RISAS: A Novel Rotation, Illumination, Scale Invariant Appearance and Shape Feature

Kanzhi Wu[1,†], Xiaoyang Li[2], Ravindra Ranasinghe[1], Gamini Dissanayake[1] and Yong Liu[2]

Abstract— This paper presents a novel appearance and shape feature, RISAS, which is robust to viewpoint, illumination, scale and rotation variations. RISAS consists of a keypoint detector and a feature descriptor both of which utilise texture and geometric information present in the appearance and shape channels. A novel response function based on the surface normals is used in combination with the Harris corner detector for selecting keypoints in the scene. A strategy that uses the depth information for scale estimation and background elimination is proposed to select the neighbourhood around the keypoints in order to build precise invariant descriptors. Proposed descriptor relies on the ordering of both grayscale intensity and shape information in the neighbourhood. Comprehensive experiments which confirm the effectiveness of the proposed RGB-D feature when compared with CSHOT [1] and LOIND[2] are presented. Furthermore, we highlight the utility of incorporating texture and shape information in the design of *both* the detector and the descriptor by demonstrating the enhanced performance of CSHOT and LOIND when combined with RISAS detector.

I. INTRODUCTION

Feature matching is a fundamental problem in both computer vision applications (e.g., object detection and image retrieval) and robotic tasks (e.g., vision based Simultaneous Localisation and Mapping). Two critical steps toward finding robust and reliable correspondences are: 1) extracting discriminative keypoints, 2) building invariant descriptors. Over the past decades, there have been enormous progress in developing robust features in two-dimensional image space such as SIFT(Scale Invariant Feature Transform) [3], SURF(Speed-Up Robust Feature) [4] and ORB(Oriented FAST and Rotated BRIEF) [5]. These methods achieve excellent performance under significant scale and rotation variations when rich texture information is available, however, their performances dramatically degrade under illumination variations or in environments with poor texture information.

With the development of low-cost, real-time depth sensors such as Kinect and Xtion, the geometric information of the environment can be accessed easily, thus it is now prudent to consider geometric information in building local descriptors. Spin Image[6] is one of the well-known 3D descriptors which is widely used in 3D surface registration tasks. Rusu et
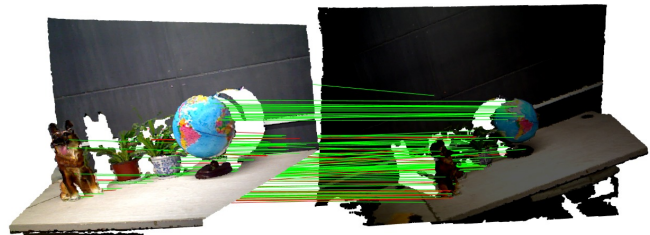

Fig. 1: Feature matching results using RISAS. The correct matches are shown using green lines and incorrect matches are denoted by red lines.

al. [7][8][9] also made contributions and proposed various depth descriptors such as PFH (Persistent Feature Histogram) and NARF (Normal Aligned Radial Feature). Despite the above developments, relying on depth information alone makes the correspondence problem more challenging due to two facts: 1) sensor information from Kinect and Xtion is generally incomplete 2) depth image is much less informative compared with RGB/grayscale image, particularly on regular shaped surfaces. Due to their complementary nature, combining appearance and geometric information to build descriptors with the aim of improved matching performance has been explored by a number of researchers. CSHOT(Color Signature of Histogram and OrienTation) and BRAND(Binary Robust Appearance and Normal Descriptor) are examples of such RGB-D descriptors. However, there has been no attempt specifically to design keypoint detectors for these descriptors. Thus the selected keypoints may not reflect the best available regions in the scene for generating a robust descriptor.

The main contribution of this paper is a novel Rotation, Illumination and Scale invariant Appearance and Shape feature (RISAS) which tightly couples a discriminative RGB-D keypoint detector and an invariant feature descriptor. As a result of using texture and shape information in the design of both the detector and the descriptor, RISAS shows superior performance over current state-of-the-art methods under various conditions. Fig. 1 demonstrates the capability of RISAS for obtaining correspondences under severe rotation, scale and illumination changes. Furthermore, benefits of using a RGB-D keypoint detector is highlighted by the enhanced performance of CSHOT descriptor when combined with the RISAS detector. In addition, a public domain dataset which can be used for future evaluations of RGB-D features is also made available.

The paper is structured as follows: Section II reviews

†Kanzhi Wu is with DJI Technology, Shenzhen, China, 518057 `kanzhi.wu@dji.com`
[1]Kanzhi Wu, Gamini Dissanayake and Ravindra Ranasinghe are with Centre for Autonomous Systems, University of Technology Sydney, Australia, 2007 `kanzhi.wu, gamini.dissanayake, ravindra.ranasinghe@uts.edu.au`
[2]Xiaoyang Li and Yong Liu are with Institute of Cyber-Systems and Control, Zhejiang University, China, 310027 `zjulixiaoyang@163.com, yongliu@iipc .zju.edu.cn`

the related work on both texture and shape based features. Section III introduces the novel keypoint detector computed using both grayscale image and depth information. A novel RGB-D descriptor, that is built on LOIND(Local Ordinal Intensity and Normal Descriptor)[2] with significant enhancements is also described in Section III. In Section IV, the proposed feature is experimentally evaluated using an existing public dataset as well as with a new dataset that includes variations in viewpoint, illumination, scale and rotation separately. Conclusion and future work are discussed in Section V.

