 4: Instance-level average accuracy, average recall and average precision results for the 10 category subset of RGB-D Object Dataset

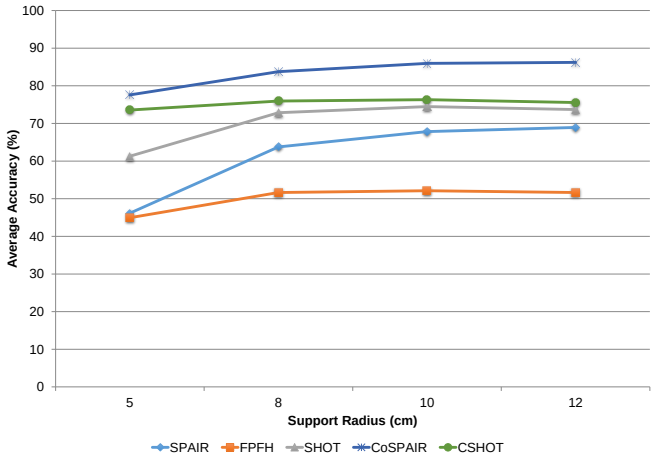
		sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
							Leave-sequence-out						
		Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR		52.41	52.55	51.10	64.36	64.62	66.09	70.44	70.54	72.75	68.97	69.25	70.81
FPFH	[12]	52.52	52.62	48.89	49.48	49.76	48.00	51.36	51.71	48.58	50.10	50.44	45.46
SHOT	[22]	57.44	57.58	59.81	63.10	63.66	63.01	61.84	62.36	62.16	61.43	61.89	64.18
PFH	[11]	47.69	47.85	45.51	44.23	44.51	44.16	41.09	41.39	40.65	38.99	39.26	40.13
CoSPAIR		**78.41**	**78.59**	**76.88**	**80.92**	**81.16**	**79.29**	**84.91**	**85.28**	**86.64**	**85.53**	**85.90**	**86.06**
CSHOT	[16]	75.47	76.02	74.20	76.31	76.88	76.24	79.66	80.17	77.85	78.20	78.75	75.99
PFHRGB	[17]	71.43	71.64	69.95	68.55	68.80	66.91	62.89	63.06	63.29	65.83	66.04	64.84
							Alternating contiguous frames						
SPAIR		55.45	55.07	55.98	65.11	64.98	65.85	66.87	66.76	67.56	66.76	66.69	67.42
FPFH	[12]	56.21	55.95	55.64	57.14	56.99	56.00	56.73	56.63	55.01	56.51	56.38	55.36
SHOT	[22]	62.62	62.54	64.72	65.71	65.78	67.75	66.12	66.18	68.62	65.85	65.87	68.09
PFH	[11]	52.18	52.02	50.94	49.69	49.60	48.32	47.50	47.45	48.46	46.68	46.72	48.90
CoSPAIR		**90.82**	**90.63**	**91.07**	**91.64**	**91.52**	**92.15**	**91.96**	**91.85**	**92.42**	**91.64**	**91.51**	**91.98**
CSHOT	[16]	87.16	87.08	88.11	87.20	87.19	88.73	86.15	86.12	87.71	85.87	85.84	87.26
PFHRGB	[17]	81.86	81.69	83.29	82.91	82.77	83.76	77.73	77.55	79.78	81.19	81.20	82.73

Table 5: Category-level average accuracy, average recall and average precision results for the RGB-D Object Dataset

