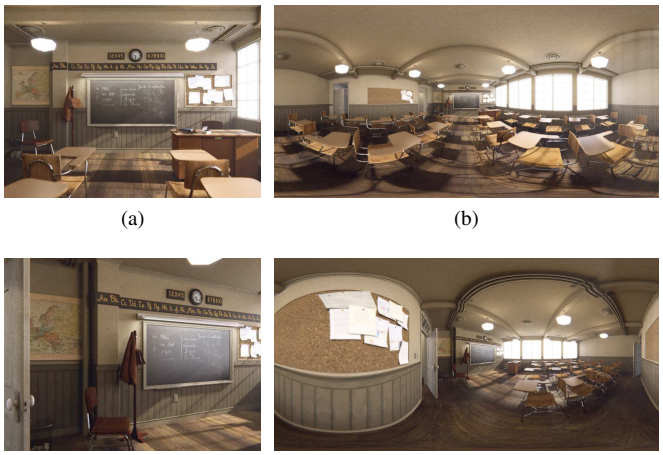
Evaluation of Keypoint Extraction and Matching for Pose Estimation using Pairs of Spherical Images

Thiago L. T. da Silveira
Institute of Informatics
Federal University of Rio Grande do Sul
Email: tltsilveira@inf.ufrgs.br

Cláudio R. Jung
Institute of Informatics
Federal University of Rio Grande do Sul
Email: crjung@inf.ufrgs.br

***Abstract*—Keypoint extraction and matching has been widely studied by the computer vision community, mostly focused on pinhole camera models. In this paper we perform a comparative analysis of four keypoint extraction algorithms applied to full spherical images, particularly in the context of pose estimation. Two of the methods chosen for the comparative study, namely A-KAZE and ASIFT, have been designed considering a perspective camera model, but were already applied in an omnidirectional structure from motion pipeline, generating successful results in the literature. The other two algorithms are properly adapted versions of the traditional descriptors SIFT and ORB to the spherical domain, subbed SSFIT and SPHORB. We conduct our tests on captures of omnidirectional cameras, both synthetic and real, arbitrarily translated and rotated with known ground-truth transformations. The extracted keypoints are fed to the well-known 8-point algorithm with RANSAC, allowing to estimate the relative camera poses. These poses (translation vector and rotation matrix) are then compared to the ground-truth transformation parameters, generating the error metrics used in our analysis. Our results indicated that spherical descriptors SSIFT and SPHORB did not produce better results than planar descriptors A-KAZE and ASIFT in the context of pose estimation, particularly in the evaluation with real image pairs.**



(a) (b) (c) (d)

Fig. 1. From left to right: perspective and omnidirectional captures of the Classroom scene. Canonical view in (a) and (b) and transformed view in (c) and (d). The camera in the latter was translated in the horizontal plane and rotated -25 degrees around the vertical axis with relation to the canonical view.

I. Introduction

Among the many algorithms and applications in image processing and computer vision, a significant portion relies on a step where relevant image points, keypoints, are extracted. Traditionally, keypoints need to encode the local information discriminatingly while being robust to affine transformations, noise and contrast changes [1]. Techniques involving multiple captures of a scene or temporal analysis of image sequences often use keypoints, along with a (keypoint) matching phase.

Although keypoint extraction is a very common task, presenting several solutions in literature when considering perspective images [2]–[6], it needs to be revisited when a different imaging paradigm arises. As an example, one can see the several algorithms dedicated to the 3D variant of this problem, being applied to data captured by, for instance, color plus depth cameras. In this specific context, prominent techniques are ISS [7], NARF [8] and adaptations of classical methods like Harris [9] and SUSAN [10].

On the other hand, few studies [1], [11] have truly addressed the keypoint extraction problem on omnidirectional imaging. Omnidirectional cameras, which can capture the complete surrounding scene with a single click, are becoming increasingly popular due to the recent release of low-cost grade-consumer devices. A variety of content is being generated for both standard devices, like desktops and smart-phones, and virtual reality (VR) head-up displays (HUDs), inciting the pair market and scientific community.

Novel applications enabled with the popularization of omnidirectional cameras, such as six degrees of freedom (6DOF) VR [12], require 3D reconstruction from multiple spherical images, which typically involves keypoint extraction and matching at some point. Once the spherical camera model has intrinsic direction-dependent distortions [13], which are easily perceived on the acquired omnidirectional images, standard perspective algorithms designed to solve a variety of problems tend to produce poor or incoherent results [1] when applied to those images.

Fig. 1 presents two views of a synthetically generated scene (Classroom benchmark[1]) in a canonical pose and after rotating and translating the camera. Perspective and omnidirectional captures (in equirectangular format) are shown side by side aiming to illustrate the aforementioned distortions (cf. structures like the mural and the pipes on the ceiling).

[1]Available under CC0 license in `https://www.blender.org`.

In this paper, we investigate the performance of four keypoint extraction algorithms when dealing with full spherical images (those that cover the $180^\circ \times 360^\circ$ scene) in the context of pose estimation. Two of them, Accelerated KAZE (A-KAZE) [6] and Affine SIFT (ASIFT) [4], although originally designed for perspective cameras, work relatively well with distorted images [14]. They were used as part of structure from motion (SfM) pipelines based on spherical images in studies like [15], [16]. Other two algorithms, namely Spherical SIFT (SSIFT) [1] and Spherical ORB (SPHORB) [11], are adaptation of known keypoint extractors/descriptors to the spherical camera projection model.

A relevant point is the definition of the error (or quality) metric used to compare different methods. In the context of keypoint extraction, repeatability [1] and precision-recall [11] have been used. However, keypoint extraction and matching is usually not the final goal in many computer vision applications: it is typically one module of a broader pipeline. One of these applications is 3D scene geometry recovery [16], which explores keypoint extraction and matching as an intermediate step, but the final goal is to extract 3D information from the scene, or estimate the camera poses.