II. RELATED WORK

In general, feature extraction can be separated into two sub-problems: keypoint detection and descriptor construction. Some of the feature extraction algorithms such as SIFT[3] and SURF[4] tightly couple these two steps while methods such as FAST(Features from Accelerated Segment Test) and BRIEF(Binary Robust Independent Elementary Features) only focus on either keypoint detection or feature description.

A. 2D Appearance Features

SIFT is one of the most well-known visual features [3]. SIFT combines a Difference-of-Gaussian interest region detector and a gradient orientation histogram as the descriptor. By constructing the descriptor from a scale and orientation normalised image patch, SIFT exhibits robustness to scale and rotation variations. SURF, proposed by Bay et al.[4], relies on integral images for image convolution. SURF uses a Hessian matrix-based measure for the detector and a distribution-based descriptor. Calonder et al.[10] proposed BRIEF which uses a binary string as the descriptor. BRIEF feature takes relatively less memory and can be matched fast using Hamming distance in real-time with very limited computational resources. However, BRIEF is not designed to be robust to scale variations. Leutenegger et al.[11] proposed BRISK(Binary Robust Invariant Scalable Keypoints) which has a scale invariant keypoint detector and binary string like descriptor. ORB, another well-known binary feature, proposed by Rublee et al.[12], has been widely used in SLAM community[13]. ORB is invariant to rotation variations and more robust to noise compared with BRIEF.

B. 3D Geometric Features

In order to select salient keypoints from geometric information, researchers have adopted different criteria to evaluate the distinctiveness of the points in the scene, e.g., the normal vector of the surface and curvature of the mesh. Survey paper from Tombari et al. [14] categorises 3D keypoint detectors into 2 classes: *fixed-scale* detectors and *adaptive-scale* detectors and provides a detailed comparison of existing 3D keypoint detectors. Hebert contributed several well-known *adaptive-scale* detectors such as LBSS (Laplace-Beltrami Scale-Space) and MeshDoG[15]. Zhong et al. proposed Intrinsic Shape Signature (ISS)[16] to characterise a local/semi-local region of a point cloud and ISS had been

combined with various 3D descriptors in RGB-D descriptor evaluation[17].

Descriptors can also be constructed using 3D geometric information. Johnson and Hebert [6] proposed spin image which is a data level descriptor that can be used to match surfaces represented as meshes. With the development of low-cost RGB-D sensors, geometric information of the environment can be easily captured thus 3D shape descriptors have attracted renewed attention. More recent developments include PFH [7], FPFH(Fast PFH) and SHOT (Signature of Histograms of OrienTations). Rusu et al. proposed PFH [7] which is a multi-dimensional histogram which characterises the local geometry of a given keypoint. PFH is invariant to position, orientation and point cloud density. Enhanced version of PFH, termed FPFH [8] reduces the complexity of PFH from $O(k^2)$ to $O(k)$ where k is the number of points in the neighbourhood of the keypoint. SHOT descriptor proposed by Tombari et al. [18] is another example of a widely used local surface descriptor. SHOT encodes the histograms of the surface normals in different partitions in the support region.

C. Combined Appearance and Depth Features

Lai et al. [19] have demonstrated that by combining RGB and depth channels together, better object recognition performance can be achieved. Tombari et al. [1] developed CSHOT via incorporating RGB information into original SHOT descriptor. Nascimento et al. [20] proposed a binary RGB-D descriptor BRAND which encodes local information as a binary string thus makes it feasible to achieve low memory consumption. They have also demonstrated the rotation and scale invariance of BRAND. More recently, Feng et al. [2] proposed LOIND which encodes the texture and depth information into one descriptor supported by orders of intensities and angles between normal vectors. Most of the current RGB-D fused descriptors adopt traditional 2D keypoint detectors that rely only on appearance information. For instance, BRAND is combined with CenSurE(Centre Surround Extremas[21]) detector and LOIND uses keypoints from multi-scale Harris detector. In CSHOT, in order to eliminate the influence of detector, the keypoints are selected randomly from the model. Clearly, selecting keypoints by exploiting geometrically information-rich regions in the scene has the potential to enhance the matching performance of a RGB-D descriptor.

Deep learning methods have been found to be very effective in object detection tasks using RGB-D images[22], [23]. However, these are not directly comparable to low-level RGB-D information processing which is the focus of the present work.

III. METHOD

In this section, we describe the proposed Rotation, Illumination and Scale invariant Appearance and Shape feature, RISAS, in detail. RISAS is built on our previous work [2]. The detector and descriptor are explained in detail in Section.III-A and Section.III-B.