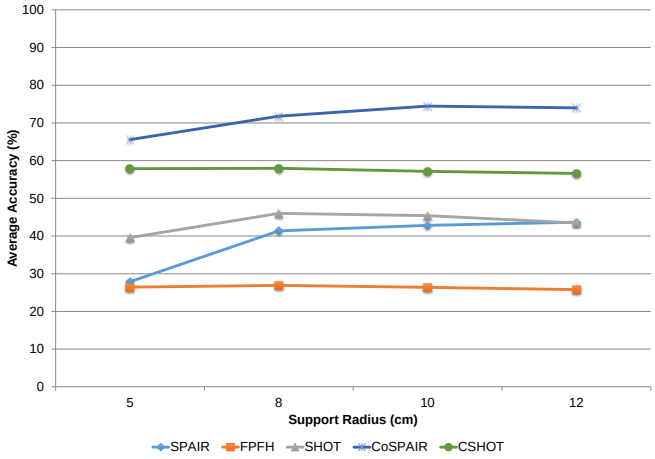
		sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
							Leave-sequence-out						
		Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR		46.16	48.61	51.63	63.77	64.31	66.93	67.84	67.11	69.29	68.93	67.85	69.83
FPFH	[12]	44.94	46.21	45.51	51.66	51.16	51.29	52.11	51.66	52.18	51.66	51.01	50.97
SHOT	[22]	61.28	63.06	64.92	72.85	72.86	73.13	74.49	73.90	74.25	73.70	73.03	73.08
CoSPAIR		**77.59**	**77.40**	**78.56**	**83.75**	**83.42**	**83.72**	**85.97**	**85.43**	**84.79**	**86.21**	**85.44**	**84.97**
CSHOT	[16]	73.55	72.74	74.59	75.95	74.71	77.03	76.31	74.86	77.54	75.55	74.14	76.10
							Alternating contiguous frames						
SPAIR		55.52	54.19	55.57	70.26	68.22	68.69	73.44	71.17	71.83	74.97	72.55	73.09
FPFH	[12]	55.97	53.45	52.45	61.60	58.47	58.82	62.40	59.10	58.96	62.48	59.12	59.03
SHOT	[22]	74.49	72.89	72.93	78.97	76.98	77.12	80.78	78.98	78.99	79.99	78.01	78.02
CoSPAIR		**92.88**	**91.98**	**92.84**	**94.98**	**94.30**	**94.43**	**95.68**	**95.03**	**95.20**	**96.15**	**95.49**	**95.63**
CSHOT	[16]	90.53	89.45	89.75	90.36	89.42	89.81	91.15	90.21	90.44	90.57	89.44	89.88

Table 6: Instance-level average accuracy, average recall and average precision results for the RGB-D Object Dataset

	sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
						Leave-sequence-out						
	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR	27.88	28.46	27.81	41.36	41.90	42.00	42.82	43.44	44.42	43.67	44.28	45.19
FPFH [12]	26.42	27.22	25.10	26.91	27.34	26.29	26.39	26.99	25.49	25.78	26.44	25.12
SHOT [22]	39.57	39.92	41.51	46.01	46.13	47.26	45.40	45.44	46.43	43.46	43.53	45.39
CoSPAIR	**65.56**	**64.26**	**64.54**	**71.79**	**70.42**	**69.23**	**74.46**	**73.22**	**72.46**	**74.01**	**72.79**	**71.66**
CSHOT [16]	57.85	56.65	55.67	57.97	56.71	56.44	57.15	55.96	55.89	56.60	55.48	55.31
						Alternating contiguous frames						
SPAIR	36.88	36.87	36.10	49.45	49.10	48.58	51.83	51.44	51.28	52.18	51.74	51.27
FPFH [12]	37.13	36.97	34.55	41.56	41.21	39.40	41.71	41.37	39.17	41.66	41.29	39.28
SHOT [22]	50.89	50.67	51.74	53.69	53.31	54.77	55.09	54.80	55.84	54.29	53.95	54.76
CoSPAIR	**87.52**	**86.41**	**88.01**	**89.26**	**88.21**	**89.30**	**89.89**	**88.90**	**90.14**	**90.09**	**89.10**	**90.29**
CSHOT [16]	81.17	80.28	81.76	78.57	77.73	79.92	79.95	79.15	81.08	79.12	78.24	80.03



(a)



(b)

Figure 12: *Leave-sequence-out* average accuracy results for the whole RGB-D Object Dataset: a) Category-level, b) Instance-level.

ond top performer CSHOT achieves 91.15% in category-level recognition for a support radius of 10 cm and 81.17% in instance-level recognition for a support radius of 5 cm.

Among the shape-only descriptors, in both category-level and instance-level recognition, SHOT performs slightly better than SPAIR, where the performance margin is larger for lower support radii and smaller for larger support radii. Among the tested descriptors, FPFH has the least performance for all support radius sizes in both category-level and instance-level recognition.

4.6. Results on Dataset 3: The BigBIRD Dataset

Since the BigBIRD dataset is an instance-level dataset and no category information is specified, only the instance-level recognition results are reported for this dataset. The average accuracy, average per class recall and precision results are given in Table 7. In addition, *leave-sequence-out* average accuracies are shown in Figure 13 to visualize the performance trend with respect to the support radius.