In this paper we present an application-driven evaluation of the four selected algorithms, in which the main goal is to address the impact of each method in context of camera pose estimation. More precisely, we use a set of synthetic and real spherical image pairs with known ground truth camera poses. For a given pair, we explore keypoint matching to extract the Essential matrix using the 8-point algorithm [17], whilst removing outliers through random sample consensus (RANSAC) [18]. Then, we extract the relative camera pose (translation vector and rotation matrix) using the Singular Value Decomposition (SVD) [19]. Finally, the recovered pose is compared to the ground-truth, generating a pair of error metrics: one for translation and another for rotation.

The remaining of this paper is organized as follows. Section II briefly describes each of the selected keypoint extraction algorithms. In Section III we succinctly explain how one can recover the camera pose from correspondence points in the spherical domain and how we propose to evaluate the algorithms in this context. Section IV presents the comparative analysis in both synthetic and real scenarios and, finally, in Section V the conclusions are drawn.

II. Related work

There are few studies focusing on the performance comparison of keypoint extraction and description algorithms. In [20], 3D keypoint descriptors, such Harris, ISS and SUSAN, are assessed in terms of repeatability under rotation, translation and scale changes. In their study, no specific application of these algorithms is explored. On the other hand, in [21], 2D keypoint algorithms, like A-KAZE and SIFT [2], are evaluated in the stereo matching context, which somehow relates to the pose estimation problem. The results presented in [21] are in terms of the number of detected keypoints, proportion of valid matches, and computation time, not exactly on the quality of the produced disparity maps.

Here, we conduct our study focused on a goal application: pose estimation using pairs of spherical images. Next, we briefly describe the keypoint extraction algorithms considered in our analysis. As explained in Section I, they were chosen either for being designed for the spherical domain, or for having being used in SfM applications that explore spherical cameras. For a complete description of these techniques the readers are referred to the original papers.

A. Accelerated KAZE (A-KAZE)

A-KAZE [6], a modified version of KAZE [5], is a fast multi-scale keypoint detection and extraction algorithm which exploits non-linear scale spaces in perspective images. In the original paper [6], the authors claim that A-KAZE outperforms methods like KAZE, SIFT and ORB [3] in terms of repeatability, when applied artificial rotation, blurring, compression, noise, etc., at same time it spends lower processing time than the first two methods.

The pipeline of the A-KAZE algorithm consists basically of: (i) building the non-linear scale-space by means of fast explicit diffusion schemes in a pyramidal way; (ii) searching for maxima responses in scale and spatial locations of the scale-normalized determinant of the Hessian of each filtered image in the scale-space; and (iii) describing the keypoint by the modified-local difference binary (M-LDB) algorithm using gradient and intensity information from the scale space.

Finding the dominant local orientation around the keypoint and keeping the relation between the grid size of the M-LDB and the scale of the filtered images makes the descriptor invariant to rotation and scale. According to Pathak et al. [16], A-KAZE works well under distortions and, for that reason it was used in their spherical SfM pipeline.

B. Affine SIFT (ASIFT)

ASIFT [4] is a planar algorithm totally based on the strengths and weaknesses of the consecrated SIFT keypoint extractor and descriptor. In the original proposal, ASIFT is supposed to replace the SIFT keypoint extractor, keeping the descriptor unaltered. In practical terms, whilst SIFT achieves invariance to translation, rotation and scale, ASIFT adds invariance to "viewpoint changes", proving to be fully affine transformation invariant [4].

In a nutshell, it becomes viewpoint invariant by simulating a comprehensive set of perspective distortions that the images can suffer and comparing them by SIFT algorithm. In order to minimize the running time of their algorithm, the authors propose to use a two-resolution mechanism. The viewpoint simulations are firstly performed in the lower resolution images and, if some keypoint matching is obtained, they are redone in the original pair of images.

The authors in [4] show their improvements with relation to SIFT, among others, when dealing with significant perspective view changes. Since Pagani and colleagues [14], [15] achieved interesting results using ASIFT in their tests when dealing with spherical images, we consider this algorithm in our analysis.

C. Spherical SIFT (SSIFT)

The authors in [1] argue that standard planar keypoint algorithms cannot be correctly applied to omnidirectional images, in a geometric sense. They claim that the distortions of omnidirectional cameras when mapped to planar image representation are not affine and, more than that, are dependent on the object position in the captured scene. In fact, an example of these distortions is illustrated in Fig. 1.

In this sense, Cruz-Mota et al. [1] completely adapt the planar SIFT algorithm to the spherical domain, with omnidirectional images mapped to the surface of the Riemannian sphere. Each step of original SIFT keypoint detector is performed on its spherical counterpart: (i) creation of the scale-space representation, (ii) computation of the Difference of Gaussians (DoGs); and (iii) local extrema extraction and filtering. The proposed local spherical descriptor, as its planar version, is invariant to rotation and scale.

The authors assess the performance of the SSIFT against its planar version through the repeatability metric on spherical images synthetically rotated and corrupted with noise. The results presented in their paper show that the SSIFT algorithm is considerably more robust to the omnidirectional sensor distortions than planar SIFT.

As pointed out as a possible application of SSIFT in [1], [13] use it as a fundamental building block for their spherical SfM-based method and application.

D. Spherical ORB (SPHORB)

The authors in [11] propose a scale invariant version of the FAST detector [22] associated to a rotation invariant ORB-like descriptor, both operating on a hexagonal geodesic grid representation of the sphere. More precisely, they presented a fast and robust algorithm that constructs binary features to describe image keypoints on the spherical domain, called Spherical ORB (SPHORB).