A. Keypoint Detector

The main advantage of using depth information in keypoint detection is the fact that information rich regions in the depth channel are also given due consideration without being ignored when these regions lack texture information. Both the proposed detector and the descriptor use similar information and thus are tightly coupled giving rise to superior matching performance.
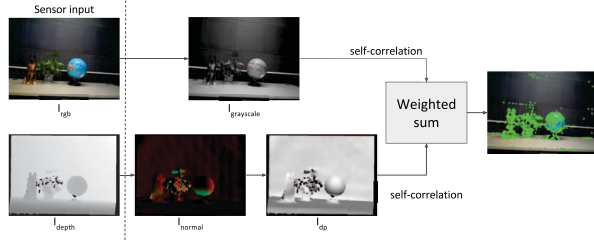

Fig. 2: Flowchart of the proposed RGB-D keypoint detector. I_{rgb} is the original RGB image and $I_{grayscale}$ is the converted grayscale image. I_{normal} is the 3 channel normal vector image and I_{dp} is the dot product image.

The flowchart of the keypoint detection method is shown in Fig. 2 and the key steps are listed below:

1) For each point in the depth image I_{depth}, we calculate the surface normal vector. From the three components of the normal vector, we create the corresponding normal image I_{normal} with three channels.

2) Using I_{normal}, we compute the three angles $[\alpha, \beta, \gamma]$ between each normal vector and the $[x, y, z]$ axis of the camera coordinate system respectively. The angle range $[0, \pi]$ is segmented into n_s sectors labelled with $[1, ..., n_s]$ and each computed angle is mapped into one of these sectors. In this work, n_s is set to be 4 as shown in Fig. 3. For example, normal vector $\mathbf{n} = \left[\frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}\right]$ has the $[\alpha, \beta, \gamma] = [54.7°, 54.7°, 54.7°]$ will be labelled as $[2, 2, 2]$;
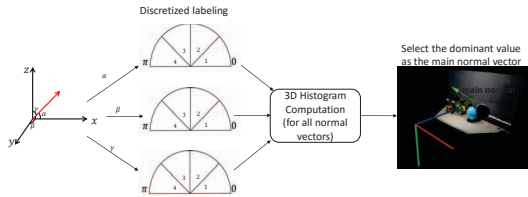

Fig. 3: Flowchart of calculating the main normal.

3) Using this labelled image, we build a statistical histogram to capture the distribution of labels along each channel. From this histogram, we choose the highest entry for each channel and use the corresponding label $[n_X, n_Y, n_Z]$ to represent the most frequent label where $n_X, n_Y, n_Z \in \{1, ..., n_s\}$. Using these three values, we define the "main" normal vector $\mathbf{n}_{main}$ of the depth image I_{depth}.

4) Calculate the dot-product between $\mathbf{n}_{main}$ and each normal vector in I_{normal}. This describes the variation

of information in depth channel. We then normalise the dot product value into range $[0, 255]$. Using this value, we create the novel dot-product image I_{dp} which is approximately invariant to the viewpoint of the sensor.

5) We adopt the similar principle as in the Harris detector to compute the response value $E(u, v)$ using the grayscale image $I_{grayscale}$ and the dot product image I_{dp}. $E(u, v)$ is thresholded to select points that show an extreme value in the weighted sum of two response values from $I_{grayscale}$ and I_{dp}, as shown in (1):

$$E(u, v) = \sum_{x,y} \omega(x, y)[\tau \left(I(x + u, y + v) - I(x, y)\right)^2 + (1 - \tau)\left(P(x + u, y + v) - P(x, y)\right)^2]$$
(1)

where (u, v) is the keypoint coordinate in image space and $\omega(x, y)$ is the window function centred at (u, v) which is selected to be a Gaussian function. $I(u, v)$ is the intensity value at (u, v) and $P(u, v)$ is the normalized dot product value at (u, v). Empirical study shows that τ plays a critical role in balancing appearance information and geometric information in keypoint detection. Fig. 4 provides precision-recall curves for different τ value for the same scenario. We selected $\tau = 0.8$ after numerical experimental evaluations.
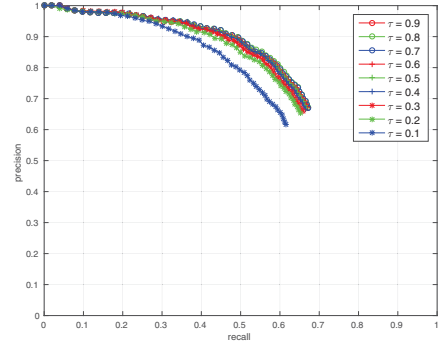

Fig. 4: Precision-Recall curves for difference τ value.

This strategy clearly identifies keypoints from regions that are information rich in both appearance and geometry.

B. Feature Descriptor

1) Scale Estimation and Neighbourhood Region Selection: For grayscale images, the scale of the keypoint is estimated by finding the extreme value in scale space using an image pyramid. Typical examples are as SIFT[3] and SURF[4]. With the development of modern RGB-D sensors such as Kinect and Xtion, the scale can be easily measured using the depth information captured from the sensor. In both LOIND[2] and BRAND[20], the following empirical equation scales the distance range between $[2, 8]$ into scale range $[1, 0.2]$ in a linear relationship. Scale value for distance less than 2m is truncated as 1.

$$s = \max\left(0.2, \frac{3.8 - 0.4\max(2, d)}{3}\right)$$
(2)

After s is estimated, the neighbourhood region that is used to build the descriptor is selected with radius R in a linear

relationship with scale value s, as shown in [2][20]. A critical deficiency in this approach is that the neighbourhood region is selected without considering the geometric continuity. Here we present a better method for selecting the neighbourhood region as shown in Fig. 5.