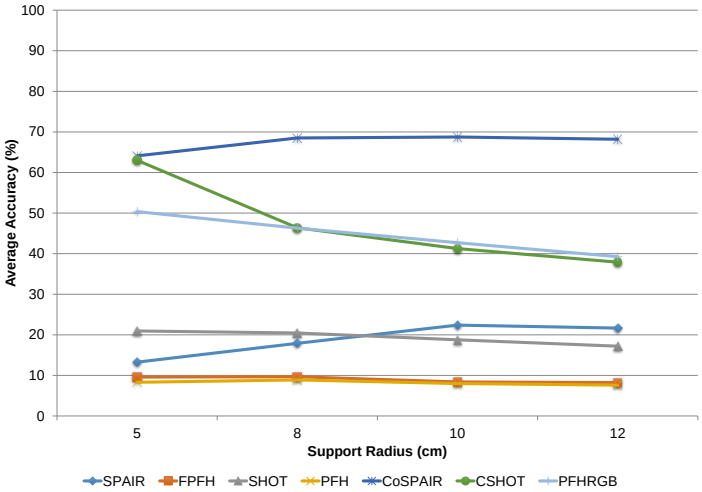


Figure 13: *Leave-sequence-out* instance-level average accuracy results for the BigBIRD dataset.

Since this dataset is instance-level, and the difference between many instances are in texture/color (see Figure 7) shape-only descriptors perform extremely poor. However, the shape + texture/color descriptors perform fairly well considering the challenging nature of this dataset. The

Table 7: Instance-level average accuracy, average recall and average precision results for the BigBIRD dataset.

		sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
		Leave-sequence-out											
		Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR		13.27	13.14	12.34	17.91	17.75	18.61	22.38	22.17	22.24	21.66	21.45	20.86
FPFH	[12]	9.59	9.51	8.78	9.67	9.58	8.21	8.39	8.32	7.89	8.23	8.15	6.43
SHOT	[22]	20.94	20.77	24.18	20.46	20.29	22.11	18.79	18.63	19.93	17.19	17.05	16.16
PFH	[11]	8.31	8.21	6.43	8.87	8.79	7.35	7.99	7.91	8.46	7.59	7.52	5.46
CoSPAIR		**64.11**	**63.58**	**67.08**	**68.51**	**67.96**	**73.00**	**68.75**	**68.57**	**72.39**	**68.19**	**68.02**	**70.13**
CSHOT	[16]	62.99	62.48	64.85	46.36	46.00	50.07	41.25	41.31	46.79	37.89	37.59	40.39
PFHRGB	[17]	50.36	49.95	50.63	46.28	45.93	46.60	42.69	42.34	42.92	39.25	38.93	37.36
		Alternating contiguous frames											
SPAIR		24.31	24.14	27.43	33.61	33.43	35.72	36.79	36.70	38.82	37.96	37.86	40.13
FPFH	[12]	24.36	24.16	26.36	30.41	30.19	33.13	31.35	31.14	33.31	31.79	31.57	33.75
SHOT	[22]	35.88	35.65	38.35	42.27	42.19	43.25	43.52	43.39	44.56	43.88	43.77	45.00
PFH	[11]	21.85	21.71	24.67	23.41	23.22	25.98	24.14	23.94	27.92	23.46	23.27	27.43
CoSPAIR		**81.29**	**80.83**	**83.36**	**81.46**	**81.20**	**83.32**	**81.18**	**80.94**	**83.46**	**79.86**	**79.51**	**82.08**
CSHOT	[16]	64.93	64.54	67.80	62.44	62.12	65.88	61.96	61.67	65.30	61.55	61.21	64.71
PFHRGB	[17]	75.42	74.78	76.90	71.44	70.88	73.84	69.20	68.64	72.18	67.97	67.45	71.49

best performing descriptor is CoSPAIR for both test scenarios. For the *leave-sequence-out* case, CoSPAIR achieves 68.75% average accuracy for support radius of 10 cm whereas the second best performer CSHOT achieves 62.99% for support radius of 5 cm. For the *alternating-contiguous-frames* scenario, CoSPAIR outperforms competitors. It achieves 81.46% average accuracy at 8 cm whereas the second top performer PFHRGB achieves 75.42% for 5cm. Although the best achieved scores can be considered close, the performance gap increases with the increasing support radii. For the *leave-sequence-out* case, although the performance gap between CoSPAIR and CSHOT is 1.12 pp at 5 cm, the gap increases up to 30.3 pp at 12 cm. Lastly, for the *alternating-contiguous-frames* scenario, the performance gap is lowest, 16.36 pp at 5 cm and highest, 19.22 pp at 10 cm.