The way omnidirectional images are represented is probably one of the main insights of their work. The authors show that the geodesic grid they use has important properties when dealing with binary features that, differently from cubic and equirectangular representations of the spherical images, helps to speed up the SPHORB algorithm.

Basically, the spherical FAST detector searches for points that are sufficiently brighter or darker than their neighborhood and attributes a weight for how distinguishable from its vicinity they are. This procedure is performed using a pyramidal structure, enabling SPHORB with robustness to scale changes. The bit-string that describes each of the selected keypoints is nothing but a series of intensity comparisons that are further reoriented in order to keep the algorithm invariant to rotations.

Zhao and collaborators [11] compare their algorithm, among others, with SSIFT and planar versions of ORB and SIFT. Their results point out the effectiveness of SPHORB regarding repeatability, precision and recall under synthetic rotation and noise corruption. The authors also present some statistics regarding the proportion of correct matchings in real pairs of images on small camera change setups.

III. The Proposed Methodology

We investigate an application-oriented evaluation of keypoint extraction/matching algorithms, targeting the two-view pose estimation problem in the spherical domain.

In this section, we briefly explain the epipolar geometry for spherical cameras, and explain the adopted algorithm to retrieve the Essential matrix based on a set of matched keypoints. Then, we revise the extraction of extrinsic camera parameters (rotation matrix and translation vector) from the Essential matrix, and introduce the proposed metrics to compare two poses (which implicitly evaluates the quality of the underlying keypoints).

A. Epipolar Geometry for Spherical Cameras

The core of spherical cameras is to project a 3D point $\mathbf{X}$ in the world coordinate system onto the unit sphere [23], as illustrated in Fig. 2.

If the spherical camera presents extrinsic parameters $[\mathbf{R}|\mathbf{t}]$, where $\mathbf{R}$ is the rotation matrix and $\mathbf{t}$ the translation vector, the projected point $\mathbf{x}$ is given by

$$\mathbf{x} = \frac{\mathbf{R}\mathbf{X} + \mathbf{t}}{\|\mathbf{R}\mathbf{X} + \mathbf{t}\|}. \tag{1}$$

Note that the projected point is a unit vector in $\mathbb{R}^3$, which can be rewritten in terms of spherical coordinates as

$$\mathbf{x} = [\cos\theta \sin\phi,\ \sin\theta \sin\phi,\ \cos\phi]^\top, \tag{2}$$

with $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi)$. Such a point can be mapped to position (x, y) of an equirectangular $w \times h$ image, which is the standard representation of spherical images, where $x = \frac{\theta w}{2\pi}$ and $y = \frac{\phi h}{\pi}$, rounded to integer values.

The authors in [23] show that the epipolar geometry for the spherical projection is analogous to its counterpart defined for perspective images. In the perspective case, a pair of projections $\mathbf{x}_1 = [x_1\, y_1\, z_1]^\top$ and $\mathbf{x}_2 = [x_2\, y_2\, z_2]^\top$ of a world point $\mathbf{X}$ in homogeneous coordinates can be related according to the epipolar constraint [23]

$$\mathbf{x}_2^\top \mathbf{E} \mathbf{x}_1 = 0, \quad \mathbf{E} = [\mathbf{t}]_\times \mathbf{R}, \tag{3}$$

where $\mathbf{R}$ is the rotation matrix and $[\mathbf{t}]_\times$ is the skew-symmetric matrix [13], [24] of translation vector $\mathbf{t}$ that relate both camera captures.

For spherical cameras, the projection onto the unit sphere corresponds to a particular set of homogeneous coordinates of the planar perspective projection, so that the underlying 3D vector presents unitary norm. Hence, the same constraints shown in Eq. (3) are valid, using directly 3D spherical coordinates instead of homogeneous coordinates.

Estimating the Essential matrix $\mathbf{E}$ can be formulated as a least squares problem in which $n \geq 8$ correspondence pairs are required. In the 8-point algorithm [17] the solution to this problem is achieved by selecting and reshaping the unit eigenvector corresponding to the smallest eigenvalue of $\mathbf{A}^\top\mathbf{A}$, where the i-th row of $\mathbf{A}$ is given by

$$\mathbf{A}_i^\top = \left[x_1^i x_2^i\ x_1^i y_2^i\ x_1^i z_2^i\ y_1^i x_2^i\ y_1^i y_2^i\ y_1^i z_2^i\ z_1^i x_2^i\ z_1^i y_2^i\ z_1^i z_2^i\right],$$

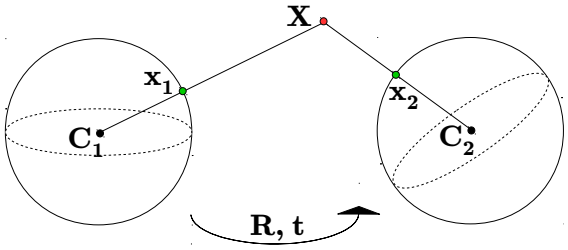


Fig. 2. Epipolar geometry for a stereo pair of spherical cameras. Points $\mathbf{X}$, $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{C}_1$ and $\mathbf{C}_2$ are the 3D world point, projection on the first and second cameras, and the centers of the first and second cameras, respectively. The cameras are related by the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$.

and $[x_1^i\ y_1^i\ z_1^i]^\top$, $[x_2^i\ y_2^i\ z_2^i]^\top$ are the spherical coordinates of the i-th correspondence pair, for $i = 1, \ldots, n$. As argued in [16], unlike in perspective case, no coordinates normalization pre-processing is required when dealing with spherical images.