Fig. 5: Neighbourhood selection: The default strategy (left) selects the whole region (shown in red) which covers both foreground and background area. However, the introduced background points have an adverse effect on the local descriptor. Our approach (right) eliminates the background points (shown in blue) and constructs the descriptor using the foreground (shown in red) only, leading to more robust descriptor matching performances.

1) Based on (2), initial value of the scale s is estimated. The radius R of the patch is computed using (3) which was derived using extensive experimentation.

$$R = \left(-5 + 25 \cdot \min\left(3, \frac{\max(0.2, s_{max})}{\max(0.2, s_{min})}\right)\right) \cdot s \quad (3)$$

Where s_{max} and s_{min} are the maximum and minimum scale values in the image. It is an empirical value based on the experiments, if scale variation is small in the neighbourhood region, we can choose a smaller R and vice versa. We denote the patch centred at keypoint k_i in 2D image space as $\mathbf{P}^{uv}(k_i)$ and the corresponding patch in 3D point cloud space is represented as $\mathbf{P}^{xyz}(k_i)$;

2) For each point $p \in \mathbf{P}^{xyz}(k_i)$, we remove the outlier neighbouring points from the keypoint k_i according to (4). This step of eliminating the background was found to produce significant improvements in the matching performance.

$$f(p) = \begin{cases} 1 & \text{if } \left\|p - k_i\right\| < t \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where t is the threshold and set to be 0.1 meter in this work. We only keep the neighbouring points with $f(p) = 1$;

3) We conduct ellipsoid fitting for the processed 3D neighbouring points $\bar{\mathbf{P}}^{xyz}(k_i)$ based on the following equation.

$$\frac{(x - x_{k_i})^2}{a^2} + \frac{(y - y_{k_i})^2}{b^2} + \frac{(x - z_{k_i})^2}{c^2} = 1 \quad (5)$$

where a, b and c are the length of the axes. We project the 3D ellipsoid into the image space for the new accurate patch $\bar{\mathbf{P}}^{uv}$ with radius $\bar{R}$ for further descriptor construction.

2) Orientation Estimation: In LOIND[2], the dominant orientation θ of the selected patch is computed from the depth information only. Although it works reasonably well under different scenarios it is sensitive to the noise in neighbourhoods where the normal vectors are similar to each other. In the following, we propose a new dominant orientation estimation algorithm which is more robust and efficient compared with LOIND[2]:

1) Given the processed 2D patch $\bar{\mathbf{P}}^{uv}$ and 3D patch $\bar{\mathbf{P}}^{xyz}$, we adopt PCA[24] to compute the eigenvalues $[e_1, e_2, e_3]$ (in *descending* order) and corresponding eigenvectors $[\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$.

2) Given the eigenvectors $[\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$, the 3D dominant orientation d_{3D} of the patch is computed as follows:

$$d_{3D} = \begin{cases} \frac{\mathbf{v}_1 \times \mathbf{v}_2}{|\mathbf{v}_1 \times \mathbf{v}_2|} & \text{if } (e_2 > \gamma e_1) \wedge (e_3 \le \gamma e_1) \\ \text{rejected} & \text{if } (e_2 > \gamma e_1) \wedge (e_3 > \gamma e_1) \quad (6) \\ \mathbf{v}_1 & \text{others} \end{cases}$$

where γ is set within $[0, 1]$. If the e_1 is significantly larger than other two, the 3D dominant orientation is set to be the corresponding eigenvector $\mathbf{v}_1$. If e_2 is close to e_1, both eigenvector $\mathbf{v}_1$ and $\mathbf{v}_2$ are considered in computing the dominant orientation by taking the cross-product of these two vectors. Further if both e_3 and e_2 are closer to e_1 which means no clear differences between 3 eigenvalues, this keypoint is regarded as a noisy point and rejected. Threshold γ determines when the second eigenvalues e_2 can be regarded as "close" enough to the largest eigenvalue e_1 which is set to be 0.8 through experiments.

3) Project the 3D dominant direction d_{3D} into the image plane and get the 2D dominant direction d_{2D}. We use θ to denote the angle between d_{2D} and u axis in image space.

3) Descriptor Construction: Based on the results from the above steps, we can construct the descriptor of keypoint $k_i = [u, v]$ using the neighbourhood region with radius R and the angle θ. We follow the main ideas used in LOIND[2]. The descriptor is based on the relative order information in both grayscale and depth channels. The descriptor is constructed in a three-dimensional space, as show in Fig. 6 below where $[x, y, z]$ axes denote the spatial labelling, the intensity labelling and the angles labelling respectively.
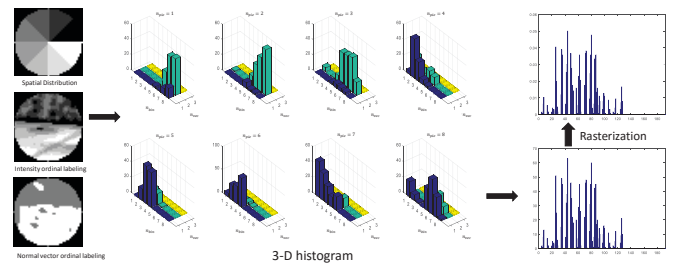

Fig. 6: Flowchart of the RGB-D descriptor.