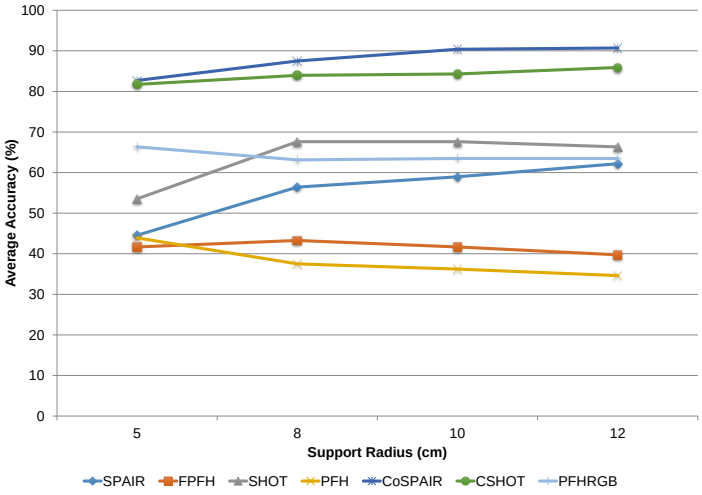


Figure 14: Instance-level average accuracy results for the Amazon Picking Challenge dataset.

4.7. Results on Dataset 4: The Amazon Picking Challenge Dataset

Like the BigBird, this dataset is an instance-level dataset and no category information is specified Thus, only the instance-level recognition results are reported. For both scenarios, CoSPAIR performs better than the competitors for all the tested support radii. For *leave-sequence-out*, CoSPAIR achieves 90.71% average accuracy for support radius of 12 cm whereas the second best performer CSHOT achieves 85.90% for the same support radius. For *alternating contiguous frames* scenario, CoSPAIR achieves 91.63% average accuracy at 12 cm whereas the second top performer CSHOT achieves 87.36% for 10 cm.

4.8. Extraction and Matching Times

For evaluating the extraction times, only a single scan for each category in the Dataset 1 with 1 cm uniform sampling is used. As a result, the query set for extraction times consists of 10 clouds with 3023 keypoints.

The average extraction times for a single keypoint/query point for 3 different support radius sizes are given in Table 9. As can be observed from the results, SHOT and CSHOT are very fast to extract whereas PFH and PFHRGB are prohibitively slow to use in practical applications. Moreover, while SPAIR and CoSPAIR are slower than SHOT, they are significantly faster than FPFH, PFH and PFHRGB. The main reason behind the speed of SHOT and CSHOT despite being longer is to use a single reference frame for each descriptor whereas SPAIR, CoSPAIR, PFH and RGBPFH fit a reference axis for each pair of points between which angular relations are computed.

And lastly, the brute-force matching times together with the size of the descriptors are given in Table 10. In this test, the full reference and query sets in the Dataset 1 were used where the query set contains 78,442 keypoints from 475 objects and the reference set contains 143,234 keypoints from 946 objects, thus the total number of comparisons were over 11 billion. Since the same matching method is used for all descriptors, the matching time is mainly related to the type and the length of the descrip-

Table 8: Instance-level average accuracy, average recall and average precision results for the Amazon Picking Challenge Dataset