In order to guarantee that the matrix $\mathbf{E}'$ obtained by this process is in fact an Essential matrix, rank two constraint must be imposed, for instance, using SVD [17]. More precisely, if $\mathbf{U}'\mathbf{S}'\mathbf{V}'^\top = \mathbf{E}'$ is the SVD of $\mathbf{E}'$, then its two first singular values are set to their average value and the last one is zeroed [24]

$$\mathbf{S} = \operatorname{diag}\left(\frac{s_{11}+s_{22}}{2}, \frac{s_{11}+s_{22}}{2}, 0\right), \tag{4}$$

where $\mathbf{S}' = \operatorname{diag}(s_{11}, s_{22}, s_{33})$. Moreover, if the determinants of $\mathbf{U}'$ and $\mathbf{V}'^\top$ are negative, then the last column of these matrices is negated, avoiding the estimation of improper rotation matrices [25]:

$$\mathbf{U}_3 = \begin{cases} \mathbf{U}'_3, & \text{if } \det(\mathbf{U}') > 0 \\ -\mathbf{U}'_3, & \text{otherwise} \end{cases} \tag{5}$$

and

$$\mathbf{V}_3^\top = \begin{cases} \mathbf{V}'^\top_3, & \text{if } \det(\mathbf{V}') > 0 \\ -\mathbf{V}'^\top_3, & \text{otherwise} \end{cases}. \tag{6}$$

Finally, the Essential matrix is given by $\mathbf{E} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$.

In our study, this procedure is applied in a RANSAC framework in order to better estimate $\mathbf{E}$, whilst rejecting a small number of erroneous keypoint matches (outliers). The correspondence check is given by the Sampson distance [24] thresholded to 10^{-2}, and the minimum quantity of inliers is set to 50% the number of matchings, i.e. n.

B. Pose Estimation from Essential Matrix

Pose estimation can be understood as obtaining the translation vector $\mathbf{t}$ and rotation matrix $\mathbf{R}$ that relate two views. Once the Essential matrix $\mathbf{E}$ is estimated, the extraction of candidates for these movement components is immediate. Thus, the rotation matrix is given by

$$\mathbf{R} = \mathbf{U}\mathbf{W}\mathbf{V}^\top \quad \text{or} \quad \mathbf{R} = \mathbf{U}\mathbf{W}^\top\mathbf{V}^\top, \tag{7}$$

where $\mathbf{U}\mathbf{S}\mathbf{V}^\top = \mathbf{E}$ is the SVD of $\mathbf{E}$ and $\mathbf{W}$ is given by [24]

$$\mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{8}$$

The translation vector is defined, up to a scale, as

$$\mathbf{t} = -\mathbf{U}_3 \quad \text{or} \quad \mathbf{t} = \mathbf{U}_3. \tag{9}$$

To get rid of this ambiguity one can check for the positivity of [14]

$$\mathbf{x_1}^\top\tilde{\mathbf{X}} \quad \text{and} \quad \mathbf{x_2}^\top(\mathbf{R}\tilde{\mathbf{X}} + \mathbf{t}),$$

where $\tilde{\mathbf{X}} \approx \alpha\mathbf{X}$ and $\alpha \in \mathcal{R}$, can be computed from the candidates $\mathbf{R}$ and $\mathbf{t}$ through, for instance, direct linear transformation (DLT) [24].

C. Comparing Two Poses

The core of this paper is to evaluate how well different keypoint extraction/matching methods work in the context of two-view spherical pose estimation. For that purpose, it is important do define a metric that evaluates how close the estimated extrinsic parameters are from the ground-truth pose.

It is also important to point out that the translation vector can only be estimated up to a scale factor from the Essential matrix. Hence, in this work we explore only the direction of the estimated translation vector. More precisely, we compare the ground-truth and estimated translation vectors, $\mathbf{t}_{gt}$ and $\mathbf{t}_{est}$, by their angular distance [19]

$$\mathrm{d}_\mathbf{t}(\mathbf{t}_{gt}, \mathbf{t}_{est}) = \cos^{-1}\left(\frac{\mathbf{t}_{gt}.\mathbf{t}_{est}}{\|\mathbf{t}_{gt}\|\|\mathbf{t}_{est}\|}\right). \tag{10}$$

On the other hand, the actual and estimated rotation matrices, $\mathbf{R}_{gt}$ and $\mathbf{R}_{est}$, are compared by their Riemannian distance [26], which is defined as the angle between two elements of $SO(3)$ group needed to make them equal. The Riemannian distance between $\mathbf{R}_{gt}$ and $\mathbf{R}_{est}$ is given by

$$\mathrm{d}_\mathbf{R}(\mathbf{R}_{gt}, \mathbf{R}_{est}) = \frac{1}{\sqrt{2}}\|\log(\mathbf{R}_{gt}^\top.\mathbf{R}_{est})\|_F, \tag{11}$$

where

$$\log(\mathbf{R}) = \begin{cases} \mathbf{0}, & \text{if } \psi = 0 \\ \frac{\psi}{2\sin\psi}(\mathbf{R} - \mathbf{R}^\top), & \text{otherwise} \end{cases}, \tag{12}$$

$$\psi = \cos^{-1}\left(\frac{\operatorname{tr}(\mathbf{R}) - 1}{2}\right). \tag{13}$$

and $\operatorname{tr}(.)$ is the trace operation.