- **Encoding Spatial Distribution**
  For spatial distribution, the pixels in the region (u, v, R, θ) are labeled based on n_{pie} equal-size spatial sectors. Larger the number of sectors, the more discriminative the descriptor, but this clearly effects on timing for both construction and matching.
- **Encoding Grayscale Information**
  Instead of constructing the descriptor in the absolute intensity space, we build the statistical histogram using the relative intensity with respect to the intensity value of the keypoint, in order to enhance illumination invariance. According to the rank of all the pixels in the patch, we group the intensity values into n_{bin} equally sized bins. For example, given 100 intensity levels and 10 bins, each bin has 10 intensity levels (i.e., orderings of $[1, 10], [11, 20], \ldots, [91, 100]$).
- **Encoding Geometric Information**
  Given the normal vector of each point, we first compute the dot product between the normal vector of the selected keypoint $\mathbf{n}_{p_k}$ and the normal vector of each point in the neighbourhood patch $\mathbf{n}_{p_i}$.

$$\rho_i = |\langle \mathbf{n}_{p_k}, \mathbf{n}_{p_i} \rangle| \qquad (7)$$

Due to the fact that normal vectors from small patches are similar to each other, thus the distribution of ρ_i is highly unbalanced where the majority of ρ_i falling into the range close to 1. We set a threshold $\bar{\rho} = 0.9$ and any $\rho_i \geq \bar{\rho}$ are grouped in to one category. The remaining dot products are ranked and grouped into n_{vec} bins. Points are then labelled based on the group they belong to respectively. Therefore, in normal vector space encoding, there are overall $n_{vec} + 1$ labels.

During the empirical study, we tested 12 different combinations of parameters $n_{pie} = \{4, 8, 12\}, n_{bin} = \{8, 16\}$ and $n_{vec} = \{1, 2\}$. Corresponding precision-recall curves are presented in Fig. 7. Considering both performance and efficiency, we set parameters as $n_{pie} = 8, n_{bin} = 8, n_{vec} = 2$ thus we use a 192-dimensional ($\dim = n_{pie} \cdot n_{bin} \cdot (n_{vec}+1)$) descriptor for the experiments reported.
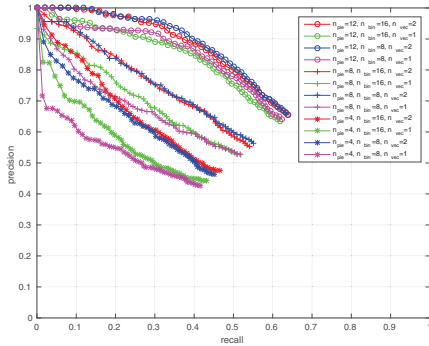

Fig. 7: Parameter selection for descriptor construction.

IV. EXPERIMENTAL RESULTS

In this section, we compare the performance of RISAS against CSHOT, LOIND and other methods. We also report the results obtained using SIFT, to highlight the value of using both appearance and depth channels. We use a public RGB-D dataset which has been designed for object detection[1]. This dataset does not include examples of rotation, scale or illumination changes independently and therefore is not able to fully illustrate the effectiveness of the RISAS in such situations. Therefore we have designed our own dataset for further detailed evaluations[2].

A. Evaluation Method

First we extract keypoints from two frames and construct the descriptors for all these keypoints. Nearest Neighbour Distance Ratio (NNDR) is used to establish the correspondences of keypoints between a pair of images. We use the reprojection error to determine whether a correspondence is correct using the equation below:

$$||p_i - (\mathbf{R}p_j + \mathbf{t})|| \leq d_{\min} \qquad (8)$$

where p_i and p_j are 3d points from frames i and j. $\mathbf{R}$ and $\mathbf{t}$ denote the groundtruth rotation and translation and are given during the evaluation. If the re-projection error is less than $d_{\min}$(set to be 0.05 m), the match is regarded as a correct one. In the next subsection, we use the *percentage of inliers* to describe the invariance of the features w.r.t scale variations and we adopt *Precision-Recall* curves to evaluate the performance of the RGB-D features under other types of variations as used in [25] .

B. Experimental Results and Analysis

In this section, we present the following comparative experiments to evaluate RISAS:

1) 3D ISS keypoint detector and RGB-D CSHOT descriptor: ISS has been combined with different 3D descriptors for evaluation in Guo et al.'s survey [17]. Implementations of these in PCL [26] were used in our experiments.
2) Uniform sampled keypoints and RGB-D CSHOT descriptor: Uniform sampling method for keypoint detection was used in Aldoma et al.'s work [27] for 3D object recognition[3] In our experiments, the uniform sampling method was adopted and the methods provided in PCL were used.
3) 2D SIFT keypoint detector and RGB-D CSHOT descriptor: We used publically available implementations of SIFT detector from VLFeat [28] and CSHOT descriptor from PCL [26]. This was used as an example of combination between a 2D keypoint detector and a RGB-D descriptor.
4) Proposed RISAS keypoint detector and RGB-D CSHOT descriptor: Matlab implementation of the RISAS detector together with the PCL implementation of CSHOT was used.