		sr = 5cm			**sr = 8cm**			**sr = 10cm**			**sr = 12cm**		
						Leave-sequence-out							
		Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.	Acc.	Rec.	Prec.
SPAIR		44.55	42.66	39.14	56.41	54.15	54.09	58.97	56.73	55.61	62.18	59.81	59.80
FPFH	[12]	41.67	40.14	42.83	43.27	41.62	43.27	41.67	40.10	38.36	39.74	38.31	36.21
SHOT	[22]	53.53	51.61	51.32	67.63	65.19	68.43	67.63	65.25	67.71	66.35	64.01	68.46
PFH	[11]	43.91	42.25	45.07	37.50	36.06	31.97	36.22	34.83	33.26	34.62	33.33	31.39
CoSPAIR		**82.69**	**79.88**	**83.50**	**87.50**	**84.38**	**88.04**	**90.38**	**87.22**	**88.97**	**90.71**	**87.65**	**88.42**
CSHOT	[16]	81.73	78.95	81.16	83.97	81.05	82.20	84.29	81.36	82.13	85.90	82.90	83.72
PFHRGB	[17]	66.35	63.81	70.58	63.14	60.92	59.41	63.46	61.33	61.84	63.46	61.33	63.24
						Alternating contiguous frames							
SPAIR		46.18	45.29	49.17	60.49	59.45	62.80	64.89	63.80	65.57	67.10	66.05	68.07
FPFH	[12]	40.86	40.03	42.41	45.36	44.42	46.39	47.24	46.33	48.22	45.70	44.77	46.60
SHOT	[22]	62.51	61.59	63.56	75.00	74.02	76.22	75.82	74.73	76.63	76.06	75.03	76.40
PFH	[11]	42.45	41.67	41.44	42.83	42.15	44.07	42.07	41.35	43.33	40.44	39.95	41.65
CoSPAIR		**89.47**	**88.30**	**89.32**	**91.97**	**90.83**	**91.61**	**91.39**	**90.22**	**91.05**	**91.63**	**90.44**	**91.08**
CSHOT	[16]	84.06	82.99	84.45	86.88	85.79	86.68	87.36	86.26	87.58	87.02	85.92	87.12
PFHRGB	[17]	80.57	79.50	81.49	79.76	78.70	81.53	79.42	78.41	81.30	81.10	80.06	82.42

Table 9: Average extraction times (ms) of the descriptors for a single keypoint/query point. (Platform: i5 4670 CPU using a single core)

		sr=5cm	sr=10cm	sr=12cm
SPAIR		4.37	11.98	15.23
FPFH	[12]	16.83	49.22	63.53
SHOT	[22]	1.27	2.55	3.10
PFH	[11]	506.50	5456.31	9409.57
CoSPAIR		5.37	14.27	18.22
CSHOT	[16]	1.45	3.96	5.04
PFHRGB	[17]	918.67	10049.05	17304.95

tors. As all the descriptors are of type *float*, descriptor length is the only factor affecting the matching performance. This can be directly seen from the results that FPFH, being the shortest descriptor, is the fastest to match and CSHOT, being the largest, is the slowest to match.

Table 10: Lengths and matching times (seconds) of the descriptors. (Platform: i5 4670 CPU utilizing all 4 cores)

		Length	Matching Time (s)
SPAIR		189	119
FPFH	[12]	33	34
SHOT	[22]	392	170
PFH	[11]	125	88
CoSPAIR		378	197
CSHOT	[16]	1344	581
PFHRGB	[17]	250	136

5. Conclusion and Future Work

In this paper, we have proposed two new local 3D descriptors. In our descriptors, the support radius is divided into concentric spherical shells. We have demonstrated that such partitioning of space allows encoding enhanced spatial information more effectively.

We have compared the proposed descriptors with the state-of-the-art local 3D descriptors that are available in the Point Cloud Library. The shape only descriptor, SPAIR is shown to be one of the best in its class (shape only) while CoSPAIR is shown to outperform the tested state-of-the-art descriptors both for category-level and instance-level object recognition. We have observed up to 9.9 percentage points gain in category-level recognition and 16.49 percentage points gain in instance-level recognition over the second-best performing descriptor in the RGB-D dataset.

5.1. Future Work

The descriptors proposed in this article can be improved in a number of ways in the future. One of them is the way the spatial information is integrated into the descriptors. In the proposed descriptors, the support radius is divided into equally-sized shells. Since the contribution of the central shells and the outer shells may be different, one may consider shells having varying thicknesses and learning the optimal division of the support radius into the shells. Another similar line of work that is to extend the system to use a weighted combination of the histograms coming from different shells.

In addition to these, the information extracted in each shell can be extended by other 3D or 2D information, such as, local 3D curvature, local 3D shape category via the method of shape index, 2D textural features. These can enrich the representation in each shell.

References

[1] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, M. Vincze, Point cloud library, IEEE Robotics & Automation Magazine 1070 (9932/12).

[2] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Transactions on Graphics (TOG) 21 (4) (2002) 807–832.

[3] R. B. Rusu, G. Bradski, R. Thibaux, J. Hsu, Fast 3d recognition and pose using the viewpoint feature histogram, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010, pp. 2155–2162.

[4] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. B. Rusu, G. Bradski, Cad-model recognition and 6dof pose estimation using 3d cues, in: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2011, pp. 585–592.

[5] W. Wohlkinger, M. Vincze, Ensemble of shape functions for 3d object classification, in: IEEE International Conference on Robotics and Biomimetics (ROBIO), 2011, pp. 2987–2992.