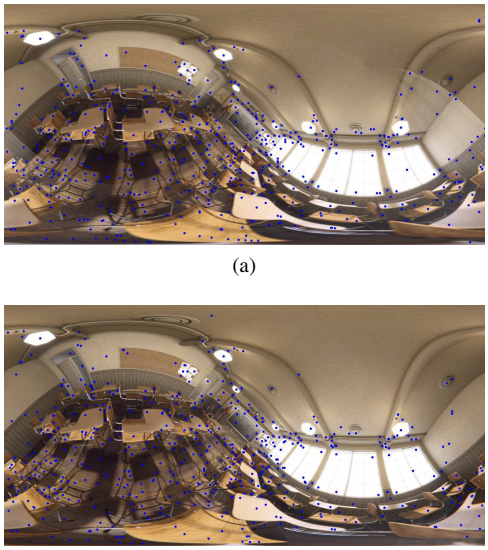
As both functions $\mathrm{d}_\mathbf{t}(\cdot,\cdot)$ and $\mathrm{d}_\mathbf{R}(\cdot,\cdot)$ are error metrics, given in radians, smaller values indicate better estimates.

IV. Experimental results

In this section we assess the considered keypoint extraction algorithms in terms of the error when estimating camera pose. We consider both synthetic and real scenes, where camera has arbitrary translation and rotation with known ground-truth.

Here, OpenCV[2] implementations of A-KAZE and ASIFT, and the original source codes for SSIFT[3] and SPHORB[4] provided by the authors are used. Adjustable parameters are

[2]Available in `http://opencv.org`.
[3]Available in `http://www.javiercruz.com`
[4]Available in `https://github.com/tdsuper/SPHORB`



(a)

(b)

Fig. 3. SSIFT keypoints computed from (a) a rotated image based on the canonical capture in Fig.1b and (b) rendered image with camera rotation set in 3D space. Keypoints in both captures are highlighted in the blue dots.

kept as indicated in each study, differently from [11], where the methods were tunned aiming to get approximately the same number of keypoints. The ratio matching strategy [27], thresholded in 0.75, is adopted here since it is natively applied in each one of the source codes. Once the considered source codes are in different programming languages, C++ and Matlab, we did not evaluate running times in this analysis, focusing on accuracy only.

In the following, the experimental setups are described and the obtained results are presented and discussed.

A. Synthetically Generated Scene Experiments

In [1], [11], omnidirectional images are projected to the sphere, rotated with relation to the principal camera axis, and then backprojected to the plane in equirectangular format, allowing the authors to have a "pair of views" for testing their methods. This process may cause both spurious insertion and loss of information, since it is needed to interpolate regions of the scene that now occupy more pixels than in the original unrotated image and to downsample in the reverse case. As a consequence, keypoint extraction and description, and possibly matching, may be affected, as depicted in Fig. 3. Although most keypoints are detected approximately around the same region in both equirectangular images, the reader can see that in the artificially rotated image some of them are missing, and others that are not effectively important are detected .

Differently from those studies, we would like to simulate rotations in all the three axes besides translational movements without suffering from the aforementioned problems. Thus, we create a dataset[5] with different views of a synthetically generated 3D scene (as illustrated in Fig. 1b and Fig. 4), allowing us to precisely set the camera poses. The realistic Classroom scene, rendered with Blender[6], was chosen since it presents a variety of repetitive structures (the lamps, the chairs, the pipes on the ceiling, the windows, etc.) configuring a challenging scenario for the selected application. Furthermore, there are parts of the scene which are practically textureless (the ceiling, the doors, the top portion of the walls, etc.), also making the keypoint matching task harder. Although our goal was to perform our analysis using several synthetic environments, we were not able to find other realistic indoor scenes with free license.

The views are captured in such a way that the camera is translated twice the magnitude for one direction with relation its opposite, one at time, along the three axes of Cartesian system coordinates. Furthermore, the rotation angles are set to 15° in one direction and 45° to its opposite direction along the three Cartesian axes, again one at time. Once the three dimensions of the scene differ in the 3D space, the distortions are perceived differently. With that, we introduce in our experiments a small but representative set of distortions caused by the six degrees of freedom motion to the generated equirectangular images. These twelve camera transformations (translations and rotations), starting from the canonical view shown in Fig. 1b, are depicted in Fig. 4.

Considering the way this dataset was organized, we divide our experiments focusing separately on the three kinds of camera movement: purely translational, purely rotational and both translation and rotational. Differently from the papers that propose A-KAZE, ASIFT, SSIFT and SPHORB, which assess only rotation movement, we evaluate the robustness of each method in the pose estimation problem considering all pairwise combinations of those twelve camera transformations, besides the canonical view. Hence, there are $P(7,2) = 42$ permutations of pairs in pure translation and pure rotational view changes, and $P(13,2) - 2P(7,2) = 72$ permutations for mixed translation *and* rotational movement.

It is important to highlight that we can only assess the selected algorithms in the pose estimation context when RANSAC is able to accept at least 50% of matchings as inliers during the Essential matrix estimation, as discussed in Section III-C. In the evaluation, we first analyzed the percentage of times RANSAC fails to converge for each method, as well as the percentage of inliers within the total number of matched keypoints. When RANSAC converged (and the pose was effectively estimated), we also compute the average angular and Riemannian errors, along with the standard deviations. A summary of results is shown in Table I.

One can note from Table I that SSIFT, although presenting the best average angular distance when estimating the pose, failed 21 out of 42 times it was tested in the experiment that consider only translation. Also, when RANSAC converged for SSIFT, the inlier ratio was very close to the limit threshold for RANSAC, i.e., 50%. These results indicates that SSIFT, with the original parameters, generates some very accurate matches,

[5]This dataset will be available on the authors website.