---

[1]http://rgbd-dataset.cs.washington.edu/
[2]This dataset can be downloaded from http://kanzhi.me/ rgbd-descriptor-dataset/ to make it possible for the community to use this in future research
[3]Random sampling is used in the SHOT [18] paper and CSHOT paper[1].

5) 2D SIFT feature (detector and descriptor) as implemented in VLFeat.
6) Proposed RISAS keypoint detector and LOIND descriptor that were implemented in Matlab.

All of the experiments were performed on a standard desktop PC equipped with an Intel i5-2400 CPU.

C. Object Recognition Dataset

We selected the information-rich sequence $table\_1$ from the RGB-D scene dataset [19] and we present some of the results in Fig. 8. As the figure indicates, RISAS and the combination of RISAS detector and CSHOT descriptor show larger area under the curve thus demonstrate the best performance.

(a) Image 33 and 38.   (b)

(c) Image 25 and 32.   (d)
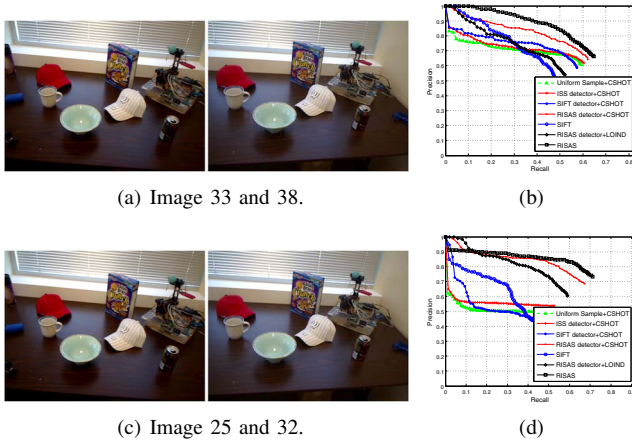
Fig. 8: Evaluation results on RGB-D scene dataset.

D. RGB-D Feature Evaluation Dataset

In the constructed dataset, we consider four common variations: 1) viewpoint, 2) illumination, 3) scale and 4) rotation.

1) Viewpoint Invariance: We collected 24 images by moving the sensor around the objects in approximately $60°$ at 0.7 meters away from the objects. The angle between each pair of consecutive frames is approximately $3°$. In order to estimate the true transformation between each pair of frames and to further evaluate the performance of descriptors, we adopted RGBD-SLAM [29] to compute the optimised poses and regarded the optimised poses as the ground-truth. We selected the image which faces straight forward to the object (in the middle with index 12) as the reference image and matched two images on both left and right side (with indices $1, 6, 18$ and 24) to the reference one. Image 12 and 24 are presented in Fig. 9. The Precision-Recall curves of these four pairs of images are shown in Fig. 10. RISAS is significantly superior compared with all other methods. CSHOT performs well when used with the RISAS detector while performing surprisingly poor with SIFT and ISS detectors, and also with uniform sampling. We also noticed that SIFT doesn't perform as expected under these scenarios with approximate $30°$ of viewpoint change.
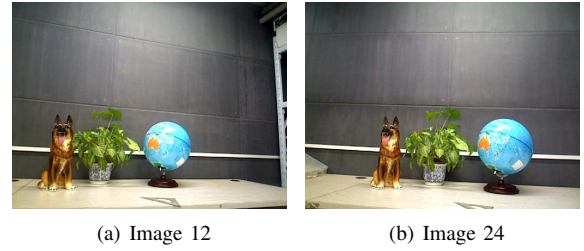

(a) Image 12   (b) Image 24

Fig. 9: Example images of viewpoint variations.

(a) Between image 12 and 1   (b) Between image 12 and 6

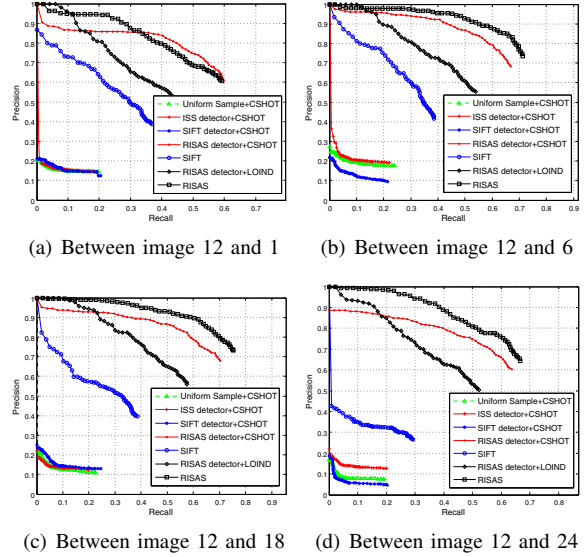(c) Between image 12 and 18   (d) Between image 12 and 24

Fig. 10: Precision-Recall curves under viewpoint variations.

*2) Illumination Invariance:* In order to validate the performance of RISAS under illumination variations, we constructed a dataset which consists of five different levels of illumination variations: 1) square 2) square root 3) cube, 4) cube root and 5) natural illumination variation, as shown in the left column in Fig. 11. The lightning condition of reference image is similar to that is shown in Fig. 12(a). As Fig. 11 demonstrates, the proposed RISAS feature shows the best performance compared with other approaches, i.e. the precision value of RISAS is almost equal to 1.0 when the recall value is 0.7 regardless of the extent of the illumination variation. It is interesting to note that SIFT performs quite well while at the same time performance of CSHOT is significantly enhanced by using it together with the RISAS detector.