[6] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3d shape descriptors, in: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '03, 2003, pp. 156–164.

[7] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, M. Ouhyoung, On visual similarity based 3d model retrieval, in: Computer graphics forum, Vol. 22, Wiley Online Library, 2003, pp. 223–232.

[8] R. Ohbuchi, K. Osada, T. Furuya, T. Banno, Salient local visual features for shape-based 3d model retrieval, in: IEEE International Conference on Shape Modeling and Applications, SMI, 2008, pp. 93–102.

[9] C. B. Akgul, B. Sankur, Y. Yemez, F. Schmitt, 3d model retrieval using probability density-based shape descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (6) (2009) 1117–1133.

[10] A. E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3d scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (5) (1999) 433–449.

[11] R. B. Rusu, N. Blodow, Z. C. Marton, M. Beetz, Aligning point cloud views using persistent feature histograms, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS), 2008, pp. 3384–3391.

[12] R. B. Rusu, N. Blodow, M. Beetz, Fast point feature histograms (fpfh) for 3d registration, in: IEEE International Conference on Robotics and Automation, ICRA'09, 2009, pp. 3212–3217.

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

[14] A. Zaharescu, E. Boyer, K. Varanasi, R. Horaud, Surface feature detection and description with applications to mesh matching, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2009, pp. 373–380.

[15] F. Tombari, S. Salti, L. Di Stefano, Unique signatures of histograms for local surface description, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 356–369.

[16] F. Tombari, S. Salti, L. Di Stefano, A combined texture-shape descriptor for enhanced 3d feature matching, in: 18th IEEE International Conference on Image Processing (ICIP), 2011, pp. 809–812.

[17] R. B. Rusu, S. Cousins, 3d is here: Point cloud library (pcl), in: IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 2011.

[18] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view rgb-d object dataset, in: IEEE International Conference on Robotics and Automation (ICRA), 2011, pp. 1817–1824.

[19] L. A. Alexandre, 3d descriptors for object and category recognition: a comparative evaluation, in: Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, Citeseer, 2012.

[20] A. Singh, J. Sha, K. S. Narayan, T. Achim, P. Abbeel, Bigbird: A large-scale 3d database of object instances, in: IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 509–516.

[21] A. Singh, K. Narayan, B. Kehoe, S. Patil, K. Goldberg, P. Abbeel, Amazon picking challenge object scans, http://rll.berkeley.edu/amazon_picking_challenge/.

[22] S. Salti, F. Tombari, L. Di Stefano, Shot: Unique signatures of histograms for surface and texture description, Computer Vision and Image Understanding 125 (2014) 251–264.

[23] E. Wahl, U. Hillenbrand, G. Hirzinger, Surflet-pair-relation histograms: a statistical 3d-shape representation for rapid classification, in: IEEE Fourth International Conference on 3-D Digital Imaging and Modeling, 3DIM 2003, 2003, pp. 474–481.

[24] L. Bo, X. Ren, D. Fox, Unsupervised feature learning for rgb-d based object recognition, in: Experimental Robotics, Springer, 2013, pp. 387–402.

[25] G. James, Majority vote classifiers: theory and applications, Ph.D. thesis, Stanford University (1998).

[26] G. Bradski, The OpenCV Library, Dr. Dobb's Journal of Software Tools.

[27] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Information Processing & Management 45 (4) (2009) 427–437.

[28] Y. Zhong, Intrinsic shape signatures: A shape descriptor for 3d object recognition, in: IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), 2009, pp. 689–696.

[29] S. Filipe, L. A. Alexandre, A comparative evaluation of 3d keypoint detectors in a rgb-d object dataset, in: 9th International Conference on Computer Vision Theory and Applications, Lisbon, Portugal, 2014.

[30] F. Tombari, S. Salti, L. Di Stefano, Performance evaluation of 3d keypoint detectors, International Journal of Computer Vision 102 (1-3) (2013) 198–220.

[31] L. A. Alexandre, Exact list of rgb-d dataset subset, http://www.di.ubi.pt/~lfbaa/files/train_test_file_ids.tar.gz.

[32] A. Singh, J. Sha, K. Narayan, T. Achim, P. Abbeel, Bigbird: (big) berkeley instance recognition dataset, http://rll.berkeley.edu/bigbird/.

[33] K. S. Narayan, J. Sha, A. Singh, P. Abbeel, Range sensor and silhouette fusion for high-quality 3d scanning, sensors 32 (33) (2015) 26.