[6]Available in `http://www.blender.org`

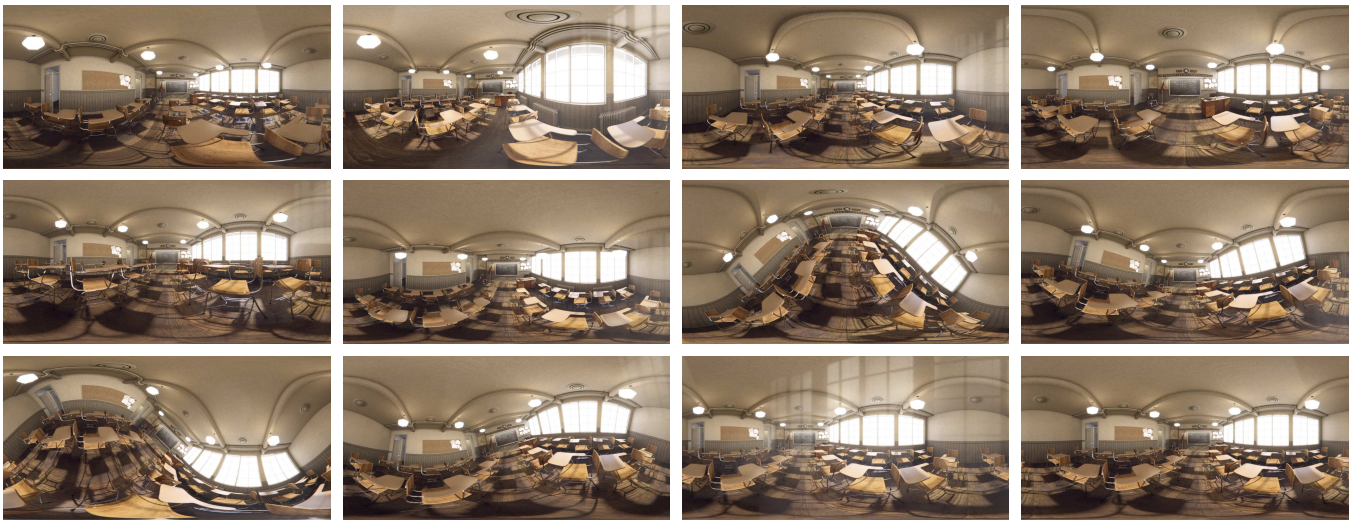


Fig. 4. Synthetic dataset of the Classroom scene with different pose transformations starting from the canonical position in Fig.1b. In pairs, the first six images present camera translations of different magnitudes towards the positive and negative directions of the three axes of the Cartesian coordinate system. The other six images, also in pairs, present camera rotations in different proportions towards the positive and negative angles around the same axes.

TABLE I
AVERAGE RESULTS FOR THE SYNTHETIC SCENE VIEWS EXPERIMENTS.

Method	Purely translational			Purely rotational			Translational and rotational			
	Angular dist. (rad)	Inliers (%)	RANSAC failures (%)	Riemann dist. (rad)	Inliers (%)	RANSAC failures (%)	Angular dist. (rad)	Riemann dist. (rad)	Inliers (%)	RANSAC failures (%)
A-KAZE	0.411 ± 0.729	66.478	0.000	0.079 ± 0.157	89.486	0.000	0.262 ± 0.508	0.019 ± 0.038	67.709	0.000
ASIFT	0.252 ± 0.435	78.430	0.000	0.051 ± 0.151	88.617	4.762	0.262 ± 0.365	0.020 ± 0.050	84.733	0.000
SSIFT	0.133 ± 0.303	58.490	50.000	0.078 ± 0.156	89.335	0.000	0.231 ± 0.315	0.031 ± 0.044	61.732	16.667
SPHORB	0.281 ± 0.509	67.041	0.000	0.077 ± 0.159	84.486	2.380	0.346 ± 0.511	0.040 ± 0.084	63.424	0.000

but also some very bad ones (at least for for this evaluation scenario).

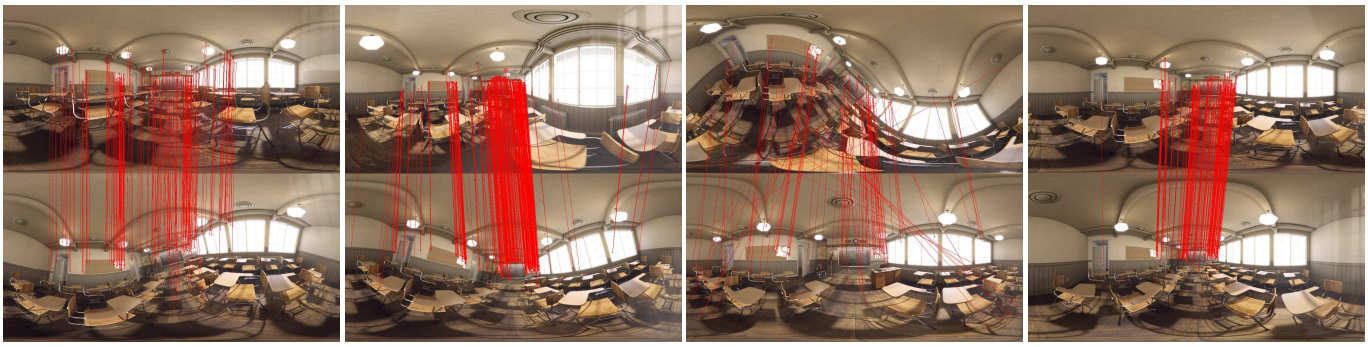
SSIFT presented a similar behavior for the mixed camera movement experiments: RANSAC did not converge in one sixth of cases, but when it did the angular translational error was the smallest. However, on average, very close results were achieved by the two planar algorithms. For pure rotational movements, ASIFT performed better than the other three algorithms. ASIFT and SPHORB failed by far less then SSIFT, and only in rotational tests. Their performance in the translational experiment was similar to each other, but ASIFT generated smaller errors (both translational and rotational) than SPHORB in mixed camera movement tests.