3) Scale Invariance: In this experiment, we collected 10 images with the variations in z axis of the sensor coordinate system. The first frame captured at 1.1 m from the object was selected as the reference image and all other images were captured by moving the camera backwards in step of 0.1 m. A pair of images of scale variations is shown in Fig. 12 and the matching accuracy w.r.t the scale variation is shown in Fig. 13. While RISAS gives the best performance, RISAS detector used with CSHOT also demonstrates good results. All the other methods are significantly inferior.
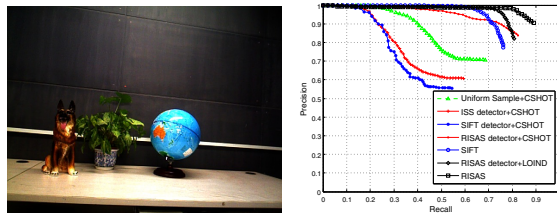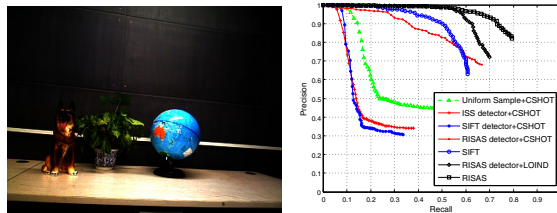
(a) Square root illumination
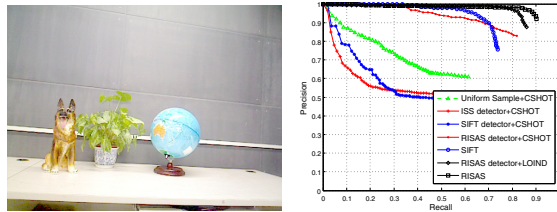
(b) Precision-Recall curve
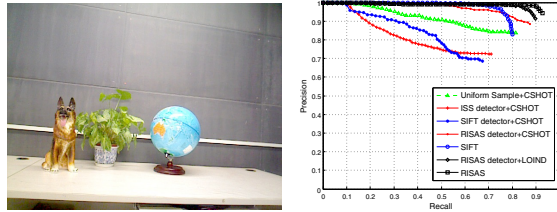
(c) Square illumination
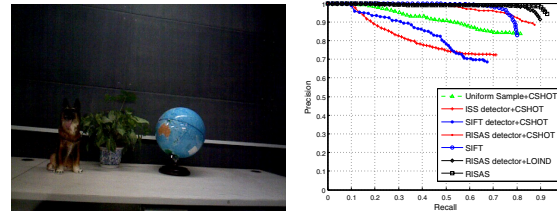
(d) Precision-Recall curve

(e) Cube root illumination

(f) Precision-Recall curve

(g) Cube illumination

(h) Precision-Recall curve

(i) Illumination change using ND mirror

(j) Precision-Recall curve

Fig. 11: RISAS evaluation under illumination variations.

4) Rotation Invariance: We evaluated RISAS under 3D rotation as illustrated in Fig. 14. The reference image is similar to that is shown in Fig. 11(i). Precision-recall curves are presented in Fig. 15. RISAS and the combination of RISAS detector and CSHOT performs best under 3D rotations.

Discussion

Results from the experiments shows that overall, RISAS provides the best results when compared with other approaches. RISAS shows clear advantages over other methods under viewpoint variations. Under illumination variations,
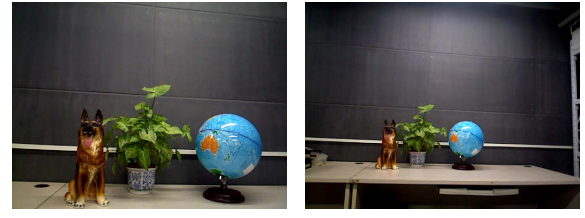
(a) Original image as reference, captured at distance $\approx 1.1m$

(b) Image captured at the distance $\approx 1.9m$
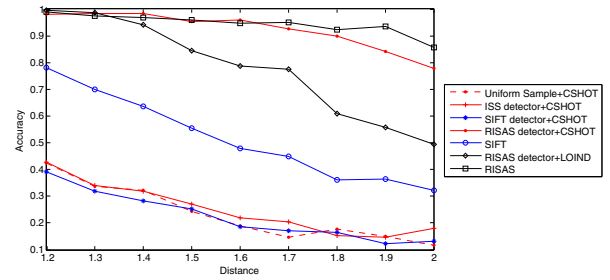
Fig. 12: Example images of scale variations.

Fig. 13: Comparative matching results under scale variations.

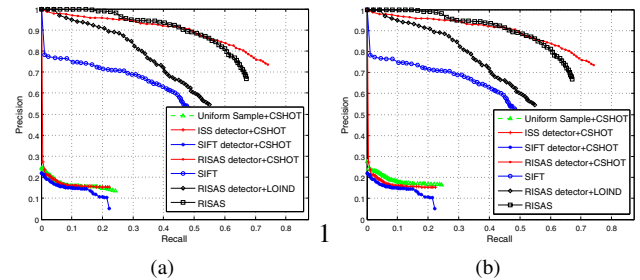Fig. 14: Example images of 3D rotations.

(a)

(b)

Fig. 15: Precision-Recall curves corresponding to Fig. 14.

RISAS outperforms other methods significantly except for LOIND. For the case of LOIND results are comparable. Under scale and rotation variations, RISAS and the combination of RISAS detector and CSHOT descriptor demonstrate the best performance.

It is clear that using the RISAS detector with CSHOT significantly enhances its matching performance. This confirms our view that a suitable RGB-D detector is critical for the performance of a RGB-D descriptor. In RISAS, the descriptor performs well if the neighbourhood of the keypoint shows higher normal vector variations. This variation is precisely what we consider in developing the detector.

In its current unoptimised Matlab based implementation, RISAS takes 20 seconds to complete both keypoint detection

and descriptor construction for an image 640×480 captured from Kinect/Xtion. On the same PC with C/C++ implementations in PCL [26], ISS[16] takes nearly 6 seconds and CSHOT takes almost 1 second to process a similar frame. Our expectation is that RISAS can be speeded up to about 2 seconds/frame when implemented in C/C++.

V. CONCLUSION

This paper presents an RGB-D feature which consists of a highly coupled RGB-D keypoint detector and descriptor. A novel 3D representation, dot-product image is combined with grayscale image to extract the keypoints using a principle similar to that of the Harris detector. We also propose an enhanced RGB-D descriptor based on our previous LOIND descriptor which significantly improves the matching performance. RISAS is demonstrated to be invariant to viewpoint, illumination, scale and rotation. RISAS detector is shown to enhance the performance of CSHOT and LOIND that are currently the best performing RGB-D descriptors. The future work will focus on adopting RISAS in various robotic applications such as object detection and point cloud alignment under challenging illumination conditions.

ACKNOWLEDGMENT

REFERENCES

[1] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3d feature matching," in *Proc. IEEE International Conference on Image Processing (ICIP'2011)*, Sep 2011, pp. 809–812.

[2] G. Feng, Y. Liu, and Y. Liao, "Loind: An illumination and scale invariant rgb-d descriptor," in *Proc. IEEE International Conference on Robotics and Automation (ICRA' 2015)*, May 2015, pp. 1893–1898.

[3] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. European Conference on Computer Vision (ECCV' 2006)*, 2006, vol. 3951, pp. 404–417.

[5] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proc. IEEE International Conference on Computer Vision (ICCV' 2011)*, Nov 2011, pp. 2564–2571.

[6] A. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 21, no. 5, pp. 433–449, May 1999.

[7] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Persistent point feature histograms for 3d point clouds," in *Proc. International Conference on Intelligent Autonomous Systems (IAS'2008)*, 2008.

[8] R. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *IEEE International Conference on Robotics and Automation (ICRA' 2015)*, May 2009, pp. 3212–3217.

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

[10] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," *Proc. European Conference on Computer Vision (ECCV'2010)*, pp. 778–792, 2010.

[11] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2548–2555.

[12] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: an efficient alternative to sift or surf," in *IEEE International Conference on Computer Vision (ICCV'2011)*. IEEE, 2011, pp. 2564–2571.

[13] R. Mur Artal and J. D. Tardos, "Probabilistic Semi-Dense Mapping from Highly Accurate Feature-Based Monocular SLAM," in *Proc. Robotics: Science and Systems (RSS)*, Rome, Italy, Jul 2015.

[14] F. Tombari, S. Salti, and L. Di Stefano, "Performance evaluation of 3d keypoint detectors," *International Journal of Computer Vision*, vol. 102, no. 1-3, pp. 198–220, 2013.

[15] R. Unnikrishnan and M. Hebert, "Multi-scale interest regions from unorganized point clouds," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*. IEEE, 2008, pp. 1–8.

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

[17] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, "A comprehensive performance evaluation of 3d local feature descriptors," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 66–89, 2016.

[18] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. European Conference on Computer Vision (ECCV'2010)*, 2010, pp. 356–369.

[19] K. Lai, L. Bo, X. Ren, and D. Fox, "Sparse distance learning for object recognition combining rgb and depth information," in *Proc. IEEE International Conference on Robotics and Automation (ICRA'2011)*, May 2011, pp. 4007–4013.

[20] E. Nascimento, G. Oliveira, M. Campos, A. Vieira, and W. Schwartz, "Brand: A robust appearance and depth descriptor for rgb-d images," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS' 2012)*, Oct 2012, pp. 1720–1726.

[21] M. Agrawal, K. Konolige, and M. R. Blas, "Censure: Center surround extremas for realtime feature detection and matching," in *European Conference on Computer Vision*. Springer, 2008, pp. 102–115.

[22] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from rgb-d images for object detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 345–360.

[23] M. Schwarz, H. Schulz, and S. Behnke, "Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1329–1335.

[24] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[25] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[26] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–4.

[27] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. Rusu, S. Gedikli, and M. Vincze, "Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation," *IEEE Robotics & Automation Magazine*, vol. 3, no. 19, pp. 80–91, 2012.

[28] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[29] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-d mapping with an rgb-d camera," *Robotics, IEEE Transactions on*, vol. 30, no. 1, pp. 177–187, Feb 2014.