One can also note from Table I that, in purely rotational movement, the inlier ratio for all the compared methods was quite similar. This kind of movement is exactly the one which is tested in the works that originally propose the techniques designed for spherical domains, namely SSIFT and SPHORB. Fig. 5 depicts the general behavior when applying the compared method to the synthetic view pairs. SSIFT tends to spread much more the matchings throughout the images, whilst SPHORB frequently find several matches around the same neighborhoods. A-KAZE and ASIFT perform similarly to SSIFT and SPHORB, respectively, although in most of the cases the matches are around the image center, which is less deformed. This behavior is briefly commented in [1], when SSIFT is compared to its planar counterpart.

B. Real scene experiments

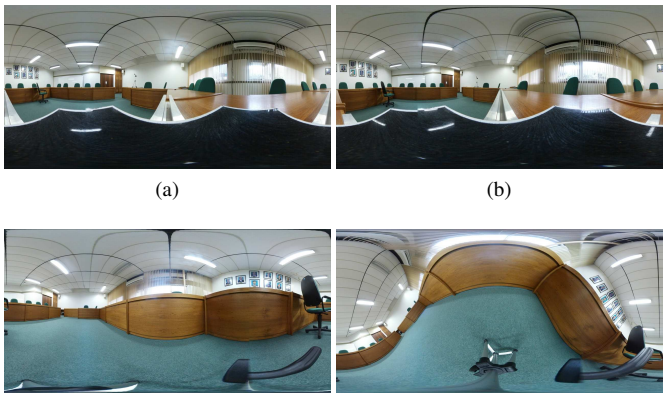
Besides assessing the methods using synthetic images, we also present the analysis for a set of real scene captures. The real omnidirectional images were captured with the first release of the Samsung Gear 360 camera, and stitching was performed by their official software. The captures were obtained in five indoor and static scenarios with meticulously positioned cameras over a fixed trail with a rotating base, for performing horizontal translations and rotations on the plane, and in a height adjustable tripod, for applying vertical translations and rotations along the three axes. Fig. 6 depicts two pairs of captures.

Although the controlled camera positioning provides a clue of its pose, we obtain well approximated rotation matrices and translation vectors by manually selecting about twenty correspondence points in the equirectangular image pairs in such a way that the selected points appear spatially distributed throughout the images. They are then used to estimate the pose as described in Section III with no RANSAC outliers (they are assumed to be correct). Since it is practically impossible



(a) A-KAZE (89.630%) (b) ASIFT (77.130%) (c) SSIFT (64.400%) (d) SPHORB (65.437%)

Fig. 5. Examples of matching pairs set as inliers on both translational and rotational camera movement using the four considered keypoint extraction algorithms. Inlier proportion is given for each pair of views.



(a) (b) (c) (d)

Fig. 6. Two pairs of real scene captures used in our experiments. Camera transformation in pair (a–b) is dominantly translational on the horizontal plane, and in pair (c–d) it is basically rotational (with a rotation angle of -60° on the vertical axis).

to keep the camera without the rotations (translations) when performing the translational (rotational) experiments, we refer to the real scene experiments as "dominantly" translational or rotational, instead of "pure", and both translational *and* rotational when the pair of transformations were designedly applied.

The average results for a total of fifteen pairs of images are shown in Table II, which corroborate the findings for the synthetic experiments. It is noticeable, in both angular distance and inlier ratio columns in Table II, that the methods performed significantly better in dominantly translational movement experiments. The results for these techniques were, on average, also better in dominantly rotational camera movement when considering real scene tests instead of the synthetic scene ones. A possible explanation is the fact the real scenes do not have so many "ambiguous scene objects", which might facilitate keypoint extraction, description and matching.

Representative examples of real matching pairs obtained by the four considered methods are presented in Fig. 7. It is interesting to note the high inlier ratio achieved by ASIFT, a planar keypoint extractor/descriptor, even in very distorted images, as presented in Fig. 7b.

V. Conclusion

In this paper we evaluated four algorithms for extraction and description of keypoints on the context of pose estimation using full spherical images. Two of them were originally proposed for perspective cameras (planar descriptors), and the other ones tailored to the spherical domain. We selected two figures of merit that measure the error between estimated translation vector and rotation matrix from the respective ground-truths. According to the experiments performed in this work, with both a challenging synthetic dataset and real indoor image pairs, using spherical descriptors did not produce more accurate pose estimates than using traditional planar descriptors. It is important to note that in the papers that presented SSIFT and SPHORB, the authors showed that spherical descriptors performed better than planar ones. Since they used generic metrics in their evaluation (e.g. repeteability), their findings and ours are not necessarily contradictory. Nevertheless, we plan to further investigate if this discrepancy was due to the evaluation procedure (generic versus application-driven) or to the used datasets.

As future work, we intend to enlarge our database to expand the comparative analysis. We also intend to investigate how the spatial distribution of the extracted keypoints impacts the pose estimation and further 3D reconstruction of the environments.

Acknowledgment

The authors would like to thank Brazilian agencies CNPq and Capes.

References

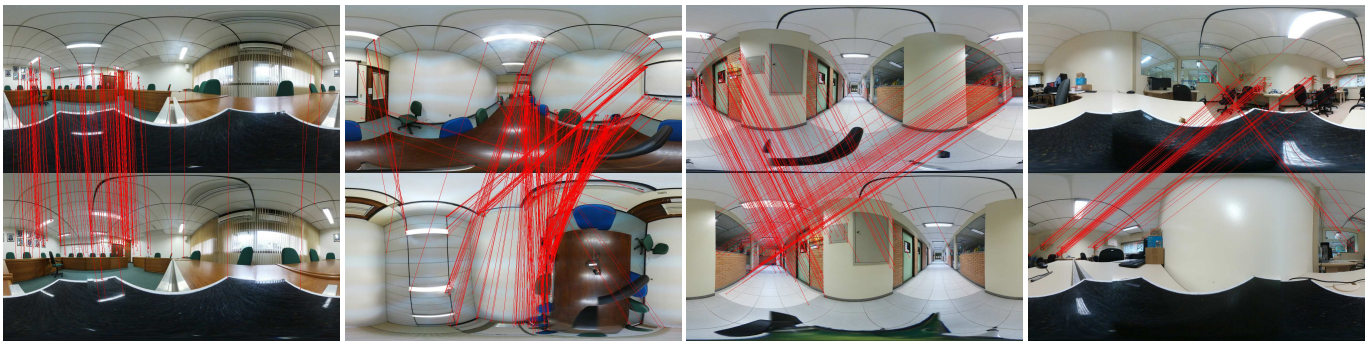
[1] J. Cruz-Mota, I. Bogdanova, B. Paquier, M. Bierlaire, and J. P. Thiran, "Scale invariant feature transform on the sphere: Theory and applications," *International Journal of Computer Vision*, vol. 98, no. 2, pp. 217–241, 2012.

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

[3] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571.

TABLE II
AVERAGE RESULTS FOR THE REAL SCENE VIEWS EXPERIMENTS.

Method	Dominantly translational			Dominantly rotational			Translational and rotational			
	Angular dist. (rad)	Inliers (%)	RANSAC failures (%)	Riemann dist. (rad)	Inliers (%)	RANSAC failures (%)	Angular dist. (rad)	Riemann dist. (rad)	Inliers (%)	RANSAC failures (%)
A-KAZE	0.082 ± 0.050	75.930	0.000	0.044 ± 0.030	68.244	0.000	0.306 ± 0.217	0.106 ± 0.095	60.557	0.000
ASIFT	0.090 ± 0.051	83.262	0.000	0.082 ± 0.078	69.480	0.000	0.193 ± 0.214	0.075 ± 0.076	78.321	0.000
SSIFT	0.100 ± 0.062	67.570	0.000	0.061 ± 0.047	69.780	0.000	0.299 ± 0.178	0.083 ± 0.084	56.447	20.000
SPHORB	0.080 ± 0.060	73.669	0.000	0.046 ± 0.037	65.174	0.000	0.295 ± 0.221	0.120 ± 0.102	66.044	0.000



(a) A-KAZE (75.091%) (b) ASIFT (90.246%) (c) SSIFT (74.097%) (d) SPHORB (60.156%)

Fig. 7. Examples of matching pairs set as inliers on (a) dominantly translational, (b), (d) translational and rotational, and (c) dominantly rotational camera movements using the four considered keypoint extraction algorithms. Inlier proportion is given for each pair of views.

[4] G. Yu and J.-M. Morel, "ASIFT: An Algorithm for Fully Affine Invariant Comparison," *Image Processing On Line*, vol. 1, pp. 11–38, 2011.

[5] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, *KAZE Features*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 214–227.

[6] P. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013.

[7] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3d object recognition," in *IEEE International Conference on Computer Vision*, 2009, pp. 689–696.

[8] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 2601–2608.

[9] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proc. of Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[10] S. M. Smith and J. M. Brady, "SUSAN—a new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45–78, 1997.

[11] Q. Zhao, W. Feng, L. Wan, and J. Zhang, "SPHORB: A Fast and Robust Binary Feature on the Sphere," *International Journal of Computer Vision*, vol. 113, no. 2, pp. 143–159, 2015.

[12] Jingwei Huang, Zhili Chen, Duygu Ceylan, and Hailin Jin, "6-DOF VR Videos with a Single 360-Camera," in *Proceedings of the IEEE Virtual Reality*, 2017.

[13] H. Guan and W. A. P. Smith, "Structure-From-Motion in Spherical Video Using the von Mises-Fisher Distribution," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 711–723, 2017.

[14] A. Pagani and D. Stricker, "Structure from Motion using full spherical panoramic cameras," in *IEEE International Conference on Computer Vision*, 2011, pp. 375–382.

[15] A. Pagani, C. Gava, Y. Cui, B. Krolla, J.-M. Hengen, and D. Stricker, "Dense 3D Point Cloud Generation from Multiple High-resolution Spherical Images," *International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)*, pp. 1–8, 2011.

[16] S. Pathak, A. Moro, H. Fujii, A. Yamashita, and H. Asama, "3D reconstruction of structures using spherical cameras with small motion," in *International Conference on Control, Automation and Systems*, 2016, pp. 117–122.

[17] R. Hartley, "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580–593, 1997.

[18] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[19] J. E. Gentle, *Numerical Linear Algebra for Applications in Statistics*. Springer New York, 1998.

[20] S. Filipe and L. A. Alexandre, "A comparative evaluation of 3d keypoint detectors in a rgb-d object dataset," in *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, vol. 1, 2014, pp. 476–483.

[21] A. Satnik, R. Hudec, P. Kamencay, J. Hlubik, and M. Benco, "A comparison of key-point descriptors for the stereo matching algorithm," in *International Conference Radioelektronika*, 2016, pp. 292–295.

[22] E. Rosten and T. Drummond, *Machine Learning for High-Speed Corner Detection*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 430–443.

[23] T. Akihiko, I. Atsushi, and N. Ohnishi, "Two-and three-view geometry for spherical cameras," *Proceedings of the Sixth Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, vol. 105, pp. 29–34, 2005.

[24] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, 2003.

[25] W. X. W. Xu and J. Mulligan, "Robust relative pose estimation with integrated cheirality constraint," *International Conference on Pattern Recognition*, no. m, pp. 6–9, 2008.

[26] M. Moakher, "Means and averaging in the group of rotations," *SIAM Journal on Matrix Analysis and Applications*, vol. 24, no. 1, pp. 1–16, 2002.

[27] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," in *IEEE Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2, 2003, pp. II–257–II–263 vol.2.